

**UNITED STATES INTERNATIONAL TRADE COMMISSION
WASHINGTON, D.C.**

**Before the Honorable Cameron Elliot
Administrative Law Judge**

**IN THE MATTER OF CERTAIN AUDIO
PLAYERS AND COMPONENTS
THEREOF II**

Investigation No. 337-TA-1330

COMPLAINANT'S INITIAL MARKMAN BRIEF

TABLE OF CONTENTS

TABLE OF CONTENTS.....	ii
TABLE OF AUTHORITIES.....	iv
TABLE OF EXHIBITS	vi
I. INTRODUCTION.....	1
II. TECHNOLOGY BACKGROUND.....	2
A. Voice Assistant Technologies Relating to the '748, '128, and '311 Patents.....	3
B. Device Commissioning Technologies Relating to the '615 Patent.....	7
III. LEVEL OF ORDINARY SKILL IN THE ART	8
IV. LEGAL STANDARDS	9
A. Claim Construction.....	9
B. Indefiniteness	10
C. Correcting Obvious Errors	10
V. '748 Patent.....	10
A. Overview of the '748 Patent.....	10
B. Agreed Constructions.....	11
C. “[perform/performing] speech recognition” (All Claims).....	12
D. “a command to temporarily refrain from performing speech recognition” (All Claims) 15	
E. “predetermined spoken audio command” (All Claims).....	17
F. “[determine/determining] ... based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input” (All Claims).....	19
VI. '128 and '311 Patents.....	23
A. Overview of the '128 and '311 Patents	23
B. “other devices of the plurality of electronic devices” / “quality scores generated by the other devices of the plurality of electronic devices” ('128 All Claims)	25
C. “second values ... from other electronic devices” / “second values from other electronic devices of the plurality of electronic devices” ('311 All Claims).....	28
D. “forgo outputting an audible response” / “forgoing outputting a response” ('128 All Claims) / “[forgoing / forgo] responding” ('311 Claims 2, 3, 12).....	29
E. “in accordance with a determination that the quality score generated by the first electronic device is the highest ... in accordance with a determination that the quality score generated by the first electronic device is not the highest” ('128 Claims 1, 2, 3, 5) / “in accordance with a determination that the first value is higher than the second values ... in	

accordance with a determination that the first value is not higher than the second scores” (’311 Claim 2).....	33
F. “cause the electronic device” (’128 Claims 11, 12, 13, 15).....	37
G. “the second scores” (’311 Claims 2, 3).....	39
H. “the detected output” (’311 Claim 12)	42
VII. ’615 Patent.....	44
A. Overview of the ’615 Patent.....	44
B. “short range signal transmission path” (All Claims)	45
C. “in response to receiving the request, receiving one or more network configuration packets” (All Claims).....	47
VIII. CONCLUSION.....	49

TABLE OF AUTHORITIES

Page

Cases

<i>AIA Eng'g Ltd. v. Magotteaux Int'l S/A</i> , 657 F.3d 1264 (Fed. Cir. 2011).....	36
<i>Bicon, Inc. v. Straumann Co.</i> , 441 F.3d 945 (Fed. Cir. 2006)	21
<i>Certain Multiple Mode Outdoor Grills and Parts Thereof</i> , Inv. No. 337-TA-895, Comm'n Op., 2015 WL 13817118 (Feb. 3, 2015)	10
<i>Cheese Sys., Inc. v. Tetra Pak Cheese & Powder Sys., Inc.</i> , 725 F.3d 1341 (Fed. Cir. 2013).....	26
<i>Cybersettle, Inc. v. Nat'l Arbitration Forum, Inc.</i> , 243 F. App'x 603 (Fed. Cir. 2007).....	34
<i>Dayco Prods., Inc. v. Total Containment, Inc.</i> , 258 F.3d 1317 (Fed. Cir. 2001).....	27
<i>Eidos Display, LLC v. AU Optronics Corp.</i> , 779 F.3d 1360 (Fed. Cir. 2015).....	39
<i>Festo Corp. v. Shoketsu Kinzoku Kogyokabushiki Co.</i> , 535 U.S. 722 (2002).....	10
<i>GE Lighting Sols., LLC v. AgiLight, Inc.</i> , 750 F.3d 1304 (Fed. Cir. 2014).....	28
<i>Helmsderfer v. Bobrick Washroom Equip., Inc.</i> , 527 F.3d 1379 (Fed. Cir. 2008).....	9
<i>Hoffer v. Microsoft Corp.</i> , 405 F.3d 1326 (Fed. Cir. 2005).....	10
<i>Howmedica Osteonics Corp. v. DePuy Orthopaedics, Inc.</i> , No. 11-CV-6498 SDW MCA, 2013 WL 3455727 (D.N.J. July 9, 2013).....	44
<i>Hytera Commc'ns Co. v. Motorola Sols., Inc.</i> , 841 F. App'x 210 (Fed. Cir. 2021).....	34, 37
<i>Interactive Gift Express, Inc. v. Compuserve, Inc.</i> , 256 F.3d 1323 (Fed. Cir. 2001).....	9
<i>Interdigital Tech. Corp. v. Lenovo Holding Co., Inc.</i> , No. CV 19-1590-LPS, 2021 WL 1856937 (D. Del. May 10, 2021).....	35, 37
<i>Joy MM Delaware Inc. v. Cincinnati Mine Mach. Co.</i> , No. CIV.A. 09-1415, 2010 WL 3984822 (W.D. Pa. Oct. 9, 2010), <i>aff'd</i> , 497 F. App'x 970 (Fed. Cir. 2012).....	38

<i>K-2 Corp. v. Salomon S.A.</i> , 191 F.3d 1356 (Fed. Cir. 1999).....	9, 20
<i>Lincoln Nat. Life Ins. Co. v. Transamerica Life Ins. Co.</i> , 609 F.3d 1364 (Fed. Cir. 2010).....	37
<i>Merck & Co. v. Teva Pharms. USA, Inc.</i> , 395 F.3d 1364 (Fed. Cir. 2005).....	21
<i>Microsoft Corp. v. I4I Limited Partnership</i> , 564 U.S. 91 (2011).....	10
<i>Nautilus, Inc. v. Biosig Instruments, Inc.</i> , 572 U.S. 898 (2014).....	10
<i>Niazi Licensing Corp. v. St. Jude Med. S.C., Inc.</i> , 30 F.4th 1339 (Fed. Cir. 2022)	47
<i>Oatey Co. v. IPS Corp.</i> , No. 07-1214 (Fed. Cir. Jan. 30, 2008).....	29
<i>Omega Eng'g, Inc. v. Raytek Corp.</i> , 334 F.3d 1314 (Fed. Cir. 2003).....	22
<i>Omeprazole Patent Litigation</i> , 84 F. App'x 76 (Fed. Cir. 2003)	27
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005).....	9
<i>Sonix Tech. Co. v. Publications Int'l, Ltd.</i> , 844 F.3d 1370 (Fed. Cir. 2017).....	47
<i>Thorner v. Sony Computer Entm't Am. LLC</i> , 669 F.3d 1362 (Fed. Cir. 2012).....	9
<i>TQ Delta, LLC v. 2Wire, Inc.</i> , 373 F. Supp. 3d 509 (D. Del. 2019)	49
<i>U.S. Surgical Corp. v. Ethicon, Inc.</i> , 103 F.3d 1554 (Fed. Cir. 1997).....	9, 17
<i>Versa Corp. v. Ag-Bag Int'l Ltd.</i> , 392 F.3d 1325 (Fed. Cir. 2004).....	26
<i>York Prods., Inc. v. Cent. Tractor Farm & Family Ctr.</i> , 99 F.3d 1568 (Fed. Cir. 1996)	26

TABLE OF EXHIBITS

Exhibit No.	Description
1	U.S. Patent No. 9,632,748
2	U.S. Patent No. 9,812,128
3	U.S. Patent No. 11,024,311
4	U.S. Patent No. 11,050,615
5	<i>The Smart Audio Report</i> , NPR (2020)
6	Declaration of David V. Anderson
7	Fernández et al., <i>An Application of Recurrent Neural Networks to Discriminative Keyword Spotting</i> , ICANN 2007
8	U.S. Patent No. 9,318,107
9	Morgan and Bourlard, <i>Continuous Speech Recognition</i> , IEEE Signal Processing Magazine (May 1995)
10	Attachment A (Excerpts) to Sonos's December 7 Supplemental Responses to Google's First Interrogatories (Invalidity Contentions)
11	Keshet, <i>Automatic Speech and Speaker Recognition</i> (2009)
12	File History (Excerpts) of U.S. Patent No. 11,024,311
13	U.S. Patent App. Pub. No. 2022/0244905
14	U.S. Patent App. Pub. No. 2022/0224971

Pursuant to Order No. 6, Complainant Google LLC (“Complainant” or “Google”) respectfully submits this Initial Markman Brief for disputed terms in the claims of U.S. Patent Nos. 9,632,748 (“the ’748 patent”), 11,024,311 (“the ’311 patent”), 9,812,128 (“the ’128 patent”), and 11,050,615 (“the ’615 patent”) (collectively, the “Asserted Patents”) (Exhibits 1-4, respectively).

I. INTRODUCTION

Across the Asserted Patents, the meaning of all but two of the thirteen terms proposed by Respondent Sonos, Inc. (“Sonos”) is so plain that the terms do not require construction. (Two require simple corrections as the result of obvious drafting errors.) Sonos’s approach takes one of two forms: Sonos either contorts the claims with lengthy claim construction proposals that—when stripped down to their core—vitiate claim limitations, inject new limitations unsupported by the intrinsic evidence, or both; or, seizing on contrived inconsistencies or clerical errors that a person of ordinary skill in the art would easily have been able to understand, Sonos argues that the claims are indefinite.

For instance, with respect to the ’748 patent, Sonos’s proposal for “[determine/determining]...based at least in part on information...” omits the “based at least in part on” clause altogether, and adds a new negative limitation (“without any of the plurality of computing devices processing...”) that has no support in the intrinsic evidence. For the ’128 and ’311 patents, Sonos asserts that a POSITA would be unable to understand method claims reciting alternative conditional steps, when the specifications describe precisely the claimed methods using the exact same language. Similarly, for the ’615 patent, Sonos ignores the plain claim language and disclosures in the specification to assert that the term “short range signal transmission path” is indefinite.

For the reasons set forth herein, Google respectfully requests that the ALJ adopt its proposals, and decline to adopt Sonos’s constructions.

II. TECHNOLOGY BACKGROUND

In recent years, voice assistants—software that generally allows interaction with, and control of, electronic devices by audible phrases and commands—have exploded in popularity. Ex. 5 (*The Smart Audio Report*, NPR) at 5-6, 16. Google has been a pioneer in the development and advancement of voice-assistant technology since the early 2010s. Google’s innovations began with software that enabled voice-based search on Android smartphones and the Chrome web browser, and evolved into Google Assistant, a service that gives users a convenient interface for performing tasks on smart phones, speakers, watches, televisions, and more. Google Assistant, which is part of the domestic industry in this Investigation, has transformed smart device management, connected homes, and mobile computing.

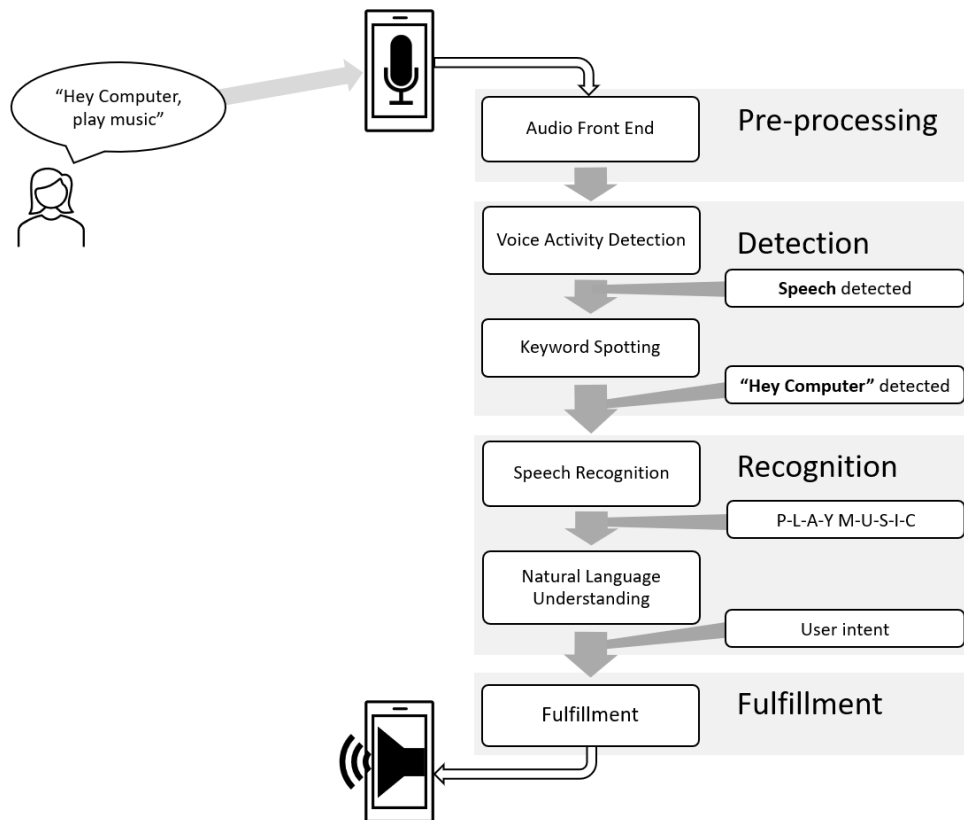
As voice assistant technology was implemented in a greater variety of devices, problems arose from the use of the technology in multiple devices in the same environment. For example, an open living space may have more than one device running a voice assistant application—e.g., a smart television and smart speaker, along with a smartphone and/or watch. As adoption of technology is and was often premised on its ease of use, and the number of smart devices (including those supporting voice assistant technology) began to increase, it became more important for new devices to be easily introduced (or “commissioned”) into an environment (e.g., a home network). It was also important that these newly-introduced devices (or multiple devices in a particular environment) did not interfere with one another in the execution of their respective voice assistant applications. If commissioning a new device onto a network was difficult, or if multiple devices’ voice assistant applications interfered with one another, the user’s experience would be adversely affected. The innovations of the Asserted Patents are designed to enhance these operations.

Below is a background summary of technologies relevant to the Asserted Patents and claim construction disputes.

A. Voice Assistant Technologies Relating to the '748, '128, and '311 Patents

Voice assistants and the smart devices that support them process audio in multiple stages—a “pipeline”—that generally increase in computational intensity and processor usage as the pipeline proceeds. A voice assistant pipeline typically starts with a device’s microphones receiving a user request and concludes with fulfillment of the request (i.e., execution of a requested action) by (a) the device that received it or (b) another device at receiving device’s direction. As shown below, the stages of the pipeline can be grouped together by function (with the understanding that not every “pipeline” in a given system necessarily includes every stage). Ex. 6 (Declaration of David V. Anderson (hereafter, “Anderson Decl.”)) at ¶¶ 36-37.

In general, the pipeline will include a pre-processing function (“audio front end”), detection functions for detecting the utterance of voice and/or a keyword (“voice activity detection” and “keyword spotting”), recognition functions for recognizing the words spoken by the user, translating the recognized words to text and processing the intent of those words (“speech recognition” and “natural language understanding”), and a fulfillment function for executing an action requested by a command that follows the keyword. Ex. 6 (Anderson Decl.) at ¶ 37.



Pre-processing (Audio Front-End): A typical first stage in the pipeline is an “**audio front-end**,” which pre-processes an audio signal received from the device’s microphones to clean the signal for subsequent stages by suppressing or removing background noise and/or audio feedback from the device’s speaker into the microphone, for example. Ex. 6 (Anderson Decl.) at ¶ 38.

Detection (Voice Activity Detection / Keyword Spotting): The next stages of the pipeline typically involve detection of human speech (i.e., voice) and/or specific keywords in the signal. Detection may be used as a gating mechanism for subsequent stages in the voice processing pipeline, such that subsequent processing only occurs when a keyword is detected. Ex. 6 (Anderson Decl.) at ¶ 39. As taught by the ’128 and ’311 patents, information obtained during the detection stage may also be used for “multi-device coordination,” so that when multiple devices

(e.g., in a room together) detect a voice input, the devices can automatically negotiate leadership to determine which device should respond to the detected input. '128 patent, 1:43-2:2.

“Voice activity detection” (VAD) is a process by which the signal from the audio front-end is checked for human speech. As a general matter, voice activity detectors continuously monitor for and analyze an incoming audio signal as it is received, and output a prediction regarding whether speech is present. Voice activity detectors operate specifically to detect speech presence; it is a binary decision. Ex. 6 (Anderson Decl.) at ¶ 40. For example, if a user says “Hey Computer, play music,” the voice activity detector would detect the presence of speech in the received audio signal.

“Keyword spotting,” as its name suggests, operates to detect particular keywords in an audio signal. Ex. 7 (Fernández et al., *An Application of Recurrent Neural Networks to Discriminative Keyword Spotting*, ICANN 2007) (“The goal of keyword spotting is to detect the presence of specific spoken words in (typically) unconstrained speech.”). A keyword may also be called a “hotword” or “wakeword,” examples of which include “Hey Google” or “Hey Sonos.” See, e.g., '128 patent, 11:44-67 (describing uses of hotwords and wakewords for voice assistant devices); see also Ex. 8 (U.S. Patent No. 9,318,107, incorporated by reference in the '128 and '311 patents), 1:47-2:3, 5:2-28 (describing uses of hotwords and methods for detecting hotwords). Keywords can be used to signify a user’s desire to activate voice assistant functionality. Modern approaches to keyword spotting rely on the use of neural networks, which are machine learning configurations patterned after the neurons of the human brain. Before use, the neural network is “trained” on the specific keywords of interest, after which it can predict whether an incoming audio signal contains a keyword based on characteristics of the signal. Ex. 6 (Anderson Decl.) at ¶¶ 41-42.

Keyword spotters operate specifically to detect whether one of a limited set of keywords is present in the signal based on characteristics of the audio signal; again, this is a binary decision. Ex. 6 (Anderson Decl.) at ¶ 43. For example, if a user says “Hey Computer, play music,” the keyword spotter might detect that the keyword (“Hey Computer”) is present, but it would not detect other words that are not keywords.

Recognition (Speech Recognition and Natural Language Understanding): After voice and/or a keyword is detected, the next step is to recognize the full set of spoken words (“speech recognition”) and what they mean within the context of the system (“natural language understanding”). In other words, the recognition function is directed to the more sophisticated process of figuring out what words have been spoken and translating those words into an intent associated with the execution of a function by a device. Ex. 6 (Anderson Decl.) at ¶ 44.

“**Speech recognition**,” also called “automatic speech recognition” or “ASR,” involves the recognition of words in the signal and conversion of these words to text (which can then be used to process commands). Ex. 6 (Anderson Decl.) at ¶ 45. Ex. 9 (Morgan and Bourlard, *Continuous Speech Recognition*, IEEE Signal Processing Magazine (May 1995) (“The basic task of *Automatic Speech Recognition* (ASR) is to derive a sequence of words from a stream of acoustic information.”)). Speech recognition employs an acoustic model to evaluate a received audio signal against thousands of known sequences of “phonemes,” or units of sound that distinguish one word from another, to determine the best candidate textual words for each spoken word. Ex. 6 (Anderson Decl.) at ¶ 45. The speech recognition system then uses a language model to find the most likely word sequence from these candidate words. The language model evaluates the candidate words based on the likelihood that they will appear together in a given context—for example, it considers the likelihood that the candidate words “plan to wait” will appear together,

as opposed to the candidate words “plant weight.” Ex. 6 (Anderson Decl.) at ¶ 45. Following the example described above, if a user says “play music” after the keyword, the speech recognizer would transcribe that utterance into text for further processing by subsequent stages in the voice assistant pipeline.

As compared with the detection technologies describe above (voice activity detection and keyword spotting), speech recognition is computationally expensive, requiring more memory and processing resources. Ex. 6 (Anderson Decl.) at ¶ 46. For this reason, a voice assistant pipeline may be designed such that speech recognition does not begin until a keyword has been detected.

“**Natural language understanding**” involves analysis of the recognized text corresponding to the user’s utterance to determine what the user wants the voice assistant to do, typically referred to as an “intent” (e.g., a request to play music). Ex. 6 (Anderson Decl.) at ¶ 47.

Fulfillment: The final stage in the voice assistant pipeline is fulfillment. At this stage, an action is taken based on the determined user intent, and potentially based on additional information collected (for example, by the voice assistant asking the user for clarification). This stage also may involve a determination by the voice assistant as to which device(s) should fulfill the user’s request. For example, as taught by the ’128 and ’311 patents, a user might request playback of content on a different device from the device that handles the request initially. ’128 patent, 9:38-43. The user interacts with and receives responses—e.g., requests for clarification from the user or confirmation that the requested action (music playback) will be carried out—from the first device, and the user’s request to play music is then fulfilled by the other device playing the requested content. ’128 patent, 10:2-20, 24:61-25:7. Ex. 6 (Anderson Decl.) at ¶¶ 48-49.

B. Device Commissioning Technologies Relating to the ’615 Patent

A communication network, like a Wireless Local Area Network (“WLAN”), links devices together and can connect devices to the Internet. Before a new device can join an existing

communication network it must be “commissioned” (i.e., added) to that network. The commissioning process for many networks requires any new device to have certain information before that device can join the network, including (1) the appropriate network identifier (for example, a service set identifier or “SSID”) for the network, and (2) the password for that network. A variety of mechanisms can be used to supply configuration information to a new device and commission that device to a network. The novel approach in the ‘615 patent uses a smart wireless device that is already on the communication network to commission a new device to that network. The smart wireless device, such as a smartphone or tablet, establishes a separate short-range connection with the new device and uses that short-range path to provide configuration information to the new device. This permits the new device to join the communication network without disrupting the smart wireless device’s existing connection to that same network.

III. LEVEL OF ORDINARY SKILL IN THE ART

The levels of ordinary skill at the time of the respective inventions of the Asserted Patents are set forth below:¹

’748, ’128, and ’311 Patents: The parties agree that the appropriate level of skill for the ’748, ’128, and ’311 patents is “[t]he equivalent of a four-year degree from an accredited institution (typically denoted as a B.S. degree) in electrical or computer engineering, or a related field, and approximately 2-4 years of experience in the field of voice and/or audio processing, or an equivalent level of skill, knowledge, and experience.” Ex. 10 (Sonos Invalidity Contentions) at 2, 4.

¹ The relevant dates of invention of the asserted patent claims are as follows: April 11, 2012 (’615 patent, all claims); July 29, 2014 (’748 patent, all claims); October 9, 2014 (’128 patent, claims 1-3, 6-8, and 11-13, and ’311 patent, claims 1-3, 8-10, 12, 14-18, and 20); April 1, 2015 (’128 patent, claims 4, 9, and 14, and ’311 patent, claim 19); and October 3, 2016 (’128 patent, claims 5, 10, and 15, and ’311 patent, claim 11).

'615 Patent: The parties agree that the appropriate level of skill for the '615 patent is “at least (a) a Bachelor’s degree in computer science, computer engineering, electrical engineering, or an equivalent thereof, and (b) 2-4 years of professional experience in the fields of networking and network-based systems or applications or an equivalent level of skill, knowledge, and experience.” *Id.*, 69.

IV. LEGAL STANDARDS

A. Claim Construction

The words of a patent claim are generally given their “ordinary and customary meaning” which is “the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention” when read and understood in the context of the claims, the entire specification, and the prosecution history. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005). “There are only two exceptions to this general rule: 1) when a patentee sets out a definition and acts as his own lexicographer, or 2) when the patentee disavows the full scope of a claim term either in the specification or during prosecution.” *Thorner v. Sony Computer Entm’t Am. LLC*, 669 F.3d 1362, 1365 (Fed. Cir. 2012). When a claim term is “clear on its face,” the claim construction process is limited to “determining if a deviation from the clear language of the claims is specified” under the rule articulated in *Thorner*. *K-2 Corp. v. Salomon S.A.*, 191 F.3d 1356, 1363 (Fed. Cir. 1999); *Interactive Gift Express, Inc. v. Compuserve, Inc.*, 256 F.3d 1323, 1331-32 (Fed. Cir. 2001); *see also U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997) (“Claim construction . . . is not an obligatory exercise in redundancy.”).

Extrinsic evidence may also “assist the court in determining the meaning of particular terminology to those of skill in the art of the invention.” *Phillips*, 415 F.3d at 1318. But extrinsic evidence should not “contradict the meaning otherwise apparent from the intrinsic record.” *Helmsderfer v. Bobrick Washroom Equip., Inc.*, 527 F.3d 1379, 1382 (Fed. Cir. 2008).

B. Indefiniteness

Invalidity of a patent claim must be proven by clear and convincing evidence. *Microsoft Corp. v. I4I Limited Partnership*, 564 U.S. 91, 95 (2011). A patent claim is not indefinite if the claim, read in light of the specification and the prosecution history, “inform[s], with reasonable certainty, those skilled in the art about the scope of the invention.” *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 901 (2014). *Nautilus* “mandates clarity, while recognizing that absolute precision is unattainable.” *Id.* at 910. It also recognizes that “[s]ome modicum of uncertainty . . . is the ‘price of ensuring the appropriate incentives for innovation.’” *Id.* at 909 (quoting *Festo Corp. v. Shoketsu Kinzoku Kogyokabushiki Co.*, 535 U.S. 722, 732 (2002)).

C. Correcting Obvious Errors

The Commission applies the same standard as district courts to determine whether an obvious claim-drafting error should be corrected during claim construction. Specifically, obvious errors are correctable where “(1) correction is not subject to reasonable debate based on consideration of the claim language and the specification *and* (2) the prosecution history does not suggest a different interpretation of the claims.” *Certain Multiple Mode Outdoor Grills and Parts Thereof*, Inv. No. 337-TA-895, Comm’n Op., 2015 WL 13817118 at *16 (Feb. 3, 2015) (citation omitted) (emphasis in original). As the Commission has explained, “absent evidence of culpability or intent to deceive, a patent claim should not be invalidated based on an obvious clerical error.” *Id.* (citing *Hoffer v. Microsoft Corp.*, 405 F.3d 1326, 1331 (Fed. Cir. 2005)).

V. ’748 PATENT

A. Overview of the ’748 Patent

The ’748 patent claims systems and methods for designating a device from a plurality of devices to perform speech recognition. *See generally, e.g.*, ’748 patent, 2:36-39. This designation prevents parallel-processing issues that may otherwise arise from multiple devices attempting to

simultaneously “create, access, modify, or delete the same data.” *Id.*, 3:7-13, 5:37-56. This designation is made in part based on information received from the various devices, such as a configured hierarchy based on type of device (7:37-64), audio input quality of each device’s environment (7:65-8:13), the number of predetermined audio commands each device is capable of handling (8:14-25), the “context” of each device such as location, the time, applications installed on or executing on the device, or other contextual information (8:26-52); or other information related to the user’s identity (8:53-62). Upon determining it is the device best suited to respond to an audio input, the designated device may send “an indication of the designation” to the other devices, and the other devices may “cease processing of audio input.” The designated device is then able to perform any necessary follow-up and fulfill the user’s request (or instruct another device to do so, if appropriate). *Id.*, 18:17-20, 25:43-57.

Once a device has been designated through this process, it may “utilize speech recognition techniques to determine text from obtained audio input, and compare the text to a set of predetermined audio commands ... to determine whether the obtained audio input includes a predetermined audio command.” *Id.*, 4:57-63. Predetermined audio commands are spoken inputs that are “associate[d] ... with an action to be performed by the respective computing device,” such as “make a call,” “what’s the time,” “give me directions,” or “send email.” *Id.*, 4:8-21, 9:61-62, 20:16-17, 24:28-30. If a predetermined audio command is detected, the device may then request any necessary additional information and/or execute the action associated with the command. *Id.*, 24:31-51.

B. Agreed Constructions

Claim Term	Agreed Construction
“the sec-end first computing device” (Claim 3)	“the first computing device”

C. “[perform/performing] speech recognition” (All Claims)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning), i.e., “transforming speech to corresponding text”	“processing spoken audio to determine whether it includes a predetermined spoken audio command”

The term “speech recognition” has a well-understood meaning in the art, it refers to a process for transforming speech to corresponding text, as reflected in Google’s proposal. Sonos’s proposal, on the other hand is overly broad and would effectively write the words “speech recognition” out of the claims.

The term “speech recognition” is well-understood in the art to refer to a process for transforming speech to corresponding text. Ex. 6 (Anderson Decl.) at ¶ 52. Nothing in the specification or prosecution history indicates that the term should have anything other than its plain and ordinary meaning. Sonos’s proposal is unclear and effectively excludes the plain and ordinary meaning of the term. It would improperly sweep detection techniques (voice activity detection and keyword (hotword) detection) within the term’s scope (*see generally supra* Section II.A), and is redundant of other elements recited in the claims.² Indeed, Sonos’s construction is even broader than that, seeking to ensnare **any** processing of “spoken audio to determine whether it includes a predetermined spoken audio command.” Ex. 6 (Anderson Decl.) at ¶ 54, 58. Thus, Sonos’s construction, which is overly broad in certain ways and unduly narrow in others, should be rejected.

The ’748 patent uses the term “speech recognition” consistent with its ordinary meaning of deriving text from speech. The ’748 patent describes the claimed computing devices’ “audio input modules may be operable to utilize ***speech recognition techniques to determine text from***

² For example, “determining, by the first computing device, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a predetermined spoken audio command.” ’748 patent, cl. 1.

obtained audio input.” ’748 patent, 4:57-59 (emphasis added). A POSITA would understand from this disclosure that the term “speech recognition,” as used in the ’748 patent has its plain and ordinary meaning. Ex. 6 (Anderson Decl.) at ¶ 53; *supra* Section II.A. As reflected in literature disclosing the state of the art at the time of the ’748 patent, Google’s proposal (and the description in the ’748 patent) is the well-accepted meaning of “speech recognition.” *See, e.g.*, Ex. 6 (Anderson Decl.) at ¶ 52; Ex. 9 (Nelson Morgan & Hervé Bourlard, *Continuous Speech Recognition*, IEEE Signal Processing Magazine (May 1995)) at 3-4 (“The basic task of Automatic Speech Recognition (ASR) is to derive a sequence of words from a stream of acoustic information”); Ex. 11 (Keshet, *Automatic Speech and Speaker Recognition* (2009)) at 3 (the goal of “automatic speech recognition” is “to transcribe a recorded speech utterance into its corresponding sequence of words.”).

Sonos’s proposed construction is demonstrably incorrect. First, Sonos seeks to broaden “speech recognition” to include any kind of “processing” (without explaining what this term means), including keyword or hotword detection. However, a POSITA would recognize that speech recognition (transforming speech to text) differs from keyword or hotword detection, which have different uses and implications. Ex. 6 (Anderson Decl.) at ¶¶ 54-55; *see, e.g.*, Ex. 11 (Keshet, *Automatic Speech and Speaker Recognition* (2009)) at 3 (the goal of “automatic speech recognition” is “to transcribe a recorded speech utterance into its corresponding sequence of words.”), 13 (“The core objective of a keyword spotting system is to discriminate between utterances in which a given keyword is uttered to utterances in which the keyword is not uttered.”); *see also id.* at 12 (describing limitations of keyword spotting: “Most of these limitations are **not specific to the keyword spotting problem**, and are common to **other tasks such as speech recognition**, as pointed out in Chapter 1.”); *supra* Section II.A. The distinction between “speech

recognition” and keyword or hotword detection is reflected in the specification. As noted above, the specification is unequivocal in its description of speech recognition techniques “**to determine text from obtained audio input.**” ’748 patent, 4:57-59. The **only** time the ’748 patent uses the term “hotword” is in the context of monitoring for a **predetermined audio command**, not speech recognition. See ’748 patent at 2:66-67.

Second, and similarly, Sonos’s construction conflates two separate processes performed by the claimed computing devices, only one of which is “performing speech recognition.” The ’748 patent explains that the computing devices may “process the audio input by performing one or more speech-to-text transformations corresponding to the received audio input” (i.e., speech recognition) or, “[i]n other examples,” by performing “audio analysis to recognize and identify speech, such as performing frequency analysis to identify tonal characteristics or other sonic identifiers present in the audio input.” ’748 patent, 15:28-38. A POSITA reading the ’748 patent would thus understand the latter to refer to processes separate from speech recognition within the context of the patent. Specifically, a POSITA would understand “performing frequency analysis to identify tonal characteristics or other sonic identifiers present in the audio input” to refer to voice activity detection (i.e., whether the audio received by the device was a voice) or keyword detection, but not speech recognition. Ex. 6 (Anderson Decl.) at ¶¶ 56-57.

Sonos’s inclusion of “determin[ing] whether it includes a predetermined spoken audio command” thus cannot be correct.³ For the foregoing reasons, “[perform/performing] speech

³ Sonos’s proposed construction is also at odds with, and undermined by, its own invalidity contentions in this Investigation. Sonos contends that “performing speech recognition” refers to “processing spoken audio,” but simultaneously argues in its contentions that “speech recognition” is not sufficiently described in the ’748 patent. Ex. 10 (Sonos Invalidity Contentions) at 3 (“[T]he specification lacks enablement and written description support for this recited sequence. Indeed, the specification only mentions the term ‘speech recognition’ twice in a more general sense. See, e.g., *id.*, 4:57-63, 10:21-25. There is no disclosure of a first computing device ‘determin[ing] whether [it] should perform speech recognition on the spoken

recognition” should be given its ordinary meaning, and Sonos’s overly broad proposal should be rejected.

**D. “a command to temporarily refrain from performing speech recognition”
(All Claims)**

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“a message instructing the second computing device to temporarily cease performing any and all speech recognition”

This term should be accorded its plain and ordinary meaning, consistent with the intrinsic record. Sonos’s proposed construction substitutes some words of the claim and adds others (i.e., “any and all”). Sonos’s changes do not, on their face, appear to alter the ordinary meaning of the disputed claim phrase—e.g., substituting “message instructing” for “command,” and “cease” for “refrain.” But the original claim language itself is clear, and Sonos’s substitutions are therefore unnecessary and disfavored, particularly if Sonos advocates an interpretation of its construction that differs from the ordinary meaning articulated by Google below. Sonos’s other addition—“any and all”—merely adds words that lack support in the intrinsic evidence (through lexicography or disclaimer) and should also be rejected.

The term “a command to temporarily refrain from performing speech recognition” appears in each of the independent claims of the ’748 patent. For example, claim 1 recites “sending, by the first computing device and to at least a second computing device from the plurality of computing devices, *a command to temporarily refrain from performing speech recognition.*” ’748 patent, cl. 1 (emphasis added). Claims 7 and 11 include similar limitations. The claim language is clear on its face. It uses common English words and employs them using their standard

audio input,’ much less performing additional functions (i.e., send, perform, and determine) based on such a determination.”).

meanings. Each independent claim requires, “responsive to determining that the first computing device should perform speech recognition on the spoken audio input,” “send[ing], [by the first computing device and] to at least a second computing device from a plurality of computing devices, a command to temporarily refrain from performing speech recognition.” ’748 patent, cls. 1, 7, 11. The plain meaning of this claim language is directed to the operation of the claimed first computing device, which sends a message to at least the second computing device that at least the second computing device should not perform speech recognition for a limited amount of time.

The specification confirms the claim language’s ordinary meaning to a POSITA. Ex. 6 (Anderson Decl.) at ¶¶ 61-62. The ’748 patent explains that, “[i]n general, techniques of the present disclosure may enable one or more computing devices from among a plurality of computing devices to designate a particular computing device to monitor for and/or act on audio input” ’748 patent, 2:36-41. Once that device is designated, the other devices may “temporarily cease to monitor for audio input.” *Id.*, 2:52-57. One way this can be accomplished is for the designated computing device to send instructions (a command) to the other, non-designated computing devices informing them that they should temporarily cease monitoring for audio input. *Id.*, 9:36-41 (“In some examples, the designated particular computing device may output instructions to computing devices other than the particular computing device to cause the respective computing devices other than the particular computing device to temporarily deactivate respective audio input modules.”).

A POSITA would have no difficulty understanding the meaning of the disputed claim language in view of these specification disclosures. Ex. 6 (Anderson Decl.) at ¶¶ 62-63.

Sonos’s construction changes some words in the claim and adds others without support in the claims or intrinsic record, without altering the meaning of the disputed phrase in any apparent

way. Specifically, Sonos changes “command” to “message instructing,” “refrain” to “cease,” and “speech recognition” to “any and all speech recognition.” It is not clear how, if at all, Sonos intends those changes to affect the claim scope, or how Sonos’s changes could provide a POSITA with any added clarity as to the meaning of the claim language. Claim construction is “not an obligatory exercise in redundancy”; there is no reason to construe claim language that is clear on its face, particularly in a way that does not add any clarity or affect claim scope. *U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997).

For these reasons, Google’s proposal of plain and ordinary meaning should be adopted.

E. “predetermined spoken audio command” (All Claims)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“any audio input usable to initiate interaction with a computing device, such as a spoken hotword”

“Predetermined spoken audio command” requires no construction. It has a plain and ordinary meaning to a POSITA in light of the specification. A predetermined spoken audio command is a spoken utterance that includes a directive for the claimed computing device to execute an associated action. Ex. 6 (Anderson Decl.) at ¶ 66. Sonos’s construction for this phrase is directed to “initiat[ing] interaction” with a computing device, and would therefore vitiate the requirement that the predetermined spoken audio command actually include a **command**, i.e., an instruction to execute an action, as recited in the asserted claims. Only Google’s proposal is consistent with the intrinsic evidence.

The claims require the “predetermined spoken audio command” to include a directive that can be executed by the device. Ex. 6 (Anderson Decl.) at ¶¶ 67-68. Each of independent claims 1, 7, and 10 recites “determin[e/ing], by the first computing device, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a **predetermined**

spoken audio command; and responsive to determining that the spoken audio input includes the predetermined spoken audio command, ***executing***[, by the first computing device], ***an action associated with the predetermined spoken audio command.***” ’748 patent, cls. 1, 7, 10.

The specification consistently and repeatedly associates predetermined audio commands with corresponding actions. *See, e.g.*, ’748 patent, 1:15-18 (“In some cases, a computing device may monitor an environment for audio input that includes a predetermined audio command and, if the predetermined audio command is detected, perform a corresponding action.”), 4:8-10 (“one or more of audio input modules 6 may associate a predetermined audio command with an action to be performed by the respective computing device”).

Indeed, the specification is replete with specific examples of predetermined spoken audio commands and their associated actions. The ’748 patent explains that “[o]ne or more of audio input modules 6 may associate a predetermined audio command with an action to be performed by the respective computing device.” ’748 patent, 4:8-10. For example, predetermined audio commands can include “answer” (to answer a telephone call), “navigation” or “directions” (to request directions), “send email,” “send text message,” and other directive phrases. *Id.*, 4:14-21. Rather than simply “directions,” the phrase “[g]ive me directions” may also be a predetermined spoken audio command. *Id.*, 9:61-62 (“The user may speak a predetermined audio command (e.g., ‘Give me directions’).”). In another example, “the audio input includes the predetermined audio command ‘what’s the time.’” *Id.*, 20:15-18. “Go silent” is another example. *Id.*, 24:36-39. The specification also provides an example where “the predetermined audio command may be the word ‘play’ and the subsequent audio input may be the title of a song, TV show, or other media content item.” *Id.*, 24:41-46.

Sonos seeks to read the word “command” out of the claims entirely. By defining “predetermined spoken audio command” to cover “**any** audio input usable to **initiate interaction** with a computing device, such as a spoken hotword,” Sonos effectively replaces a “command” in the claims with a keyword (e.g., “spokenhotword”). This cannot be correct, at least because many “commands” expand beyond the bounds of a hotword. As discussed above, the specification discloses predetermined audio commands associated with a variety of actions to be carried out by the device beyond just entry into an audio input mode (as would be the case with a hotword), including answering telephone calls and sending messages. *See, e.g.,* ’748 patent, 4:8-30. Sonos’s proposed construction would exclude these disclosed “actions,” thereby unduly restricting and otherwise changing the meaning of the claim term, and should be rejected.

Sonos’s construction is unduly narrow and would exclude clear “commands,” as described above. Excluding such embodiments is disfavored and, for the reasons described above, the plain and ordinary meaning of “predetermined spoken audio command” should govern.

F. “[determine/determining] ... based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input” (All Claims)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“determining whether the first computing device should perform speech recognition on the spoken audio input without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command”

This claim term, as written, is clear. The term appears in each of the independent claims of the ’748 patent. For example, claim 1 recites: “determining, by the first computing device and based at least in part on information received from the one or more additional computing devices,

whether the first computing device should perform speech recognition on the spoken audio input.” ’748 patent, cl. 1. Claims 7 and 11 include similar language. ’748 patent, cls. 7, 11.

The parties do not dispute this portion of the claim phrase: “determining . . . whether the first computing device should perform speech recognition on the spoken audio input.” The dispute centers on Sonos’s wholesale replacement of the claim language “based at least in part on information received from the one or more additional computing devices” with its proposed construction of “without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command” (i.e., “speech recognition,” as it has construed that term). Sonos’s construction would alter the plain meaning of the claim in two material ways: (1) it would broaden the claim in one respect by eliminating the requirement that the determination of “whether the first computing device should perform speech recognition” be “based at least in part on information received from the one or more additional computing devices”; and (2) it would narrow the claim in another respect by adding a requirement that this determination be done without any other device of the plurality of computing devices performing additional “processing” (i.e., “speech recognition”).

Because the term is clear on its face, Sonos must demonstrate that the patentee either intended this limitation to have a special meaning or disavowed claim scope. *K-2 Corp.*, 191 F.3d at 1363. Neither occurred here. The specification explains that “one or more computing devices from a plurality of computing devices may designate a particular computing device from among the plurality of computing devices to process audio input to determine whether the audio input includes a predetermined audio command.” ’748 patent, 6:61-66. The computing devices may determine which of them should be designated as the “particular computing device” using an arbitration process, or each can make the decision individually. *Id.*, 7:30-36.

The determination can be “based at least in part on” a variety of criteria, such as a configured hierarchy based on type of device (7:37-64), audio input quality of each device’s environment (7:65-8:13), the number of predetermined audio commands each device is capable of handling (8:14-25), the “context” of each device such as location, the time, applications installed on or executing on the device, or other contextual information (8:26-52); or other information (8:53-62). The claim language (“based at least in part on information received from the one or more additional computing devices”) is consistent with these examples, and the plain meaning of the claim language would be readily understood by a POSITA.⁴ Ex. 6 (Anderson Decl.) at ¶ 72.

Sonos’s construction impermissibly broadens the scope of the claims by removing the requirement that the “determining” step must be “based at least in part on information received from the one or more additional computing devices.” It is well-established that “each element contained in a patent claim is deemed material to defining the scope of the patented invention.” *Bicon, Inc. v. Straumann Co.*, 441 F.3d 945, 950 (Fed. Cir. 2006); *see also Merck & Co. v. Teva Pharms. USA, Inc.*, 395 F.3d 1364, 1372 (Fed. Cir. 2005) (“a claim construction that gives meaning to all the terms of the claim is preferred over one that does not do so”).

Sonos’s construction also narrows the scope of the claims by adding a negative limitation: that the “determining” step be performed “without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command.” This aspect of Sonos’s construction—which would exclude from the claim scope a system in which the component devices process at least some spoken audio input even if they are

⁴ Whereas the above examples describe designating a device “to process audio input,” the claims recite determining “whether the first computing device should perform speech recognition.” As described above in Section V.C, speech recognition is a form of audio processing.

not the designated device⁵—finds no support in the intrinsic record. Moreover, negative limitations must find support either in “the words of the claim” or through an “express disclaimer or independent lexicography in the written description that would justify adding that negative limitation.” *Omega Eng ’g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1323 (Fed. Cir. 2003). There is no such support here.

Sonos’s construction would require the claims’ determination “whether the first computing device should perform speech recognition on the spoken audio input” to be performed without any other device performing “speech recognition” (as defined by Sonos). However, nothing in the intrinsic evidence precludes the non-designated computing devices from beginning speech recognition before a first computing device determines that it should do so; the patentee left those implementation details open. There is nothing precluding such operation in the specification, which explicitly permits all computing devices to monitor the environment and perform some processing on the received audio input. For example, where the designation decision is based on “a level of audio input quality associated with one or more of computing devices 2,” the ’748 patent explains that “one or more of computing devices 2 may obtain audio input from the environment . . . and communicate with others of computing devices 2 to compare the quality of the audio input and/or of speech included therein.” *Id.*, 8:2-7. That communication would require some processing, but there is no reason that processing could not include determining whether the audio input includes a predetermined audio command. Ex. 6 (Anderson Decl.) at ¶ 74. Nor does the specification foreclose speech recognition from initiating at other devices before they receive an instruction from the first computing device to refrain from doing so after the first computing

⁵ This restriction is also incorrect for the reasons addressed above with respect to the term “[perform/performing] speech recognition.” See *supra* Section V.C.

device determines that it will perform speech recognition and, correspondingly, execute an action associated with the predetermined spoken audio command. The claims (consistent with the specification) are focused on the operation of the first electronic device in deciding to perform speech recognitions, and the steps the first electronic device takes (including instructions issued to other devices) in response to making that determination.

Further, even when the non-designated devices are instructed not to perform speech recognition, the focus of the claim is on the issuance of the instruction and not on how exactly and when any device that receives the instruction operates in response. As discussed above with respect to the term “a command to temporarily refrain” in Section V.D, in some embodiments the designated device may send a command to the other devices to refrain from performing speech recognition on the spoken audio input. When the designated device sends an instruction to the non-designated devices to “temporarily deactivate respective audio input modules,” the ’748 patent does not specify that the deactivation must be immediate or that it must occur before any of the spoken audio input is processed. *Id.*, 9:36-50. Therefore, there is no basis to construe the claims to require that the determination whether to perform speech recognition must be made “without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command.”

For the foregoing reasons, this term should also be accorded its plain and ordinary meaning, and Sonos’s construction should be rejected.

VI. ’128 AND ’311 PATENTS

A. Overview of the ’128 and ’311 Patents

The ’128 and ’311 patents are related and share a common specification. See ’311 patent, p. 2 (Related U.S. Application Data). The patents describe devices with “voice assistants” that allow users to “implement voice-activated functions” such as “initiate a phone call, conduct a

restaurant search, start routing on a map, create calendar events, add a post to a social network, recognize a song, and complete many other tasks.” ’128 patent, 1:34-37. As the patents explain, a location (“e.g., a room or space within a home”) may include multiple such devices, and it is “desirable for there to be a leader amongst the voice assistant devices that is responsible for responding to the user’s voice inputs, in order to reduce user confusion.” *Id.*, 1:43-53.

To achieve these goals, the patents describe methods and systems for “multi-device coordination” to “negotiate leadership amongst multiple devices for response to a speaker.” *Id.*, 1:57-2:2. This coordination can be performed by comparing values associated with detection of the voice inputs at the respective devices, which may be affected by multiple factors, including “surrounding noise, capabilities of the device, distance and position of the device relative to the user, position of the device within the space, and so on.” *Id.*, Figs. 4A-4B and 6, 20:45-67, 28:60-29:8. The patents describe that each device determines “a level of quality” of the detected voice input, which “may be expressed as a score or some other value,” and communicates that value or score to other devices. *Id.*, 21:1-4, 21:14-22, 29:9-31. The values or scores of the respective devices are then compared, and based on that comparison, a determination regarding whether a device should respond to the user’s voice input or forgo responding is made. *Id.*, 21:58-22:1, 29:32-30:12.

The above-described comparison based on values or scores determined based on detection of a voice input is reflected, for example, in claim 1 of the ’128 patent, which recites “detecting a voice input” and “determining a quality score for the detected voice input.” ’128 patent, cl. 1. The patents also describe how devices might use additional information gained from speech recognition to make a separate determination whether to respond based on the content of the user’s request. For example, claim 5 of the ’128 patent recites “recognizing a command in the voice

input” and responding if “the type of the command is related to the first electronic device.” ’128 patent, cl. 5.

A device that responds to a detected voice input outputs audible or visual responses to the user, for example to request additional information or to confirm that the user’s request has been fulfilled. *Id.*, 10:2-20, 24:61-25:7. However, a user’s request might be fulfilled by a different device from the device that interacts with the user; in other words, a device could fulfill the user’s request even if it forgoes responding to the detected voice input based on the above-described comparison of values or scores. *Id.*, 9:38-43, 24:45-51.

B. “other devices of the plurality of electronic devices” / “quality scores generated by the other devices of the plurality of electronic devices” (’128 All Claims)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“other devices” means “two or more other devices”; and “quality scores” means “two or more quality scores”

The independent claims of the ’128 patent recite “a first electronic device of a plurality of electronic devices.” Sonos has not identified the “plurality” term for construction, and there is no dispute that the ordinary meaning of “plurality” is “at least two.” Instead, the parties’ dispute centers on the “other devices”/“quality scores” terms, which appear in the following limitations:

communicating the quality score to the ***other devices of the plurality of electronic devices***;

receiving ***quality scores generated by the other devices of the plurality of electronic devices*** for detection of the voice input by the other devices;

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, wherein the ***other devices of the plurality of electronic devices*** forgo outputting an audible response to the detected voice input; and

’128 patent, claims 1, 6, 11 (emphases added).

Specifically, Sonos contends that the terms “other devices” and “quality scores” require two or more devices and two or more scores (thereby excluding a system having only two devices—a first electronic device and an “other device”). This is incorrect. The meaning of these terms is clear in the context of the claims and the specification. “Other devices” may refer to one or more “other devices;” it does not require two or more devices. This principle also applies to “quality scores,” which may refer to a single “quality score” (for example, when there is only one “other device”) or multiple “quality scores” (when there are multiple “other devices”).

The ordinary meaning of the term “plurality,” recited in the asserted claims of the ’128 patent, “simply means two or more.” *Cheese Sys., Inc. v. Tetra Pak Cheese & Powder Sys., Inc.*, 725 F.3d 1341, 1348 (Fed. Cir. 2013); *see also York Prods., Inc. v. Cent. Tractor Farm & Family Ctr.*, 99 F.3d 1568, 1575 (Fed. Cir. 1996). By using this term, the asserted claims are directed to a system of “two or more” electronic devices. A POSITA would understand that, with the claimed plurality of electronic devices including the “first electronic device,” there need be only one “other device” and one “quality score” received from that “other device” to constitute a “plurality.” Ex. 6 (Anderson Decl.) at ¶ 79. The Federal Circuit has repeatedly cautioned that plural terms (and whether those terms can cover a single device) should be construed in light of the context in which they appear. In the ’128 patent, the claims use the plural form of “other devices” and “quality scores” to maintain grammatical consistency with the “plurality of other devices.” Given that context, and the patent’s disclosure of a “plurality of electronic devices” and a “first electronic device,” the recited “other devices” and “quality scores” should be construed to encompass a configuration with one other device and its associated quality score (e.g., a two-device “plurality of electronic devices”). *See Versa Corp. v. Ag-Bag Int’l Ltd.*, 392 F.3d 1325, 1330 (Fed. Cir. 2004) (holding “means for creating air channels” did not require a supporting structure that made

multiple channels); *Omeprazole Patent Litigation*, 84 F. App'x 76, 80 (Fed. Cir. 2003) (non-precedential) (claims reciting “one or more layers of materials” did not require two or more materials because “materials” reflects an attempt to achieve grammatical consistency with the tense of “layers”).

Indeed, the Federal Circuit’s decision in *Dayco Prods., Inc. v. Total Containment, Inc.*, which addressed claims reciting a “plurality” and plural claim elements within that plurality, is on all fours with the facts here. 258 F.3d 1317, 1328 (Fed. Cir. 2001). In *Dayco*, the Federal Circuit found that the district court erred in construing “a **plurality** of outwardly directed projections with **recesses** there between” to require at least three projections because the recitation of “recesses” required at least two recesses, and thus implicitly three projections. *Id.* at 1325, 1327-28 (emphases added). As the court explained, the term “plurality” “refers to two or more items, absent some indication to the contrary,” and the term “recesses” in the context of the claim meant either a single recess or a plurality of recesses. *Id.* at 1327-28;⁶ *see also Certain Elec. Devices, Including Handheld Wireless Commc’ns Devices*, Initial Determination, Inv. No. 337-TA-667 (Sept. 21, 2009) (in claims reciting a “plurality of processor devices” including an “owner processor device” and “other processor devices,” finding that “other processor devices” may refer to a single processor device when the “plurality of processor devices” includes only two processors).

Like the claim at issue in *Dayco*, the ’128 claims recite a plurality of electronic devices including a “first electronic device” and “other devices” (with corresponding “quality scores”), and the “other devices” and “quality scores” can refer to a single other device. If the ’128 patent inventors had wanted to ensure a minimum of three devices (and quality scores) in a system—i.e.,

⁶ The Federal Circuit noted that, had the patentees wanted to claim two or more recesses, they could have recited “a plurality of recesses” in a similar manner to its recitation of “a plurality of ... projections.” *Id.* at 1328.

a first electronic device and at least two “other devices”—they could have claimed a “plurality” of other devices and a “plurality” of quality scores. They did not do so.

The ’128 specification confirms that the claimed “other devices” or “quality scores” can refer to a single device or score in a two-device system:

FIG. 6 is a flow diagram illustrating a method 1200 of negotiating device leadership among multiple voice interface devices, in accordance with some implementations. The method 1200 is implemented at an electronic device (e.g., device 1002-1, 1002-2, or 1002-3) of **two or more electronic devices** 190 (electronic devices 1002), where each of the **two or more electronic devices** includes one or more microphones, a speaker, one or more processors and memory storing one or more programs for execution by the one or more processors.

’128 patent, Fig. 6 (tracking elements of claim 1), 28:63-29:5 (emphases added). Here, “an electronic device” would refer to the claimed “first electronic device,” and therefore “two or more electronic devices” would contemplate a single “other device.” Google’s proposed construction ensures that the Figure 6 embodiment is properly included within the claims’ scope, as there is no evidence that the inventors desired to exclude this embodiment. *See GE Lighting Sols., LLC v. AgiLight, Inc.*, 750 F.3d 1304, 1311 (Fed. Cir. 2014) (“where claims can reasonably [be] interpreted to include a specific embodiment, it is incorrect to construe the claims to exclude that embodiment, absent probative evidence on the contrary”); *Oatey Co. v. IPS Corp.*, No. 07-1214 (Fed. Cir. Jan. 30, 2008) (similar).

For these reasons, Google’s construction should be adopted, and Sonos’s attempt to limit the claim scope should be rejected.

C. “second values ... from other electronic devices” / “second values from other electronic devices of the plurality of electronic devices” (’311 All Claims)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“second values” means “two or more second values”; and “electronic devices” means “two or more electronic devices”

As with the '128 claims addressed in Section VI.B, *supra*, independent claims 1 and 10 of the '311 patent recite “a first electronic device of a plurality of electronic devices.” Independent claim 16 of the '311 patent recites “a plurality of electronic devices.” Again, Sonos has not identified the “plurality” term for construction, nor contested its ordinary meaning of “at least two.” The parties’ dispute thus centers on the following limitations:

receiving ***second values associated with the voice input from other electronic devices*** of the plurality of electronic devices (claims 1 and 10)

receive respective ***second values from other electronic devices of the plurality of electronic devices*** (claim 16)

'311 patent, cls. 1, 10, 16 (emphasis added).

For the same reasons described in Section VI.B, the meaning of these terms is clear: “other electronic devices” may refer to one or more devices, and “second values” may refer to one or more “second values.” Sonos’s constructions would exclude two-device configurations, are unsupported by the intrinsic evidence, and contravene the relevant legal authority.

D. “forgo outputting an audible response” / “forgoing outputting a response” ('128 All Claims) / “[forgoing / forgo] responding” ('311 Claims 2, 3, 12)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“[refrain]/[refraining] from outputting any and all [audible] responses”

The meaning of the forgo/forgoing terms is clear, and no construction is necessary. Sonos nevertheless offers a construction that improperly seeks to inject the requirement that the device forgo “any and all” responses. Sonos’s added restriction highlights the parties’ dispute regarding what qualifies as “responding” to a detected voice input—under Sonos’s interpretation, a device that “forgo[es]” responding cannot perform any action whatsoever relating to a detected voice input, including “fulfilling” a user request. This undue restriction lacks support in the claims and is inconsistent with the specification.

The forgo/forgoing terms appear in the following limitations of each independent claim of the '128 patent:

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, wherein the other devices of the plurality of electronic devices ***forgo outputting an audible response to the detected voice input;***

in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic device, ***forgoing outputting a response to the detected voice input.***

'128 patent, cls. 1, 6, 11 (emphases added). Claims 2, 3, and 12 of the '311 patent also recite “forgo[ing] responding” to the detected voice input. As the plain language of the claims makes clear, if a device in the plurality of electronic devices is determined to not have the highest quality score (or, in the context of the '311 patent, to not have a higher first value), then the device does not output an audible and/or visual response to the detected voice input. A POSITA would understand this language as written, and therefore there is no need to construe it. Ex. 6 (Anderson Decl.) at ¶ 94.

The specification confirms that the claimed “forgo[ing]” is tied to a response ***to the detected voice input***. For example, the specification describes that, “[i]f its score is not the best (e.g., highest) amongst the scores, the electronic device ***forgos outputting responses...to the voice input.***” '128 patent, 21:61-65 (emphasis added); *see also id.*, 23:4-7, 29:59-30:12. Examples of responding to a detected voice input include outputting an audible response (e.g., “voice confirmation of the user request, reading out of the user-requested information”) and/or providing a visual response (e.g., “a notification or a LED light pattern”). *Id.*, 29:32-54. In contrast, a device that forgoes responding to a detected voice input may, for example, power down a speaker or associated amplifier. *Id.*, 21:61-65, 23:4-7, 29:67-30:1. In this manner, the voice assistant devices

determine a “leader” that will be responsible for responding to the user’s voice inputs, and in order to reduce user confusion, other devices that are not the “leader” forgo responding to the user. *Id.*, 1:43-53.

The specification also describes the distinct process of fulfillment, by which a device takes an action or operation in response to an instruction relating to the user’s request. For example, the specification describes that a user may make “a request for information or for performance of an action or operation,” and “[t]he electronic device 190 **responds to the request with voice message responses to the user**, where the response may include, for example, requests for additional information to **fulfill the request**, confirmation that the request has been **fulfilled**, notice that the request cannot be **fulfilled**, and so forth.” ’128 patent, 10:2-20 (emphases added); *id.*, 24:61-25:7 (a device may determine a request from a voice input and “generate **a response that is responsive to the request** (e.g., an audible acknowledgement of the request, an audible inquiry to the user for more information needed to **fulfill the request**).”) (emphases added). Thus, the specification distinguishes responses to a detected voice input (e.g., requests for additional information or confirmation messages) from acts devices might take to fulfill an action requested within that voice input.

As described in the specification, fulfillment need not be performed on a device that detects a voice input; it may take place on a device other than the device that interacts with the user. For example, “the user can issue a media play request by saying to the Wi-Fi speaker ‘OK Google, play cat videos on my Living room TV.’ The voice assistant service then **fulfils the media play request** by playing the requested media content on the requested device using a default or designated media application.” ’128 patent, 9:38-43 (emphasis added). In this case, if a device other than “Living room TV” detected the voice input, the “response” to the detected voice input

is the instruction sent from the detecting device to the Living room TV (i.e., the fulfilling device) instructing that TV to perform the requested action; the fulfillment performed by the Living room TV (i.e., playing cat videos) would not be the “response to the detected voice input.” *Id.*, 24:45-51 (“For example, in response to a hotword voice input to awaken the electronic device 190, the electronic device 190 generates and presents (e.g., outputs) a...**response [which] may include a voice output and/or instructions to another device to perform an operation.**”) (emphasis added). Thus, the ’128 and ’311 patents describe responding to a voice input as separate and distinct from fulfillment, and the claims do not preclude a device that forgoes outputting a response to a voice input from fulfilling an action based on an instruction from the device that does respond to the detected voice input.

By injecting the requirement that a device “refrain from outputting **any and all** responses,” Sonos’s proposed construction would broaden the scope of the claimed “response[s]” to encompass fulfillment (or any action whatsoever).⁷ As described above, this restriction is contrary to the intrinsic evidence. The patents contemplate that a device may respond to a detected voice input by issuing instructions to other devices to fulfill a user’s request, but does not describe the operations or actions taken by the other devices in response to those instruction as “responding to the voice input.” ’128 patent, 9:38-43, 24:45-51. Instead, the patents refer to that those actions as “fulfill[ment].” *Id.*, 10:2-20, 24:61-25:7. Sonos’s proposal would preclude devices that “forgo” responding from “fulfill[ing]” such actions, contrary to the specification’s teachings.

⁷ Sonos also replaces the claim term “forgoing” with “refraining from,” which is unnecessary because the meaning of “forgoing” is clear based on its use throughout the specification. *See, e.g.*, ’128 patent, 21:58-65, 22:66-23:7, 29:59-30:12. In contrast, neither the word “refrain” nor the phrase “refraining from” is used in the specification.

Sonos’s proposal is also improper because it would preclude other device operations that are contemplated by the patents (separate from fulfillment of a user’s request). The patents describe that all devices may visually indicate that they are listening to a detected voice input, while only the device with the highest quality score will respond to that input. See ’128 patent, 23:35-45 (“In some implementations, ***each of the electronic devices 1002 outputs a visual indication*** (e.g., a predefined LED light pattern) ***that it is listening*** whenever it is listening to a user (e.g., receiving and detecting the user’s voice input), and then ***just the leader device outputs the response to the voice input*** per the leadership determination.”) (emphases added). Sonos’s attempt to exclude such a visual indication that a device is listening to a detected voice input, but not yet responding, underscores that its proposed construction is incorrect.

No construction of the “forgo/forgoing” terms is necessary; these terms should be given their plain and ordinary meaning. Sonos’s proposal should be rejected because it imports limitations into the claims and is based on an interpretation of “response to the detected voice input” that is inconsistent with the intrinsic record.

- E. **“in accordance with a determination that the quality score generated by the first electronic device is the highest ... in accordance with a determination that the quality score generated by the first electronic device is not the highest” (’128 Claims 1, 2, 3, 5)⁸ / “in accordance with a determination that the first value is higher than the second values ... in accordance with a determination that the first value is not higher than the second scores” (’311 Claim 2)**

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	Indefinite

The meaning of the “in accordance with a determination...” terms is clear from the context of the claims and the specification, and no construction is necessary. In asserting indefiniteness,

⁸ Claims 2, 3, and 5 of the ’128 patent depend from claim 1.

Sonos contends the claims lead to a non-sensical result, asserting that “it is impossible for an ‘electronic device’ to perform both of these method steps for a given voice input.” See Ex. 10 (Sonos Invalidity Contentions) at 6-8. That is simply incorrect.

’128 patent claim 1 and ’311 patent claim 2 recite methods in which two of the steps are presented as alternatives. Whether one or the other is taken depends on what happens at earlier conditional steps of the claims, i.e., the claim terms in dispute. Thus, for example, the step of “**outputting** an audible and/or a visual response to the detected voice input” is performed “in accordance with a determination that the quality score generated by the first electronic device is **the highest** amongst the generated quality score and received quality scores for the voice input.” ’128 patent, cl. 1. And “**forgoing outputting** a response to the detected voice input” is alternatively performed “in accordance with a determination that the quality score generated by the first electronic device is **not the highest** amongst the quality scores for the voice input generated by the plurality of electronic devices.” *Id.* Claim 2 of the ’311 patent recites similar steps using a similar framework.⁹ ’311 patent, cl. 2. Nothing about these conditional steps renders these claims indefinite. Rather, the plain meaning of the disputed phrases in the context of the claims and specification is clear, and thus Google’s proposals should be adopted.

First, method claims may include conditional steps. *Cybersettle, Inc. v. Nat’l Arbitration Forum, Inc.*, 243 F. App’x 603, 607 (Fed. Cir. 2007) (“It is of course true that method steps may be contingent.”); see also *Hytera Commc ’ns Co. v. Motorola Sols., Inc.*, 841 F. App’x 210 (Fed. Cir. 2021) (affirming validity over prior art of claim with limitation “written in conditional

⁹ Google understands Sonos’s position to be the same for the ’128 and ’311 terms. For purposes of this Markman Brief, Google uses the “quality score” and “highest” / “not highest” language from claim 1 of the ’128 patent. Claim 2 of the ’311 patent uses the same framework (i.e., “value” and “higher” / “not higher”), and the arguments presented here apply equally to that claim.

language, with two alternative conditions and corresponding responses”); *Interdigital Tech. Corp. v. Lenovo Holding Co., Inc.*, No. CV 19-1590-LPS, 2021 WL 1856937, *5 (D. Del. May 10, 2021) (construing alternative conditional limitations).

Second, the claim language and specification confirm the plain meaning of the disputed phrases. As the specification explains, “if” has the same meaning as the claim phrase “in accordance with a determination that.” *Id.*, 35:6-9 (“As used herein, the term ‘*if*’ is, optionally, construed to mean ‘when’ or ‘upon’ or ‘in response to determining’ or ‘in response to detecting’ or ‘*in accordance with a determination that,*’ depending on the context.”) (emphases added). The claim’s use of “in accordance with a determination that” thus provides alternative conditions precedent, which would have been readily understood by a POSITA within the context of the patents. Ex. 6 (Anderson Decl.) at ¶¶ 111, 113-17.

The specification also explains that the conditional aspect of the disputed claim terms derives from the very purpose of the claimed invention. The ’128 and ’311 patents present solutions to problems that arise when a single location (e.g., a room) includes “multiple devices that include voice assistant systems and one or more users of such devices.” ’128 patent, 1:43-45, 59-63. To address these issues, the patents disclose methods for “multi-device coordination” to determine “a leader amongst the voice assistant devices that is responsible for responding to the user’s voice inputs, in order to reduce user confusion.” *Id.* at 1:50-53, 59-63. In this context, a POSITA would expect the claimed invention to encompass steps that a device takes if it is the “leader,” and steps that it alternatively takes if it is not the “leader.” Ex. 6 (Anderson Decl.) at ¶ 118.

The specification thus mirrors the conditional framework of the claims, consistently using the phrase “in accordance with a determination that” to describe the alternative functionalities. For

example, the specification describes that, “[i]n accordance with a determination that the quality score generated by the first electronic device **is the highest**...the electronic device **outputs** (1210) an audible response and/or a visual response (e.g., a notification or a LED light pattern) to the detected voice input.” ’128 patent, 29:32-40 (emphases added). The specification describes the alternative using the same language: “[i]n accordance with a determination that the quality score generated by the first electronic device is **not the highest**...the electronic device 190 **forgoes outputting** a response to the detected voice input.” *Id.*, 29:59-63 (emphases added). The specification confirms the conditional nature of this step through its use of the term “if” in the next sentence: “**If** a device determines that its score is not the highest amongst the devices with respect to the voice input (e.g., devices 1002-2 and 1002-3), the device does not output a response to the voice input.” *Id.*, 29:63-67 (emphasis added); *see also id.* at 21:58-22:10 (using “if” to describe the same operations). These descriptions confirm to a POSITA that the disputed limitations recite conditional steps. Ex. 6 (Anderson Decl.) at ¶ 119.

In asserting that these limitations are indefinite, Sonos ignores both their plain meaning in the context of the patents and well-settled law. First, Sonos’s indefiniteness argument is premised on the notion that the claims lead to a non-sensical result by requiring a determination that the first electronic device’s quality score is both “the highest” **and** “not the highest,” and that the first electronic device must therefore both output a response **and** forgo outputting a response. As explained above, a POSITA would readily understand the conditional framework of the claims, particularly in light of the specification’s teachings. Ex. 6 (Anderson Decl.) at ¶¶ 111-18. A POSITA would also readily appreciate that both conditions stated in the ’128 and ’311 claims cannot be met simultaneously, and therefore would not interpret the claims as Sonos suggests. Ex. 6 (Anderson Decl.) at ¶ 120. *AIA Eng’g Ltd. v. Magotteaux Int’l S/A*, 657 F.3d 1264, 1276 (Fed.

Cir. 2011) (“We strive, where possible, to avoid nonsensical results in construing claim language.”).

Second, Sonos’s position contravenes Federal Circuit authority. *See, e.g., Hytera*, 841 F. App’x at 215-218 (discussing *Lincoln Nat. Life Ins. Co. v. Transamerica Life Ins. Co.*, 609 F.3d 1364 (Fed. Cir. 2010)). Following that authority, the argument Sonos makes here was rejected in *Interdigital*, which involved a claim with conditional, “mutually exclusive alternative[]” steps: “[s]tep two recites a **condition** in which an explicit allocation **is** received...step three recites the **condition** in which an explicit allocation **is not** received.” 2021 WL 1856937 at *5 (emphases original). The court held that “[i]t would be impossible for a system both to receive and not receive an explicit allocation at any given time” and, “to infringe[,] a system must be capable of addressing each alternative, regardless of which alternative occurs at any particular point.” *Id.* (citing *Hytera* and *Lincoln Nat. Life Ins. Co. v. Transamerica Life Ins. Co.*, 609 F.3d 1364 (Fed. Cir. 2010)). The same logic applies here.

Because a POSITA would readily understand these terms, they should be given their plain and ordinary meaning, and Sonos’s indefiniteness argument should be rejected.

F. “cause the electronic device” (’128 Claims 11, 12, 13, 15)¹⁰

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	Indefinite

The phrase “cause the electronic device” has a plain meaning in light of the ’128 patent claims and specification. Sonos’s indefiniteness position is premised on an assumption that the preamble of claim 11 does not specify whether “the electronic device” in the disputed phrase is

¹⁰ Although this phrase does not appear in claims 12 and 13, a construction would affect these claims because they depend from claim 11.

the same device as the “first electronic device.” See Ex. 10 (Sonos Invalidity Contentions) at 7-8.¹¹ It is the same device, as the ’128 patent confirms. Ex. 6 (Anderson Decl.) at ¶ 124. No construction is necessary.

First, the preamble of claim 11 recites “[a] non-transitory computer readable storage medium *storing one or more programs, the one or more programs comprising instructions, which, when executed by a first electronic device* of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more microphones, a speaker, and one or more processors, *cause the electronic device* to perform [the operations recited in the claim].” ’128 patent, cl. 11 (emphases added). The only device of “a plurality of electronic devices” that claim 11 specifies is “the first electronic device.” *Id.* Accordingly, a POSITA would understand that the only possible antecedent basis for “the electronic device” recited in claim 11 is the “first electronic device.” Ex. 6 (Anderson Decl.) at ¶ 125. See *Joy MM Delaware Inc. v. Cincinnati Mine Mach. Co.*, No. CIV.A. 09-1415, 2010 WL 3984822 (W.D. Pa. Oct. 9, 2010), *aff’d*, 497 F. App’x 970 (Fed. Cir. 2012) (claim’s recitation of “a chain and flight assembly” and “said *conveyor* chain and flight assembly” did not render it indefinite where the claim only referred to one “chain and flight assembly”).

This understanding is confirmed by the context of claim 11, which recites “instructions” that are executed by “a first electronic device.” The claim further recites that these instructions, when executed, cause “the electronic device” to perform a number of operations that the claim confirms are carried out by the “first electronic device.” ’128 patent, cl. 11 (“the quality score generated by the *first electronic device*”) (emphasis added). Because “the electronic device”

¹¹ Sonos asserts in its contentions that claims 11, 12, 13 and 15 are indefinite “[f]or at least this reason,” but has not responded to Google’s request for identification of any additional reasons for its indefiniteness position. Ex. 10 (Sonos Invalidity Contentions) at 8.

performs the recited operations based on execution of instructions, a POSITA would understand “the electronic device” to be the same device that executes those instructions (i.e., the “first electronic device”). Ex. 6 (Anderson Decl.) at ¶ 126.

Second, the specification confirms that “the electronic device” of claim 11 is the recited “first electronic device.” For example, Figure 6 of the ’128 patent illustrates a series of operations that correspond to those recited in claim 11, and shows that those operations are performed “[a]t a first electronic device of a plurality of electronic devices.” ’128 patent, Fig. 6. At the same time, the specification describes Figure 6 as showing a method that is “implemented at an **electronic device**” of two or more electronic devices. *Id.*, 28:65-30:12 (emphasis added). Based on this description, a POSITA would understand that the electronic device recited in claim 11, which is caused to perform the same operations illustrated in Figure 6, is the same as the “first electronic device” recited in claim 11. Ex. 6 (Anderson Decl.) at ¶¶ 127-28. The intrinsic record supports only this interpretation. *See Eidos Display, LLC v. AU Optronics Corp.*, 779 F.3d 1360, 1364-66 (Fed. Cir. 2015) (claim term not indefinite where the specification only described one type of structure).

For the same reasons, Sonos’s indefiniteness position regarding “cause the electronic device” is inconsistent with the intrinsic record and should be rejected. No construction is necessary, and this term should be given its plain and ordinary meaning.

G. “the second scores” (’311 Claims 2, 3)¹²

Complainant’s Proposed Construction	Respondent’s Proposed Construction
“the second values”	Indefinite

¹² Although the term “the second scores” does not appear in claim 3, a construction of “the second scores” would affect claim 3 because it depends from claim 2.

The proper construction of “the second scores” is “the second values.” Google’s construction corrects an obvious clerical error and is consistent with a POSITA’s understanding of the claim. Ex. 6 (Anderson Decl.) at ¶ 131. Sonos’s only stated basis for contending this term is indefinite is lack of antecedent basis.¹³ See Ex. 10 (Sonos Invalidity Contentions) at 7. Sonos advocates for the draconian result of indefiniteness in the face of a record that shows the Applicant’s intent—and inadvertent failure—to correct “the second scores” to “the second values” during prosecution. But Sonos’s argument should be rejected here, as the error is obvious and a POSITA would understand the scope of the claim with reasonable certainty. Ex. 6 (Anderson Decl.) at ¶ 131. Sonos’s arguments should be rejected.

First, the ’311 patent claims themselves confirm that “second scores” was intended to mean “second values.” Dependent claim 2 recites: “in accordance with a determination that the first value is **not higher** than the second scores, **forgoing responding** to the detected input.” Claim 2 is effectively the inverse of the last element of its independent claim 1, which recites, “in accordance with a determination that the first value is **higher** than the second values, **responding** to the detected input.” ’311 patent, claims 1-2 (emphasis added). A POSITA would understand from the parallel structure of the language in claims 1 and 2 that “the second scores” (of claim 2) is intended to refer to the same “second values” of independent claim 1. Ex. 6 (Anderson Decl.) at ¶ 132. Indeed, claims 1 and 2 together recite a comparison with two alternative directives for the claimed device: claim 1 covers what the device should do if the first value is higher than the second value; claim 2 adds what the device should do if the second is higher than the first. It would make no sense to compare two different **types** of metrics (i.e., scores with values), especially

¹³ Sonos asserts in its contentions that “the second scores” is indefinite “[f]or at least the reason that the term lacks antecedent basis,” Ex. 10 at 7, but Sonos has confirmed that its position regarding this term is based solely on the alleged lack of antecedent basis.

when claims 1 and 2 are directed to the same comparison to determine whether the first electronic device should respond to the detected input. This is particularly so given that the patent makes no reference to a comparison of a “score” to a “value,” and a POSITA would recognize that the claim limitations recite alternative responsive steps to a single comparison (whether the first value is higher or lower). Ex. 6 (Anderson Decl.) at ¶ 133.

Claim 4, which also depends from claim 1, confirms that “the second scores” was an error. Claim 4 recites the same limitation as claim 2, referring to the correct term “the second values,” rather than “scores”: “The method of claim 1, further comprising: in accordance with a determination that the ***first value is not higher than the second values***, identifying a criterion associated with the voice input.” A POSITA reading claims 1 and 4 would understand that “the second scores” in claim 2 was an error and that the term should be “the second values.” Ex. 6 (Anderson Decl.) at ¶ 134.

Second, the prosecution history confirms that the phrase “the second scores” appears in claim 2 in error. In a preliminary amendment, the Applicant revised then-pending claim 1 to replace the term “score” with “value,” and added new claims, including those that issued as claims 2 and 3. Ex. 12 (’311 File History) at 4 (Mar. 19, 2020 Prelim. Amend.) (“[Claim 1:] in accordance with a determination that the first value ~~score~~ is higher than the second values ~~scores~~, responding to the detected input . . . [Claim 2:] in accordance with a determination that the first score is not higher than the second scores, forgoing responding to the detected input.”) (annotations original). After the claims were allowed,¹⁴ the Applicant corrected claim 2 “for clarity and to correct for informalities.” *Id.* at 26 (Nov. 25, 2020 Response to Office Action). In that Office Action

¹⁴ The Office Action allowed the claims pending the Applicant’s filing of a terminal disclaimer. Ex. 12 (’311 File History) at 11-16, 26.

Response, the Applicant corrected the first instance of “score” to “value,” without correspondingly correcting the second instance of “scores” to “values” to be consistent with claims 1 and 4. *Id.* at 22-26; Ex. 6 (Anderson Decl.) at ¶¶ 135-37. The prosecution history thus demonstrates the Applicant’s intent to correct “scores” to “values.”

The only basis Sonos has articulated for its indefiniteness position is that “second scores” has no antecedent basis in the claim. Ex. 10 (Sonos Invalidity Contentions) at 7. But there can be no reasonable debate that the Applicant’s failure to amend “the second scores” during prosecution (as it did with the first instance of “score” in the claim) was an error that would have been obvious to a POSITA. Ex. 6 (Anderson Decl.) at ¶¶ 131, 135-37. For the reasons set forth above, a POSITA would understand with reasonable certainty that “second scores” means “second values” in claim 2 of the ’311 patent. Ex. 6 (Anderson Decl.) at ¶ 131.

H. “the detected output” (’311 Claim 12)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
“the detected input”	Indefinite

Similar to “the second scores,” a POSITA would understand “the detected output” to be an obvious typographical error. *See supra* Section VI.G. As the claims and specification show, the ’311 patent is focused on detecting voice **inputs**, not outputs, and on determining which device should respond to a given voice input. Sonos again seeks to invalidate Google’s patent claims based on a clerical error, ignoring the clear intent of the patentee.¹⁵ But the only reasonable interpretation of “the detected output” in claim 12 of the ’311 patent is “the detected input.”

¹⁵ Sonos asserts in its contentions that “the detected output” is indefinite “[f]or at least the reason that the term lacks antecedent basis,” Ex. 10 at 7, but Sonos has confirmed that its position regarding this term is based solely on the alleged lack of antecedent basis.

First, apart from the error in claim 12, the '311 patent claims only recite “voice input[s]”—never outputs. '311 patent, cls. 1-4, 8-11, 16, 20 (all reciting “a[/the] detected input,” not “output”). Significantly, claim 10, from which at-issue claim 12 depends, recites “detecting a voice input,” “determining a first value associated with the voice input,” “receiving second values associated with the voice input,” and then “**responding** to the detected input.” The first electronic device of claim 10 is handling the detected input, and its dependent claim 12 adds a limitation whereby the first electronic device “communicat[es] to the other devices . . . to **forgo responding** to the detected [input],” i.e., the input to which the first electronic device responded. These claims would not make sense together—as they must—if the first electronic device told the other devices to forgo detecting an output instead of an input. A POSITA would understand that “output” was nothing more than a clerical error. Ex. 6 (Anderson Decl.) at ¶ 140.

Second, the specification supports Google’s construction. In every described instance in which a device forgoes a response, it forgoes responding to a “detected voice input,” never to an output. See '311 patent, Fig. 6 (step 1210, “wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input;” step 1212, “forgo outputting a response to the detected voice input”), 3:8-15, 3:32-39, 3:57-64, 22:18-20 (“the leaders forgo receiving, detecting, and processing subsequent voice inputs from the user”).¹⁶ Indeed, the '311 patent uses the word “detect[ed/ing]” in connection with “voice input” more than forty times; the **only** instance of “detected voice output” is the error in claim 12.¹⁷ The intrinsic

¹⁶ Although the specification includes embodiments that forgo “outputting responses” in response to a detected input, nowhere does it disclose **detecting** an output. See '311 patent, 10:9-10, 22:4-7, 23:16-17, 29:50-52, 30:7-8, 30:20-24, 30:58-60.

¹⁷ The limitation “detected output” was added during prosecution. Claim 12 was added during prosecution in a March 19, 2020 preliminary amendment and then allowed on the next office action, pending a terminal disclaimer. Ex. 12 ('311 File History) at 6 (Mar. 19, 2020 Prelim. Amdmt.), 15 (Nov. 5, 2020 Non-Final Office Action).

record thus supports adopting Google’s proposed construction. *See Howmedica Osteonics Corp. v. DePuy Orthopaedics, Inc.*, No. 11-CV-6498SDW MCA, 2013 WL 3455727 at *28 (D.N.J. July 9, 2013) (correcting “inner surface” to “outer surface” where “[t]hroughout the claim language and patent specification, the [patent] identifies the locking member being located on the outer part of the bearing”).

Sonos contends that “the detected output” is indefinite because it lacks antecedent basis. Ex. 10 (Sonos Invalidity Contentions) at 7. For the reasons discussed above, a POSITA would understand the term in the context of the ’311 patent, and Sonos’s argument should be rejected.

VII. ’615 PATENT

A. Overview of the ’615 Patent

The invention of the ’615 patent provides “more user-friendly, secure, and effective mechanisms for device commissioning than” the known prior art commissioning processes. ’615 patent, 6:35-63. Specifically, the ’615 patent invention simplifies the process of adding a new device to a communication network, like a wireless local area network, by using a smart wireless device, like a cellphone, tablet, or other device, that is already part of that communication network to “commission” (i.e., add) the new device onto the network. *Id.*, 7:47-55. By using a smart wireless device, a user is not required to take cumbersome steps to add new devices to their existing networks. *Id.*, 4:24-30, 6:59-65, 7:64-67, 8:9-14. Additionally, by using a short-range proximity-based transmission path, the improved commissioning process in the ’615 patent is more secure than other alternatives; the wireless device must be close to the new device to carry out the commissioning process. *Id.*, 6:35-63. To do this, the ’615 patent uses proximity-based transmission paths like Bluetooth communication, radio frequency identification (RFID), Zigbee, radio frequency for consumer electronics (RF4CE), Z-Wave, and near field communication. *Id.*, 8:50-54, 62-67. This improves on prior art commissioning techniques by eliminating the need for

physical and/or administrative access to an access point, special user skills, the creation of ad hoc networks, the disruption of the existing connections to the communication network, or potential compromises to sensitive data. *Id.*, 4:24-30, 6:59-65, 7:64-67, 8:9-14.

B. “short range signal transmission path” (All Claims)

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	Indefinite

The ’615 patent describes using a short-range signal transmission path between two devices: a “wireless device” that is part of a communication network, and a “target device” that is not yet part of that communication network.¹⁸ Specifically, claim 1 of the ’615 patent recites an inventive “method for commissioning a target device onto a communication network” that includes steps of “establishing a short-range signal transmission path with a wireless device” and “receiving one or more network configuration packets from the wireless device via the short-range signal transmission path.” ’615 patent, cl. 1. Dependent claim 5 specifies that this “short-range signal transmission path” can be “based on one of Bluetooth communication, radio frequency identification (RFID), Zigbee, radio frequency for consumer electronics (RF4CE), Z-Wave, and near-field communication.” *Id.*, cl. 5. Claims 11 and 16 likewise describe devices storing instructions for “establishing a short-range signal transmission path” and “receiving one or more network configuration packets from the wireless device via the short-range signal transmission path.” *Id.*, cls. 11, 16.

¹⁸ See, e.g., ’615 patent, claim 1 (“establishing a short-range signal transmission path with a wireless device independently of the communication network”), claim 11, claim 16; see also *id.*, 8:9-13 (“In one embodiment, the smart wireless device 321 utilizes a side band, proximity based, communication channel 322 that is established between the smart wireless device 321 and the other device 307 to be commissioned onto the WLAN 304.”).

A POSITA would understand that a “short-range signal transmission path” in the ’615 patent is a communications channel established using well-known short-range protocols identified in the ’615 patent, or similar protocols. For example, the ’615 patent teaches using a “proximity based communication channel” for transmissions between the “smart wireless device” that initiates the commissioning and the “other device” (i.e., target device) that is being commissioned. ’615 patent, 8:9-13 (“the smart wireless device 321 utilizes a side band, proximity based, communication channel 322 that is established between the smart wireless device 321 and the other device 307 to be commissioned onto the WLAN”). This communication channel is used by the “transmitting and receiving elements” of the wireless device and the other device. *Id.*, 8:10-13-14. The patent also explains that these “elements” for transmitting and receiving may use specific exemplary radio frequencies (*id.*, 8:55-58), modulation techniques (*id.*, 8:58-62), and “communications standards such as, but not limited to, near field communication (NFC), RF identification (ID), BLUETOOTH LOW ENERGY, Zigbee, radio frequency for consumer electronics (RF4CE), Z-Wave, or other proprietary standards.” *Id.*, 8:62-67. The ’615 patent claims likewise confirm these protocols are each “short-range signal” protocols. *Id.*, cl. 5 (identifying Bluetooth, RFID, Zigbee, RF4CE, Z-Wave, and NFC as examples of the short-range signal transmission path in claim 1).

A POSITA would understand what a “short-range signal transmission path” is after reading the ’615 patent claims, specification, and file history. As described above, multiple objective examples are provided in the specification. The phrase “short-range” is also common in the field and well understood by persons of skill. Indeed, *Sonos itself uses “short-range” in its own*

patents, in the same way the '615 patent uses the phrase – to describe proximity-based protocols such as NFC and Bluetooth Low Energy.¹⁹

Sonos nonetheless argues “short-range signal” is indefinite in the '615 patent because “short-range” is a term of degree that, Sonos claims, has no objective boundary. This is wrong. As explained above, the '615 patent identifies a short-range signal as a “proximity based communication channel,” and discloses (and claims) multiple examples of such “short-range” signals, including NFC, RFID, BLUETOOTH LOW ENERGY, Zigbee, RF4CE, and Z-Wave. *Id.*, 8:62-67. In view of these disclosures, the '615 patent more than provides objective benchmarks that allows “a skilled artisan [to] compare” a product “with the examples in the specification to determine” if a transmission path is a short-range transmission path. *Sonix Tech. Co. v. Publications Int’l, Ltd.*, 844 F.3d 1370, 1379 (Fed. Cir. 2017); *see also Niazi Licensing Corp. v. St. Jude Med. S.C., Inc.*, 30 F.4th 1339, 1349 (Fed. Cir. 2022) (terms “resilient” and “pliable” not indefinite as “examples in the written description helped provide sufficient guidance to render the claims not invalid as indefinite”). This is all that is required for definiteness. *See Sonix*, 844 F.3d at 1379 (“visually negligible” definite where specification “includes examples of visually-negligible indicators” and discloses non-numerical criteria, including “differentiability, brightness, and homogeneity,” allowing one to “compare the examples and criteria from the written description . . . to determine whether an indicator is visually negligible”).

C. “in response to receiving the request, receiving one or more network configuration packets” (All Claims)

Complainant’s Proposed Construction	Respondent’s Proposed Construction

¹⁹ *See, e.g.*, Ex. 13 (U.S. Patent App. Pub. No. 2022/0244905), [0130]-[0131] (describing a “tap event” “generated using **a short-range wireless protocol**” such as “BLUETOOTH LOW ENERGY (BLE)” or “near-field communication (NFC)”; Ex. 14 (U.S. Patent App. Pub. No. 2022/0224971), claim 9 (claiming program instructions to stream “retrieved media in accordance with **a short-range wireless standard**”).

No construction necessary (plain and ordinary meaning)	Indefinite

The '615 patent's inventive method to “commission,” or add, a new device (the “target device” in the claims) onto a communication network begins with a “request” to start that process—i.e., “a request to initiate a commissioning process” with the target device.²⁰ Using this method, “in response to receiving *the request*” (i.e., the “request to initiate a commissioning process”), “configuration information” that lets the target device join the wireless network is received.²¹ The '615 Patent explains the commissioning process in detail, including how commissioning requests and configuration packets are received by the target device. Some embodiments place the target device “into a commissioning mode” before the target device receives network configuration data; other embodiments let the target device receive commissioning messages right away. *Id.*, 11:40-45. Some embodiments use a single commissioning message for the commissioning process; other embodiments employ multiple messages to carry out commissioning. *Id.*, 11:46-48. In claims 1, 11, and 16, multiple messages are used for commissioning, and the initial request to initiate commissioning kicks off the commissioning process. *Id.*, claims 1, 11, 16.

None of these steps are ambiguous or unclear. Sonos has not explained at all—much less by clear and convincing evidence—why “in response to receiving the request [to initiate a commissioning process with the target device], receiving one or more network configuration packets” would be indefinite.

²⁰ See, e.g., '615 patent, cls. 1, 11, 16 (“receiving a request to initiate a commissioning process with the target device”).

²¹ See *id.*

During the meet and confer process, Sonos could not articulate why it believes this claim term is indefinite. Sonos’s interrogatory responses on indefiniteness similarly fail to articulate an argument; they merely state “one of ordinary skill in the art would not understand **how** a device receives configuration packets in response to receiving the claimed request and the specification does not provide any guidance.” Ex. 10 (Sonos Invalidity Contentions), (emphasis added). But an alleged failure to disclose “how” a claim term is performed is not a basis for indefiniteness.²² It is thus unclear why Sonos contends this straightforward term is indefinite. If Sonos actually does articulate an indefiniteness argument, Google reserves the right to address that argument in its responsive brief.

VIII. CONCLUSION

For the reasons discussed above, Complainant respectfully requests that the ALJ adopt Complainant’s proposed constructions.

Dated: December 12, 2022

Respectfully submitted,

/s/ S. Alex Lasher

S. Alex Lasher

Jeff Gerchick

Nina S. Tallon

Quinn Emanuel Urquhart & Sullivan, LLP

1300 I Street, NW, Suite 900

Washington, DC 20005

Tel.: (202) 538-8000

David A. Nelson

Quinn Emanuel Urquhart & Sullivan, LLP

191 N. Wacker Dr., Suite 2700

Chicago, IL 60606

²² See, e.g., *TQ Delta, LLC v. 2Wire, Inc.*, 373 F. Supp. 3d 509, 524 (D. Del. 2019) (“[t]he asserted claims are clear on their face as to what is claimed” and an argument that “no embodiment demonstrating **how** to” perform the claim term “is not an issue of whether a person of ordinary skill in the art would understand the claim scope, but an issue of whether that claim is properly supported by written description or enabled by the patent specification”).

Tel.: (312) 705-7400

Patrick D. Curran
Quinn Emanuel Urquhart & Sullivan, LLP
111 Huntington Ave, Suite 520
Boston, MA 02199
Tel.: (617) 712-7100

Counsel for Complainant Google LLC

CERTAIN AUDIO PLAYERS AND COMPONENTS THEREOF (II)

Inv. No. 337-TA-1330

CERTIFICATE OF SERVICE

I, Jeffrey Matthews, hereby certify that on December 12, 2022, copies of the foregoing documents were served upon the following parties as indicated:

The Honorable Katherine Hiner Acting Secretary to the Commission U.S. International Trade Commission 500 E Street, S.W. Washington, D.C. 20436	☐ Via First Class Mail ☐ Via Hand Delivery ☐ Via Express Delivery ☒ Via EDIS
The Honorable Cameron R. Elliot U.S. International Trade Commission 500 E Street, S.W., Room 317 Washington, D.C. 20436 Email: michael.turner@usitc.gov	☐ Via First Class Mail ☐ Via Hand Delivery ☒ Via Electronic Mail ☐ Via EDIS
Jordan L. Coyle, Esq. ORRICK, HERRINGTON & SUTCLIFFE LLP Columbia Center 1152 15 th Street, NW Washington, DC 20005 Tel.: (202) 339-8400 Email: jcoyle@orrick.com Service Email: Sonos-1330-service@orrick.com <i>Counsel for Respondent Sonos, Inc.</i>	☐ Via First Class Mail ☐ Via Hand Delivery ☐ Via Express Delivery ☒ Via Electronic Mail

/s/ Jeffrey Matthews

Jeffrey Matthews

Exhibit 1



US009632748B2

(12) **United States Patent**
Faaborg

(10) **Patent No.:** **US 9,632,748 B2**
(45) **Date of Patent:** **Apr. 25, 2017**

(54) **DEVICE DESIGNATION FOR AUDIO INPUT MONITORING**

(56) **References Cited**

U.S. PATENT DOCUMENTS

(71) Applicant: **Google Inc.**, Mountain View, CA (US)

5,625,697 A 4/1997 Bowen et al.
6,038,614 A * 3/2000 Chan G06F 3/14
345/158
6,069,621 A * 5/2000 Schupak H04N 7/163
348/E7.061

(72) Inventor: **Alexander Faaborg**, Mountain View, CA (US)

7,184,960 B2 2/2007 Deisher et al.
2003/0167335 A1 * 9/2003 Alexander H04L 29/06
709/228

(73) Assignee: **Google Inc.**, Mountain View, CA (US)

2010/0088100 A1 * 4/2010 Lindahl G10L 15/30
704/270.1

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(Continued)

(21) Appl. No.: **14/445,255**

OTHER PUBLICATIONS

(22) Filed: **Jul. 29, 2014**

Howard, "Activate Google Now "listening" via bluetooth headset," Google Product Forums, Jun. 30, 2013, Retrieved from <<https://productforums.google.com/forum/print/msg/websearch/-cTsRaVV5F4/jeSHR0tqUqlj>> 2 pp.

(65) **Prior Publication Data**

US 2015/0370531 A1 Dec. 24, 2015

(Continued)

Primary Examiner — Marcus T Riley

(74) Attorney, Agent, or Firm — Shumaker & Sieffert, P.A.

Related U.S. Application Data

(60) Provisional application No. 62/016,603, filed on Jun. 24, 2014.

(51) **Int. Cl.**

G10L 21/00 (2013.01)
G06F 3/16 (2006.01)
G10L 15/22 (2006.01)
G10L 15/28 (2013.01)

(52) **U.S. Cl.**

CPC **G06F 3/167** (2013.01); **G10L 15/22**
(2013.01); **G10L 15/28** (2013.01); **G10L**
2015/223 (2013.01)

(58) **Field of Classification Search**

None

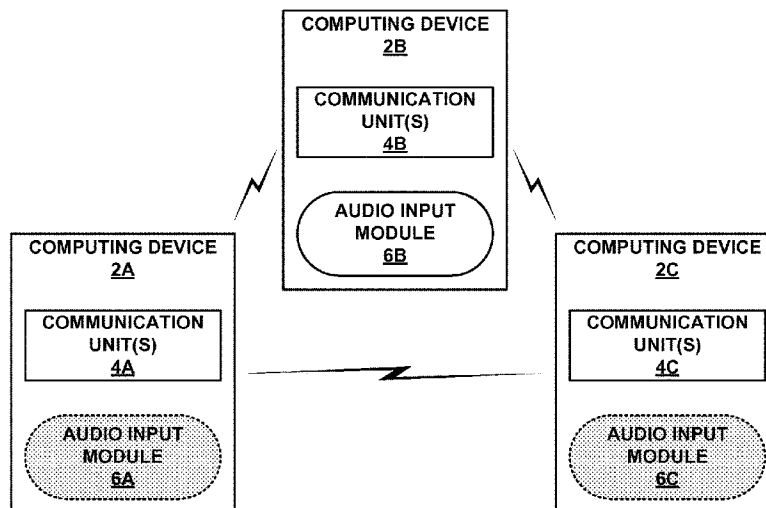
See application file for complete search history.

(57)

ABSTRACT

A computing device comprises at least one processor, and at least one module operable by the at least one processor to designate a particular computing device from a plurality of computing devices to process audio input, wherein the computing device comprises a first computing device from the plurality of computing devices. The at least one module may be further operable by the at least one processor to, if the particular computing device is not the first computing device, cease processing of audio input, and if the particular computing device is the first computing device, receive first audio input and process the first audio input to determine whether the first audio input includes a predetermined audio command.

20 Claims, 5 Drawing Sheets



(56)

References Cited

U.S. PATENT DOCUMENTS

2011/0257971	A1 *	10/2011	Morrison		G06K 9/00221
					704/233
2012/0034904	A1 *	2/2012	Lebeau		G10L 15/265
					455/414.1
2013/0091309	A1	4/2013	Bjontegard et al.		
2014/0038669	A1 *	2/2014	Shimizu		H04L 67/12
					455/556.1
2015/0370531	A1 *	12/2015	Faaborg		G06F 3/167
					704/275

OTHER PUBLICATIONS

“Issue 34348: Voice input should support Bluetooth microphones,” Android Open Source Project—Issue Tracker, Jun. 2013, Retrieved from <<https://code.google.com/p/android/issues/detail?id=34348>> 8 pp.

LeFebvre, “Tell Siri Which Audio Input to Listen to [iOS Tips],” Cult of Mac, May 14, 2013, Retrieved from <<http://www.cultofmac.com/227210/tell-siri-which-audio-input-to-listen-to-ios-tips/>> 8 pp. International Search Report and Written Opinion from counterpart International Application No. PCT/US2015/036961, dated Oct. 30, 2015, 10 pp.

“Exclusive: Google Will Soon Introduce ‘Nearby’ to Let other ‘People, Places, and Things’ Know When You’re Around,” Jun. 6, 2014, Retrieved from <http://www.androidpolice.com/2014/06/06/exclusive-google-will-soon-introduce-nearby-to-let-other-people-places-and-things-know-when-youre-around/>. 10 pp.

“Use Nearby to interact with what’s around you- Accounts Help,” Retrieved on Oct. 5, 2016 from <https://support.google.com/accounts/answer/62602867hl=en>, 5 pp.

U.S. Appl. No. 15/355,336, by Jian Wei Leong, filed Nov. 18, 2016.

* cited by examiner

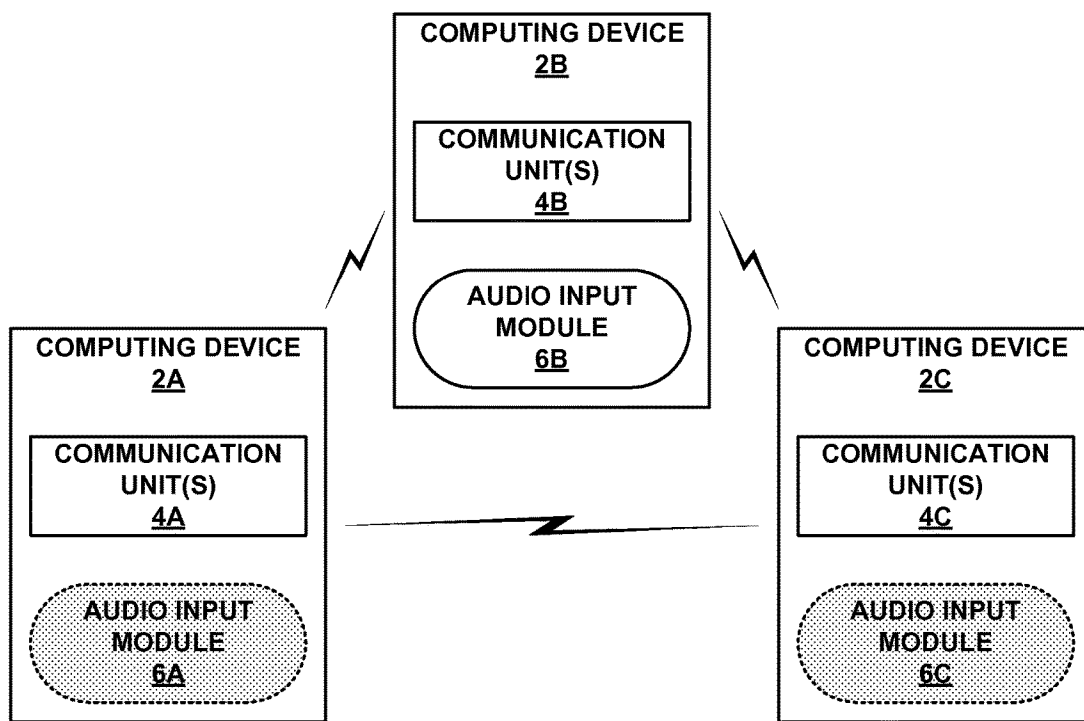


FIG. 1

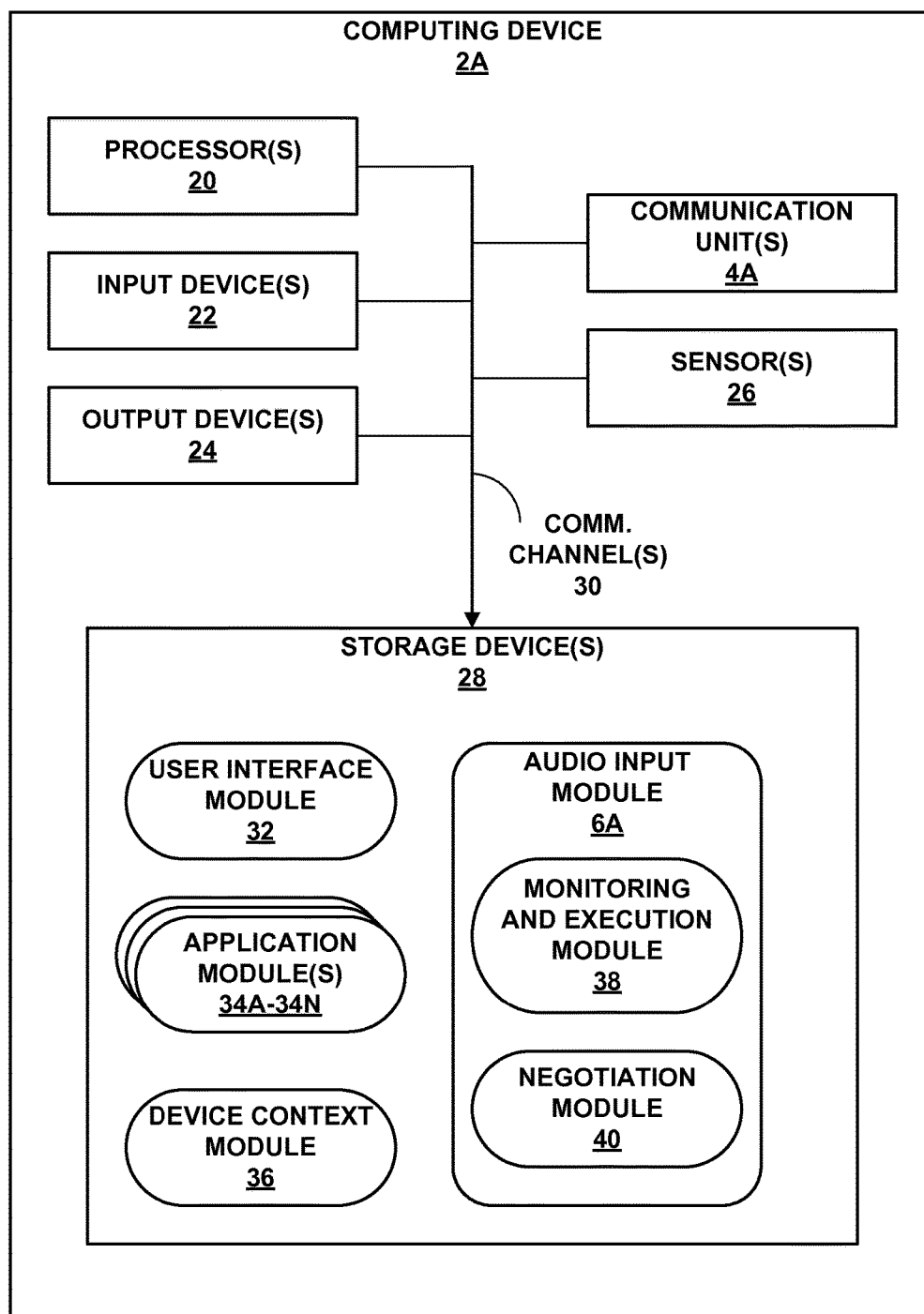


FIG. 2

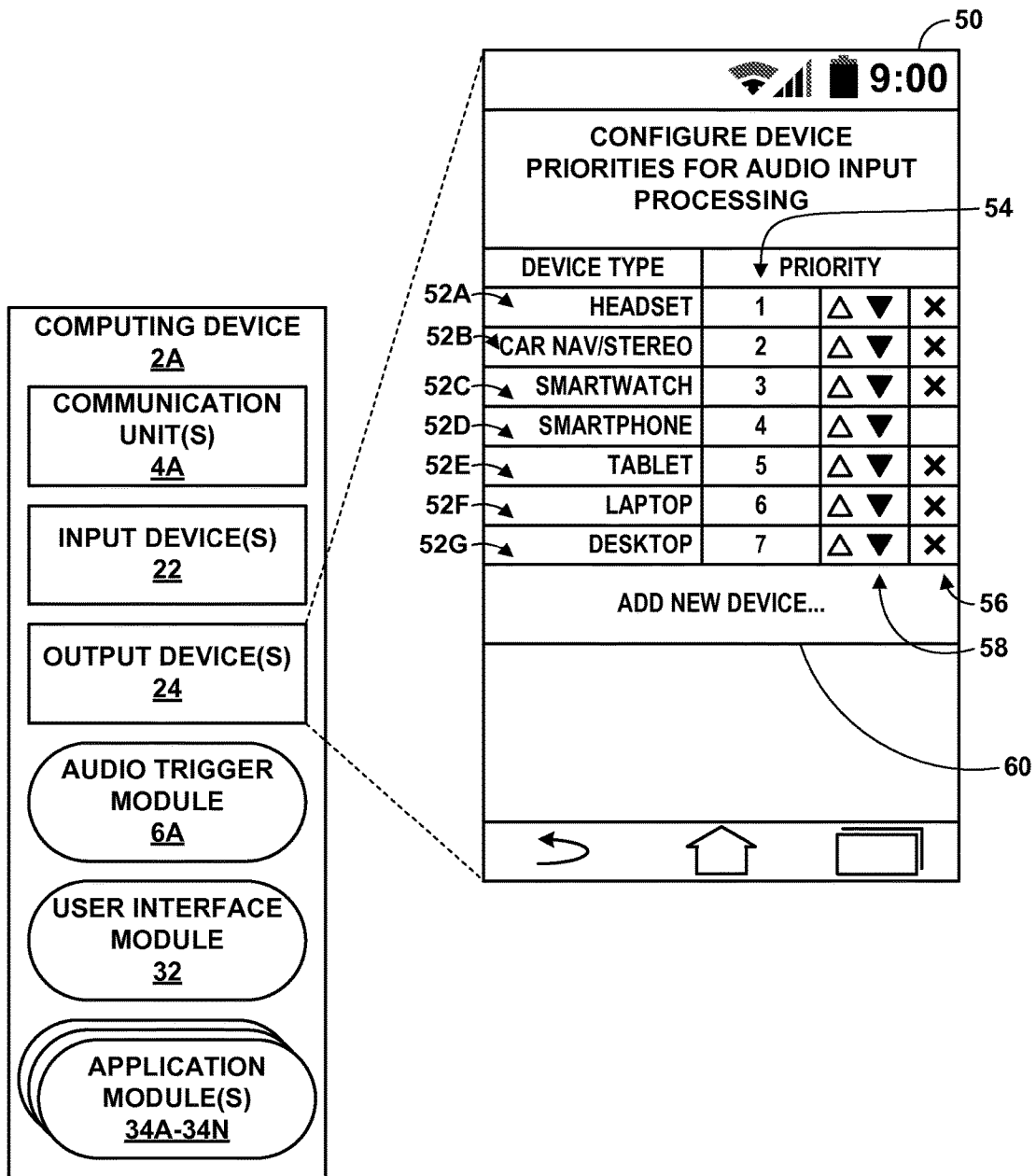
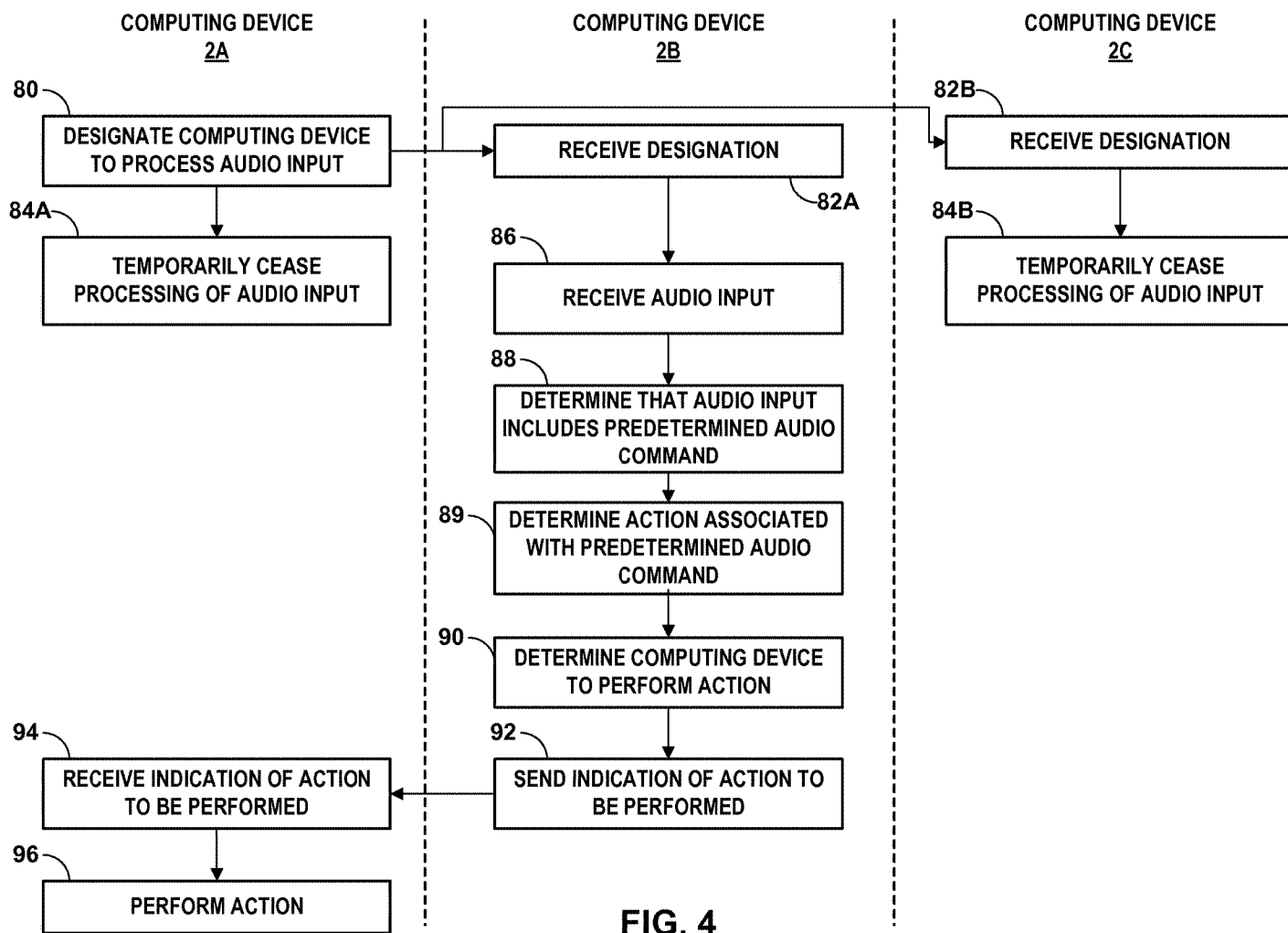


FIG. 3



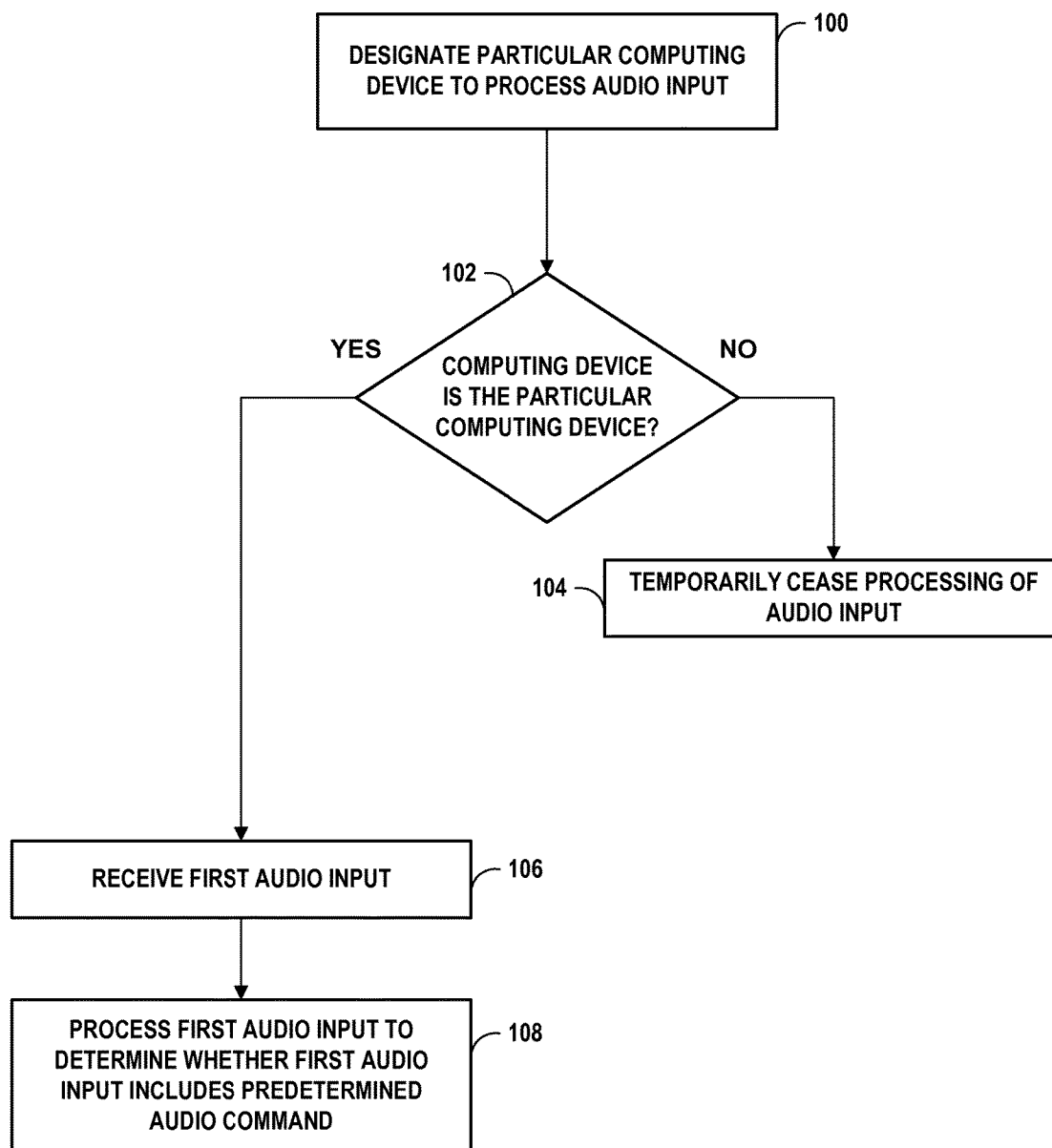


FIG. 5

1

**DEVICE DESIGNATION FOR AUDIO INPUT
MONITORING**

This application claims the benefit of U.S. Provisional Application No. 62/016,603, filed Jun. 24, 2014, the entire content of which is hereby incorporated by reference.

BACKGROUND

Computing devices (e.g., mobile phones, tablet computers, etc.) may enable a user to provide audio input. The audio input may cause the computing device to perform one or more actions associated with the received audio input, such as executing an application or creating a calendar appointment entry. In some cases, a computing device may monitor an environment for audio input that includes a predetermined audio command and, if the predetermined audio command is detected, perform a corresponding action. By monitoring for audio input that includes a predetermined audio command, a computing device may allow a user to provide, subsequent to providing the audio input that includes the predetermined audio command, one or more other audio commands without the user having to press a button or otherwise indicate that an audio command is about to be provided. In some situations, two or more computing devices associated with the user may monitor the same environment for audio input, possibly resulting in each of the computing devices performing actions based on predetermined audio input.

SUMMARY

In one example, a method includes designating, by a first computing device from a plurality of computing devices, a particular computing device from the plurality of computing devices to process audio input. The method may further include, if the particular computing device is not the first computing device, ceasing, by the first computing device, processing of audio input; and if the particular computing device is the first computing device: receiving, by the first computing device, first audio input, and processing, by the first computing device, the first audio input to determine whether the first audio input includes a predetermined audio command.

In another example, a computing device includes at least one processor and at least one module operable by the at least one processor to designate a particular computing device from a plurality of computing devices to process audio input, wherein the computing device comprises a first computing device from the plurality of computing devices. The at least one module may be further operable by the at least one processor to, if the particular computing device is not the first computing device, cease processing of audio input, and if the particular computing device is the first computing device, receive first audio input and process the first audio input to determine whether the first audio input includes a predetermined audio command.

In another example, a computer-readable storage medium is encoded with instructions that, when executed, cause at least one processor of a first computing device to receive, from a second computing device, a designation of a particular computing device from a plurality of computing devices to process audio input. The computer-readable storage medium may be further encoded with instructions that, when executed, cause the at least one processor of the computing device to, if the particular computing device is not the first computing device, cease processing of audio input, and if

2

the particular computing device is the first computing device, receive first audio input and process the first audio input to determine whether the first audio input includes a predetermined audio command.

The details of one or more examples are set forth in the accompanying drawings and the description below. Other features, objects, and advantages will be apparent from the description and drawings, and from the claims.

BRIEF DESCRIPTION OF DRAWINGS

FIG. 1 is a conceptual diagram illustrating an example computing environment that includes a plurality of computing devices configured to process audio input, in accordance with one or more techniques of the present disclosure.

FIG. 2 is a conceptual diagram illustrating details of one example of a computing device configured to process audio input, in accordance with one or more techniques of the present disclosure.

FIG. 3 is a conceptual diagram illustrating one example of a computing device configured to process audio input and an example graphical user interface (GUI) for configuring device type priorities, in accordance with one or more techniques of the present disclosure.

FIG. 4 is a flow diagram illustrating example operations of computing devices configured to process audio input, in accordance with one or more techniques of the present disclosure.

FIG. 5 is a flow diagram illustrating example operations of a computing device configured to process audio input, in accordance with one or more techniques of the present disclosure.

DETAILED DESCRIPTION

In general, techniques of the present disclosure may enable one or more computing devices from among a plurality of computing devices to designate a particular computing device to monitor for and/or act on audio input usable to initiate interaction with at least one of the plurality of computing devices. For instance, a plurality of computing devices (e.g., a smartphone, a smartwatch, and an in-vehicle entertainment/navigation system) may each be operable to monitor for and/or act on audio input that includes one or more predetermined audio commands. One or more of the computing devices (e.g., the smartphone, the smartwatch, and the smartwatch, etc.) may use techniques described herein to designate a particular computing device (e.g., an in-vehicle system) from among the plurality of computing devices as the best candidate for monitoring environmental audio for audio input that includes the one or more predetermined audio commands. Responsive to designating a particular computing device to monitor for the one or more predetermined audio inputs, other computing devices from the plurality of computing devices (e.g., the smartphone and smartwatch) may temporarily cease to monitor for audio input. Thereafter, the particular device may be the only one of the plurality of computing devices monitoring for audio input and, in some instances, upon receiving an indication of audio input that includes a predetermined audio command, the particular device may be the only one of the plurality of computing devices to perform one or more actions responsive thereto.

By designating a particular computing device to monitor for audio input and/or process indications of the audio input (e.g., monitor for one or more predetermined audio commands, such as “hotwords” or “audio triggers”) and sending

3

instructions to the other, non-designated computing devices to temporarily cease monitoring for audio input, computing devices and/or systems employing the techniques described herein may increase computational efficiency (by, e.g., avoiding redundant processing, avoiding parallel-processing hazards of operating on the same data, and/or improving device battery life). That is, when more than one computing device is operable to monitor for and/or act on audio input that includes a predetermined audio command, designating a single device to do so may ensure that a single device expends energy in the process, and that a single device performs actions responsive to the receipt of predetermined audio command.

FIG. 1 is a conceptual diagram illustrating an example computing environment 1 that includes a plurality of computing devices 2A-2C (collectively, “computing devices 2”) configured to process audio input, in accordance with one or more techniques of the present disclosure. Examples of computing devices 2 may include, but are not limited to, portable, mobile, or other devices, such as mobile phones (including smartphones), wearable computing devices (e.g., smartwatches, head-mounted devices, etc.), personal digital assistants (PDAs), in-vehicle devices (e.g., stereo systems, navigation systems, etc.), tablet computers, smart television platforms, laptop computers, desktop computers, server computers, mainframes, and the like. For instance, in the example of FIG. 1, computing device 2A may be a smartphone, computing device 2B may be an in-vehicle stereo/navigation system, and computing device 2C may be a smartwatch. In other examples, computing devices 2 may include additional computing devices, different computing devices, or fewer computing devices. That is, while described in the example of FIG. 1 with reference to three computing devices, computing devices 2 may include two computing devices, four computing devices, or more computing devices of various types.

Computing devices 2, as shown in the example of FIG. 1, include respective communication units 4A-4C (collectively, “communication units 4”). Each of communication units 4 may be operable to communicate with external devices via one or more wired and/or wireless networks by transmitting and/or receiving network signals on the one or more networks. Examples of communication units 4 include a network interface card (e.g. such as an Ethernet card), an optical transceiver, a radio frequency transceiver, a GPS receiver, or any other type of device that can send and/or receive information. Additional examples of communication units 4 may include Near-Field Communications (NFC) units, Bluetooth radios, short wave radios, cellular data radios, wireless network (WiFi) radios, infrared communication ports, as well as universal serial bus (USB) controllers. In other words, each of communication units 4 may be means by which the respective one of computing devices 2 may communicate with any other device, such as with each other.

In the example of FIG. 1, computing devices 2 include respective audio input modules 6A-6C (collectively, “audio input modules 6”). Each of audio input modules 6 may be operable to monitor the environment of the respective computing device for audio input and process the audio input to determine whether the audio input includes a predetermined audio command. The environment of a computing device may represent the real-world area immediately surrounding the computing device (e.g., within 1 meter of the computing device, within 5 meters of the computing device, within the same room as the computing device, or other designated area). A predetermined audio command, generally, may be

4

any audio input (e.g., a word, a phrase, etc.) that is usable to initiate interaction with a computing device. In other words, predetermined audio commands may be specific audio input for which a computing device may monitor environmental audio and that a user may use without having to otherwise inform the computing device that he or she is about to provide audio input.

One or more of audio input modules 6 may associate a predetermined audio command with an action to be performed by the respective computing device. For instance, one or more of audio input modules 6 may associate a word or phrase (e.g., “Hello phone,” “Okay car,” etc.) with entering an audio input mode, thereby allowing the user to provide further audio input. Other examples of predetermined audio inputs and associated actions may include a word or phrase (e.g., “answer”) that causes a computing device to answer a telephone call, a word or phrase (e.g., “navigation,” “directions,” etc.) that causes a computing device to provide directions, a word or phrase (e.g., “send email,” “send text message,” etc.) that causes a computing device to send a message, and others. In some examples, predetermined audio commands may be user-configurable. That is, a user may create, modify, enable, disable, or discard various predetermined audio commands. In some examples, which predetermined audio commands a computing device is configured to act on may be determined (e.g., by the computing device) based on a number of other criteria, such as the applications installed at the computing device, the applications executing at the computing device, the time of day, or other criteria.

Before monitoring for audio input, each of audio input modules 4 may require permission from a user of the respective one of computing devices 2. That is, each of audio input modules 4 may perform operations to obtain permission from a user prior to monitoring the environment around the respective computing device for audio input and/or processing the audio input to determine whether the audio input includes a predetermined audio command. For instance, audio input module 4A may cause computing device 2A to output a prompt to the user, asking for permission to monitor audio from the environment of computing device 2A. Audio input module 4A may only monitor for audio input in response to receiving confirmation from the user. In some examples, permission to monitor environmental audio may be transient or situation specific. In other examples, permission may be more persistent. That is, in various examples, a user may grant permission to monitor environmental audio for a short duration of time, for a long duration of time, only in specific situations, and/or until the user revokes permission.

Generally, responsive to receiving permission from a user, an audio input module may monitor for audio input by obtaining audio (e.g., via a microphone or other audio sensor) from the environment around the respective computing device and processing obtained audio input to determine whether a predetermined audio command is included in the audio input. For instance, audio input modules may be operable to utilize speech recognition techniques to determine text from obtained audio input, and compare the text to a set of predetermined audio commands (e.g., stored at the respective computing device or other location) to determine whether the obtained audio input includes a predetermined audio command.

When operating independent from one another, computing devices 2 may all monitor environmental audio for audio input that includes a predetermined audio command. That is, in some examples, computing devices 2 may not be com-

5

municating with one another or otherwise associated with one another in any way and, consequently, each of audio input modules 4 may be monitoring the environment of the respective computing device for audio input that includes predetermined audio commands. If each of audio input modules 4 is monitoring a different environment, then audio input received by one of computing devices 2 may not be received by another of computing devices 2. For instance, a user may be wearing computing device 2C (the smartwatch) while in his or her kitchen. Computing device 2A (the smartphone) may be in the user's bedroom and computing device 2B (the in-vehicle stereo/navigation system) may be in the user's garage. In such instance, computing device 2C may receive audio input that includes a predetermined audio command (e.g., "Hello device") while computing devices 2A and 2B each receive audio input that does not include a predetermined audio command. Consequently, only computing device 2C may perform a corresponding action (e.g., entering an audio input mode) because only computing device 2C was monitoring the environment in which the predetermined audio command was used (e.g., the kitchen).

If, however, two or more of computing devices 2 are monitoring the same environment, then audio input received by one of computing devices 2 may be received by one or more others of computing devices 2. In such instance, two or more of computing devices 2 may each process the audio input to determine whether the audio input includes a predetermined audio command. The two or more computing devices may each determine that the audio input includes a predetermined audio command and may each perform the corresponding action. Two or more computing devices monitoring the same environment for audio input that includes a predetermined audio command may waste processing power and/or battery life through duplicative monitoring of environmental audio as well as through duplicative processing of received audio input.

Furthermore, two or more computing devices performing the same action may, in some examples, give rise to parallel-processing concerns. For instance, if two computing devices receive the predetermined audio command "Hello device," both computing devices may, in some examples, enter an audio input mode. Thereafter, any audio command included in audio input (e.g., "Play Song 1") may be received by both computing devices. In order to perform the corresponding action, both computing devices may attempt to access a "Song 1" audio file (e.g., located at a remote location) at nearly the same time. In some examples, one computing device may begin playing "Song 1" while the other returns an error. In other examples, both computing devices may begin playing "Song 1." In yet other examples, both computing devices may return an error. Other examples of parallel-processing concerns may include two or more computing devices attempting to place a phone call, two or more computing devices attempting to create or modify a calendar entry, or otherwise create, access, modify, or delete the same data. Monitoring an environment for audio input including predetermined audio commands by only a single computing device may reduce wasted energy, avoid parallel-processing problems, and overall improve the user experience.

In the example of FIG. 1, each of computing devices 2 may be associated with one another. For instance, computing devices 2 may be associated with one another based on the fact that each of computing devices 2 is associated with the same user. As another example, computing devices 2 may be associated with one another based on established communications links between computing devices 2 via one or more wireless or wired (e.g., physical) links. In some

6

examples, the connections may be established and/or maintained by communication units 4. For instance, in the example of FIG. 1, computing devices 2 may be paired or grouped through short-range communication such as the Bluetooth protocol or the Bluetooth Low Energy (BLE) protocol. In other examples, computing devices 2 may be paired, grouped, or otherwise communicatively connected to one another using another protocol. Other examples of wired or wireless connections linking two or more of computing devices 2 may include one or more connections formed via a packet-based network connection (e.g., TCP/IP, etc.) or any other means of communicating data. In some examples, each connection may function independently. For instance, computing device 2A may communicate with computing device 2B via a first connection and communicate with computing device 2C via a second connection. In some examples, all of computing devices 2 may communicate via one or more shared connections. For instance, computing devices 2 may each communicate with each other via a bus structure, a token ring structure, a star network structure, or any other structure for shared communication among two or more devices.

In some examples, when computing devices 2 are associated with one another, computing devices 2 may determine whether or not they are in physical proximity of one another. That is, one or more of computing devices 2 may determine whether computing devices 2 are close to one another. In the example of FIG. 1, for instance, in which computing devices 2 are connected via short-range communication, computing devices 2 may determine physical proximity based on the signal strength of the short-range communications. In the example of FIG. 1, the user may be wearing computing device 2C (the smartwatch) while driving in his or her car (the in-vehicle stereo/navigation system) with computing device 2A (the smartphone) in his or her pocket. Consequently, computing devices 2 may determine that each of computing devices 2 is physically proximate to the others.

In some examples, computing devices 2 may determine physical proximity based additionally or alternatively on physical location. That is, each of computing devices 2 may include a module to determine a physical location of the respective computing device (e.g., via GPS or other means) and computing devices 2 may determine that two devices are physically proximate when the two computing devices' physical locations are substantially the same (e.g., within 5 meters of one another, within 50 meters of one another, or within some other distance). In some examples, computing devices 2 may determine physical proximity by additionally or alternatively communicating via links 8. For instance, computing devices 2 may compare audio input received from the environment and, if the audio input is determined to be substantially the same, computing devices 2 may determine that they are physically proximate to one another. In some examples, computing devices 2 may determine physical proximity based additionally or alternatively on other criteria, such as the existence or non-existence of one or more wireless networks, one or more inputs received from a user, or other criteria. In some examples, computing devices 2 may perform the techniques of the present disclosure without determining physical proximity.

In accordance with the techniques described herein, one or more computing devices from a plurality of computing devices may designate a particular computing device from among the plurality of computing devices to process audio input to determine whether the audio input includes a predetermined audio command. In the example of FIG. 1, for instance, one or more of computing devices 2 (e.g.,

7

computing device 2A, computing device 2A and computing device 2B, etc.) may designate a particular one of computing devices 2 (e.g., computing device 2B) to monitor for audio input and/or process the audio input to determine whether the audio input includes a predetermined audio command. That is, computing devices 2 may designate audio input module 6B of computing device 2B to monitor environmental audio for predetermined audio commands. The designation of computing device 2B may reflect the fact that, when travelling in a vehicle, an in-vehicle stereo/navigation system may provide better microphone placement and audio reception characteristics for voice recognition than a smartphone or a smartwatch. That is, while travelling in a vehicle, it may be likely that the user's wrist will be near the steering wheel and away from the user's mouth. Furthermore, while a user's smartphone may be sufficiently placed to receive environmental audio, it may be likely that the in-vehicle navigation system is better configured to monitor audio in the vehicle and process audio input to determine whether the audio input includes a predetermined audio command, as the system may be designed to account for road noise, echo, and other characteristics unique to the in-vehicle experience.

In some examples, designation of a particular computing device to monitor for and/or process audio input may be performed by a single computing device. For instance, computing device 2A (the smartphone) may make the designation alone. In other examples, designation of a particular computing device to monitor for and/or process audio input may be jointly performed by two or more of computing devices 2. For instance, computing devices 2 may perform an arbitration process among themselves to determine which of computing devices 2 should be designated as the particular computing device. In some examples, each of computing devices 2 may independently determine which of computing devices 2 should be designated as the particular computing device to monitor for and/or process audio input.

In various examples, one or more of computing devices 2 may designate a particular computing device to monitor for and/or process audio input based on various criteria. In some examples, designation of the particular computing device may be based on a configured hierarchy of device types (e.g., "smartphone," "smartwatch," "headset," "tablet," "laptop," "desktop," "in-vehicle system," etc.). For instance, one or more of computing devices 2 may be configured with a ranked list of device types that indicates respective priorities of various device types usable to monitor for and/or process audio input. Each of computing devices 2 may communicate (e.g., using communication units 4) an indication of its respective device type to others of computing devices 2. The one or more of computing devices 2 may designate, as the particular computing device, the computing device that is of a device type that ranks highest on the list. For instance, if each of computing devices 2 is configured with the ranked list, then each respective computing device may send a respective device type and receive device types of the other computing devices. Each respective computing device may determine whether or not the respective computing device is of the device type that ranks highest on the list. If the respective computing device is of the device type that ranks highest on the list, the respective computing device may designate itself to monitor for and/or process audio input. If the respective computing device is not of the device type that ranks highest, the respective computing device may not designate itself.

In some examples, one or more of computing devices 2 may designate the particular computing device to monitor for and/or process audio input based additionally or alter-

8

natively on a level of audio input quality associated with one or more of computing devices 2. For instance, one or more of computing devices 2 may obtain audio input from the environment (e.g., via respective ones of audio input modules 6) and communicate with others of computing devices 2 to compare the quality of the audio input and/or of speech included therein. Computing devices 2 may determine and/or compare audio input quality using various quantifiers, such as volume, signal-to-noise ratio, and/or other parameters. The one or more of computing devices 2 may designate the computing device having the highest level of audio input quality as the particular computing device to monitor for and/or process audio input.

In some examples, one or more of computing devices 2 may designate the particular computing device to monitor for and/or process audio input based additionally or alternatively on the number of predetermined audio commands that each of computing devices 2 may be operable to act on. That is, in some examples, the one or more of computing devices 2 may designate the computing device configured with the largest number of predetermined audio commands as the particular computing device. This may reflect the fact that it may be more likely that the user will be causing the computing device configured with the largest number of predetermined audio commands to perform an action.

In some examples, one or more of computing devices 2 may designate the particular computing device to monitor for and/or process audio input based additionally or alternatively on a determined context of one or more of computing devices 2. The context of a computing device may be made up of any number of items of contextual information. Example items of contextual information may include collections of data (e.g., a text data structure, a numeric data structure, or other data structure) that represents a location of the computing device (e.g., a GPS location); information indicating a time as determined by the computing device; information indicating one or more applications installed at the computing device; information indicating one or more applications currently executing at the computing device; information indicating one or more networks (e.g., wireless networks) available to the computing device; data that represents one or more other computing devices in proximity to the computing device (e.g., within 10 feet, within 100 feet, or other distance); data that represents an operating mode (e.g., silent mode, airplane mode, driving mode, standby mode, low battery mode, or any other mode of operation) of the computing device; data obtained from one or more sensors of the computing device (e.g., temperature data, ambient noise level data, light level data, acceleration/movement data, image/video data, and other data), or any other data about the status or current state of the computing device.

In some examples, an item of contextual information may additionally or alternatively be data representing information about a user of the computing device, such as a name of the user, a user identification (UID) of the user, information from one or more social media network service accounts associated with the user, information from one or more calendars or scheduling applications associated with the user, information indicating one or more social or professional relationships of the user (e.g., user contacts), or any other information about the user.

As one example of designating the particular computing device to monitor for and/or process audio input based on context, one of computing devices 2 (e.g., computing device 2A) may determine an item of contextual information indicating that computing device 2A is at a particular location

and an item of contextual information indicating that the user's workplace corresponds to the particular location. Based at least in part on the determined context of computing device 2A, one or more of computing devices 2 may designate computing device 2C as the particular computing device to monitor for and process audio input. This may reflect the fact that, while a user is at work, he or she is unlikely to be near his or her vehicle. Furthermore, while at work, the user may want to speak softly and avoid holding his or her smartphone next to his or her face in order to avoid attracting attention. That is, a smartwatch may be the device closest to the user's mouth when the user is providing audio input, and thus computing devices 2 may, based on the determined context of computing device 2A, designate computing device 2C as the particular computing device to monitor for and process audio input.

Responsive to designating the particular computing device to monitor for and/or process audio input, the techniques of the present disclosure may include temporarily deactivating, for each respective computing device from the plurality of computing devices other than the particular computing device, the respective module included in the respective computing device. In the example of FIG. 1, for instance, because computing device 2B was designated as the particular computing device to monitor for and/or process audio input, audio input modules 6A and 6C may be temporarily deactivated while audio input module 6B remains active. This is illustrated in the example of FIG. 1 by showing audio input modules 6A and 6C as shaded elements having dotted lines, while audio input module 6B remains solid and unshaded. In some examples, the same computing device or devices that designate the particular computing device may temporarily deactivate the audio input modules that are included in computing devices other than the particular computing device. In some examples, the designated particular computing device may output instructions to computing devices other than the particular computing device to cause the respective computing devices other than the particular computing device to temporarily deactivate respective audio input modules. In some examples, each computing device other than the particular computing device may temporarily deactivate its own audio input module based on the designation. In other words, those of audio input modules 6 included in one of computing devices 2 other than the designated computing device (e.g., computing device 2B) may be temporarily deactivated by one or more of computing devices 2. That is, computing devices 2 may temporarily cease monitoring for and/or processing audio input. The deactivation may be temporary in that computing devices 2 may resume monitoring for and/or processing audio input at a later time (e.g., after a specified duration of time has passed, upon receiving an indication from another computing device, upon determining that none of the other of computing devices 2 are in proximity, or based on one or more other criteria).

The techniques of the present disclosure may include receiving, by the particular computing device, audio input. In the example of FIG. 1, for instance, audio input module 6B may continue to monitor the environment of computing device 2B for audio input. The user may speak a predetermined audio command (e.g., "Give me directions"). Audio input module 6B may receive the audio input (e.g., from a microphone or other audio input device) that includes the predetermined audio command. Audio input modules 6A and 6C may remain deactivated and thus not receive the audio input that includes the predetermined audio command.

In accordance with the techniques described herein, the particular computing device may process the audio input to determine whether the audio input includes a predetermined audio command. In the example of FIG. 1, audio input module 6B may process the received audio input and determine that the audio input includes the predetermined audio command, "Give me directions." Because audio input modules 6A and 6C are temporarily deactivated, neither audio input module 6A nor audio input module 6C may waste processing power and/or battery power to monitor for audio input and/or process the audio input.

In this way, the techniques of the present disclosure may enable a plurality of computing devices to more efficiently monitor for audio input and/or process received audio input to identify predetermined audio commands included in the audio input. By designating a particular computing device to monitor for and/or process audio input and temporarily ceasing monitoring and/or processing by the other computing devices, the techniques described herein may reduce wasted energy (e.g., battery power) and avoid potential parallel-processing issues. Furthermore, by designating the particular computing device to monitor for audio input and/or process audio input based on various criteria, the techniques of this disclosure may improve the user experience by improving speech recognition in various instances.

FIG. 2 is a conceptual diagram illustrating details of one example of computing device 2A configured to process audio input, in accordance with one or more techniques of the present disclosure. The example of FIG. 2 is described below within the context of FIG. 1. FIG. 2 illustrates only one particular example of computing device 2A, and many other examples of computing device 2A may be used in other instances. In some examples, computing device 2A may include fewer components than shown in the example of FIG. 2 or additional components not shown in the example of FIG. 2.

As shown in the example of FIG. 2, computing device 2A includes one or more processors 20, one or more input devices 22, and one or more output devices 24. Computing device 2 also includes one or more communication units 4A, one or more sensors 26, and one or more storage devices 28. Storage devices 28 include user interface (UI) module 32, device context module 36, application modules 34A-34N (collectively, "application modules 34"), and audio input module 6A. Audio input module 6A, in turn, includes monitoring and execution module 38 and negotiation module 40. Communication units 4A may have the same or similar functionality as that previously described with respect to FIG. 1.

Communication channels (COMM. CHANNELS) 30 may interconnect components 4A, 6A, 20, 22, 24, 26, 28, 32, 34, 36, 38, and/or 40 for inter-component communications (physically, communicatively, and/or operatively). In some examples, communication channels 30 may include a system bus, a network connection, an inter-process communication data structure, or any other method for communicating data.

One or more processors 20 may, in the example of FIG. 2, implement functionality and/or execute instructions within computing device 2A. For example, processors 20 may receive and execute instructions stored by storage devices 28 that implement the functionality of modules 6A, 32, 34, 36, 38, and/or 40. These instructions, executed by processors 20, may cause computing device 2A to read/write/etc. information stored within storage devices 28 during program execution. Processors 20 may execute instructions of modules 6A, 32, 34, 36, 38, and/or 40 to cause

11

computing device 2A to (by itself or in association with other computing devices) designate a particular computing device to monitor for and/or process audio input, receive audio input, process the audio input to determine whether the audio input includes a predetermined audio command, temporarily cease monitoring for and/or processing audio input, and/or perform other actions. That is, modules 6A, 32, 34, 36, 38, and/or 40 may be operable by processors 20 to perform various actions or functions of computing device 2A.

In the example of FIG. 2, one or more input devices 22 may be operable to receive input. Examples of input are tactile, audio, and video input. Input devices 22, in one example, include a presence-sensitive or touch-sensitive display, a mouse, a keyboard, a voice responsive system, a video camera, a microphone or other audio sensor, or any other type of device for detecting input from a human or machine. For instance, input devices 22 may be or may include a microphone, or other audio sensors operable to receive audio input from the environment around computing device 2A.

In some examples, a presence sensitive input device may detect an object at and/or near a screen. As one example range, a presence-sensitive input device may detect an object, such as a finger or stylus that is within 2 inches or less of the screen. The presence-sensitive input device may determine a location (e.g., an (x,y) coordinate) of a screen at which the object was detected. In another example range, a presence-sensitive input device may detect an object six inches or less from the screen and other ranges are also possible. The presence-sensitive input device may determine the location of the screen selected by a user's finger using capacitive, inductive, and/or optical recognition techniques. In some examples, a presence sensitive input device also provides output to a user using tactile, audio, or video stimuli as described with respect to output devices 24.

One or more output devices 24 may, in the example of FIG. 2, be operable to generate output. Examples of output are tactile, audio, and video output. Output devices 24, in one example, include a presence-sensitive display, sound card, speaker, video graphics adapter card, speaker, cathode ray tube (CRT) monitor, liquid crystal display (LCD), or any other type of device for generating output to a human or to a machine.

While illustrated as internal components of computing device 2A, one or more of input devices 22 and/or output devices 24 may also represent external components that share a data path with computing device 2A for transmitting and/or receiving input and output. That is, in some examples, input devices 22 and/or output devices 24 may represent built-in components of computing device 2A, located within and physically connected to the external packaging of computing device 2A (e.g., a screen on a mobile phone). In another example, input devices 22 and/or output devices 24 may represent external components of computing device 2A, located outside and physically separate from the packaging of computing device 2A (e.g., a monitor, a projector, or other display device that shares a wired and/or wireless data path with a tablet computer).

In the example of FIG. 2, one or more sensors 26 may be operable to generate data for use by components of computing device 2A. Sensors 26 may include any device or component capable of obtaining data about computing device 2A, data about an environment in which computing device 2A is situated, data about a user of computing device 2A, or other data. In some examples, sensors 26 may obtain data providing contextual information regarding computing

12

device 2A, such as one or more of temperature data, ambient noise level data, light level data, acceleration/movement data, or image/video data, for example. Any of sensors 26 may be hardware, firmware, software, or a combination thereof for obtaining information. Examples of sensors 26 may include a GPS sensor, a position sensor, an accelerometer or other motion sensor, a camera, a compass, a magnetometer, a light sensor, an infrared sensor, a microphone or other audio sensor, a radiation sensor, a temperature sensor, a barometer, an altimeter, or other data gathering components.

One or more storage devices 28 may, in the example of FIG. 2, be operable to store information for processing during operation of computing device 2A. For instance, computing device 2A may store data that modules 6A, 32, 36, and/or 34 may access during execution at computing device 2A. In some examples, storage devices 28 represent temporary memories, meaning that a primary purpose of storage devices 28 is not long-term storage. For instance, storage devices 28 of computing device 2A may be volatile memory, meaning that storage devices 28 may not retain stored contents if powered off. Examples of volatile memories include random access memories (RAM), dynamic random access memories (DRAM), static random access memories (SRAM), and other forms of volatile memories known in the art.

Storage devices 28, in some examples, also include one or more computer-readable storage media. Storage devices 28 may be configured to store larger amounts of information than volatile memory. Storage devices 28 may further be configured for long-term storage of information. In some examples, storage devices 28 include non-volatile storage elements, meaning that storage devices 28 may maintain information through power on/power off cycles. Examples of non-volatile memories include magnetic hard discs, optical discs, floppy discs, flash memories, or forms of electrically programmable memories (EPROM) or electrically erasable and programmable (EEPROM) memories. Storage devices 28 may, in some examples, store program instructions and/or information (e.g., data) associated with modules 6A, 32, 34, 36, 38, and/or 40, such as during program execution.

UI module 32 may, in the example of FIG. 2, may be operable by processors 20 to perform one or more functions, such as receive input (e.g., from input devices 22) and send indications of such input to other components associated with computing device 2A, such as application modules 34. UI module 32 may also receive data from components associated with computing device 2A such as modules 6A and/or 34. Using the data received, UI module 32 may cause other components associated with computing device 2A, such as output devices 24, to provide output based on the data. For instance, UI module 32 may receive data from one of application modules 34 to display a GUI at one of output devices 24.

Application modules 8, as shown in the example of FIG. 1, may include functionality to perform any variety of operations on computing device 2. For instance, application modules 8 may include a word processor application, an email application, a web browser application, a multimedia player application, an alarm clock application, a calendar application, an operating system, a distributed computing application, a graphic design application, a video editing application, a web development application, a navigation or mapping application, or any other application.

Device context module 36 may, in the example of FIG. 2, be operable by processors 20 to perform operations to

13

determine a current context of computing device 2A. That is, if given permission by the user of computing device 2A, device context module 36 may be operable to obtain information from other components of computing device 2A (e.g., application modules 34, communications units 4A, sensors 26, etc.) and/or from other computing devices in order to determine a current context that describes computing device 2A, the environment in which computing device 2A is situated, the activities in which computing device 2A is current engaged, a user of computing device 2A, the activities in which the user of computing device 2A is currently engaged, and/or other information.

As one example, device context module 36 may communicate with input devices 22, and/or UI module 32 to obtain information indicating whether or not the user is interacting with computing device 2A. Device context module 36 may communicate with one of communication units 4A (e.g., a Bluetooth radio) and obtain information indicating whether or not one or more other computing devices are proximate to computing device 2A. Device context module 36 may communicate with one or more of sensors 26 to obtain information about the physical environment in which computing device 2A exists (e.g., a light level, an ambient noise level, a temperature level, a humidity level, a location, etc.). In some examples, device context module 36 may not communicate directly with sensors 26. In some examples, device context module 36 may obtain data from modules of computing device 2A (not shown) that process sensor data received from sensors 26.

As another example, device context module 36 may communicate with one or more of application modules 34 to obtain an indication of which (if any) application modules are executing at computing device 2A, obtain an indication of any application that is currently being used by the user (e.g., which applications are being displayed at output devices 24 or are “in focus”), and/or an indication of an activity such application is performing. For instance, device context module 36 may obtain data indicating that application 34A (e.g., an audio player application) and application 34B (e.g., an email client application) are currently executing at computing device 2A, application 34B is currently in focus, and application 34A is currently playing an audio track while application 34B is providing an email-drafting user interface. Other example activities which an application may be performing include playing a game, browsing a website, searching for information, booking travel reservations, or other activities. In some examples, device context module 36 may obtain information relating to the activity that an application is performing, such as a recipient or sender of an email being composed or read, a user name or level for a game being played, a title or artist name for a song being played, a website URL being viewed, a search query being executed, or other details.

As yet other examples of information that device context module 36 may use to determine the current context, device context module 36 may communicate with application modules 34 and/or communication units 6A to determine contextual information regarding the user of computing device 2A. For instance, device context module 36 may obtain information from an email application regarding social media network services that the user is a member of, email lists that the user subscribes to, or other information. Device context module 36 may also obtain information from a calendar application regarding appointments the user may have. Device context module 36 may communicate with communication units 6A to obtain information from one or more other computing devices or systems. For instance, if

14

permission is received from the user, device context module 36 may cause communication units 6A to obtain information from a social media network service account of the user, an email account of the user, or other sources of information.

Based on the obtained items of contextual information, device context module 36 may determine a current context. As one example, device context module 36 may determine a current context by determining values for one or more contextual categories, as well as a respective weight for each category. That is, device context module 36 may determine values based on obtained information and weigh each category based on a predicted accuracy and/or a predicted importance of the associated value. For instance, contextual categories may indicate a type and/or a name of an application being executed at computing device 2A, a location of computing device 2A, an indication of a computing device or computing devices that are proximate to computing device 2A, an activity or task in which computing device 2A is currently engaging or in which an application executing at computing device 2A is engaging, a time of day or current time, a user identification number for a user of computing device 2A, a predicted activity or task in which the user is engaging, a predicted mode of travel of the user, a predicted current availability of the user, or various other information.

A weighting applied to a contextual category may indicate an importance of the value that was determined for the category. For example, weightings may be defined on a scale (e.g., 0-10, 1-100, or other scale). If a value for a category is predicted to be an important part of the current context and/or determined as very likely to be true, then the category may receive a heavy weight, toward one end of the scale (e.g., 90 out of 100). If a value for a category is not predicted to be very important and/or is determined to include uncertainty, then the category may receive a lighter weight, at the other end of the scale (e.g., 10 out of 100). In this way, device context module 36 may prioritize some information over other information when determining the current context.

In the example of FIG. 2, device context module 36 may determine a value for a current-location category. For instance, based on information (e.g., obtained from a GPS location sensor included in sensors 26) indicating that computing device 2A is regularly located at a specified location during nighttime hours and that computing device 2A is currently located at the same specified location, as well as information (e.g., obtained from contact information for the user or from an email application of the user) indicating that the user's home address is geographically located in substantially the same specified location, device context module 36 may determine the value “Home” for the current-location category.

Device context module 36 may determine a corresponding weight for the current-location category. In the example of FIG. 2, device context module 36 may determine a relatively heavy weight, as the location information (e.g., based off of GPS data) is relatively accurate, and there are multiple pieces of information indicating the same result (e.g., that computing device 2A is at the home of the user). In other examples, such as where location information is based on other, less-accurate data (e.g., only a time of day and previous location data), the value for the current-location category may be weighted less heavily.

Device context module 36 may, in the example of FIG. 2, also determine a value for a movement category that indicates whether or not computing device 2A is moving. For instance, based on information obtained from an accelerometer (e.g., one of sensors 26) and/or others inputs, device

15

context module 36 may determine the value for the movement category as “None.” That is, device context module 36 may determine that computing device 2A is not currently moving. Device context module 36 may assign a heavy weight to the movement category, as this information is unlikely to be incorrect (e.g., because the information is obtained directly from sensors 26 of computing device 2A).

In various examples, device context module 36 may determine values and weights for additional contextual categories or alternative contextual categories. In any case, after determining the current context, device context module 36 may provide at least an indication of the current context to audio input module 6A. For instance, device context module 36 may send data indicating each contextual category, the associated value determined for the category, and the associated weight determined for the category.

Audio input module 6A, as shown in the example of FIG. 2, includes monitoring and execution module 38. When active, monitoring and execution module 38 may be operable by processors 20 to process indications of audio input to determine whether the audio input includes a predetermined audio command. In some examples, monitoring and execution module 38 may perform one or more actions corresponding to a predetermined audio command included in received audio input. For instance, one or more of input devices 22 and/or sensors 26 may represent a low power or ultra-low power audio microphone capable of receiving continuous audio input. Monitoring and execution module 38 may receive the audio input from the microphone, and may process the audio input. In some examples, monitoring and execution module 38 may process the audio input by performing one or more speech-to-text transformations to obtain textual data corresponding to the received audio input. In other examples, monitoring and execution module 38 may perform other audio analysis to recognize and identify speech, such as performing frequency analysis to identify tonal characteristics or other sonic identifiers present in the audio input.

Monitoring and execution module 38 may use the processed input to determine whether or not a predetermined audio command is included in at least a portion of the received audio input. For instance, monitoring and execution module 38 may compare the processed audio input to a set of indications (e.g., text, tonal characteristics, or other sonic identifiers) corresponding to predetermined audio commands. If monitoring and execution module 38 determines that the audio input includes a predetermined audio command, monitoring and execution module 38 may, in some examples, perform the associated action.

In accordance with the techniques described herein, audio input module 6A (e.g., negotiation module 40) may, upon execution by processors 20, designate or assist in designating a particular computing device from a plurality of computing devices to monitor for and/or process audio input. In the example of FIG. 2, for instance, computing device 2A may be a smartphone, computing device 2B may be a desktop computer, and computing device 2C may be a smartwatch. Computing devices 2A, 2B, and 2C may be associated or otherwise communicatively coupled with one another. Negotiation module 40 may communicate with computing devices 2B and 2C to designate which of the three computing devices will monitor and process audio input to determine whether the audio input includes a predetermined audio command.

In some examples, audio input module 6A may cause negotiation module 40 to designate or assist in designating the particular computing device to monitor for and/or pro-

16

cess audio input in response to receiving an indication of computing devices 2B and 2C. Audio input module 6A may receive information indicating the existence of and/or association with computing devices 2B and 2C from communication units 4A, sensors 26, application modules 34, and/or other sources. As one example, audio input module 6A may receive an indication that computing devices 2B and 2C are physically proximate to computing device 2A. In some examples, audio input module 6A may receive an indication that computing devices 2B and 2C are using a wired or wireless network (e.g., a network provided by the user’s wireless router) to which computing device 2A is connected. In some examples, computing device 2A may receive an indication that computing devices 2B and 2C are associated with a user account (e.g., an email account, a social network service account, etc.) with which computing device 2A is also associated. In other words, audio input module 6A may, in some examples, receive some indication that computing devices 2A, 2B, and 2C are associated in some manner.

In some examples, negotiation module 40 may (e.g., alone or in conjunction with negotiation modules of computing devices 2B and/or 2C) designate the particular computing device to monitor for and/or process audio input based at least in part on the context received from device context module 36. For instance, because the context of computing device 2A indicates that computing device 2A is at a home location of the user and that computing device 2A is currently not moving, negotiation module 40 may be less likely to designate computing device 2A as the particular computing device. This may reflect the fact that, based on the context, it is likely the user is not carrying his or her smartphone around the house with him or her, and instead has set the smartphone down somewhere. In this example, negotiation module 40 may designate computing device 2B (e.g., the desktop computer) or computing device 2C (e.g., the smartwatch) as the particular computing device to monitor for and/or process audio input.

In some examples, negotiation module 40 may receive an indication of a determined context of computing device 2B and 2C. For instance, computing devices 2B and 2C may each include modules analogous to modules 36, 6A, 38, and/or 40 of computing device 2A. In such instance, the respective device context module of computing devices 2B and/or 2C may send an indication of a respective context to the respective audio input module, and the respective negotiation module may send an indication of the respective context to negotiation module 40 of computing device 2A. In the example of FIG. 2, the respective contexts of the smartwatch and desktop computer may indicate the device type of the respective computing device. The respective context may also indicate that neither the smartwatch nor the desktop computer is moving. Furthermore, the respective contexts of the smartwatch and desktop computer may indicate positions (e.g., GPS positions) that are within a threshold distance of each other (e.g., within 0.5 meters, within 3 feet, or other distance) but that are not within the threshold distance of computing device 2A.

Based at least in part on the respective contexts of computing devices 2A, 2B, and 2C, negotiation module 40 may (e.g., alone or in conjunction with a negotiation module of computing device 2B and/or a negotiation module of computing device 2C) designate the particular computing device to monitor for audio input and/or process audio input. In the example of FIG. 2, for instance, negotiation module 40 may, based on the contexts, designate computing device 2B (the desktop computer) as the particular computing device. This may reflect the fact that the user is likely using

17

computing device 2B (e.g., because the smartwatch is near the desktop computer) and computing device 2A is likely not near the user (e.g., because the smartwatch and desktop computer are not near the smartphone). Furthermore, the designation of computing device 2B to monitor for and/or process audio input may reflect the fact that computing device 2B is likely not a battery-powered device. That is, computing device 2B is less likely than computing devices 2A and 2C to have limited available power.

As another example, negotiation module 40 may designate the particular computing device to monitor for audio input and/or process audio input based additionally or alternatively on an audio input quality associated with one or more of computing devices 2A, 2B, and 2C. For instance, negotiation module 40 may compare the audio input quality for each of computing devices 2A, 2B, and 2C, and designate the computing device associated with the best audio input quality as the particular computing device to monitor for and process audio input. In order to compare audio input qualities, negotiation module 40 may obtain audio input from one or more of sensors 26 and determine a level of audio input quality for the audio input. In such examples, negotiation module 40 may determine the strength of the audio input (e.g., in decibels), a ratio between speech included in the audio input and other sound (e.g., background noise or static), or other measurements. In some examples, negotiation module 40 may send an indication of the determine audio input quality for computing device 2A to one or more of computing devices 2B and 2C. Negotiation module 40 may also receive indications of audio input quality determined by computing devices 2B and/or 2C. Negotiation module 40 may then compare the audio input quality of computing device 2A to that of computing devices 2B and 2C to determine which of the computing devices is associated with the best audio input quality. Negotiation module 40 may designate, as the particular computing device to monitor for and/or process audio input, the computing device that is associated with the best audio input quality (e.g., the highest audio strength, the best speech to noise ratio, etc.).

As another example, negotiation module 40 may designate the particular computing device to monitor for audio input and/or process audio input based additionally or alternatively on a configured hierarchy of device types. For instance, the configured hierarchy may rank a desktop computer higher than a smartphone or a smartwatch and thus negotiation module 40 may designate computing device 2B as the particular computing device. That is, based on the hierarchy, negotiation module 40 may be more likely to designate a desktop computer as the particular computing device, if available, and less likely to designate a smartphone or smartwatch. This may reflect the fact that a desktop computer may have more resources (e.g., power, processing power, etc.) for monitoring for audio input and/or processing audio input. In some examples, each negotiation module of computing devices 2 may operate independently to determine which of computing devices 2 is of a type having the highest device priority level and thus which computing device to designate to monitor for and/or process audio input. In some examples, the negotiation modules of computing devices 2 may communicate with one another (e.g., via communication units 4) to determine, cooperatively, which computing device is of a type having the highest device priority level.

In some examples, negotiation module 40 may designate the particular computing device to monitor for audio input and/or process audio input based on a hierarchy or ranked

18

list of device types and on a context of one or more of computing devices 2. For instance, computing device 2A may be configured with multiple configured hierarchies. A first hierarchy may rank a desktop computer higher than a smartphone or smartwatch, but only if some contextual criteria are satisfied. If the context of a desktop computer indicates that the desktop computer includes a microphone or other audio input device, the criteria may be satisfied, and negotiation module 40 may designate the desktop computer based on the first hierarchy. If the context of the desktop computer indicates that the desktop computer does not include a microphone, the criteria may not be satisfied, and negotiation module 40 may designate the particular computing device based on a different configured hierarchy or in some other way.

In the example of FIG. 2, negotiation module 40 may send an indication of the designation (e.g., via communication units 4A) to the respective negotiation module of computing devices 2B and 2C. That is, a computing device that designates (or computing devices that designate) a particular computing device from a plurality of computing devices to monitor for and/or process audio input may, in some examples, send indications of the designation to others of the plurality of computing devices. In some examples, the computing device that is designated as the particular computing device may send indications of the designation. That is, in some examples, responsive to being designated as the particular computing device, the particular computing device may send an at least an indication of the designation to each respective computing device from the plurality of computing devices other than the particular computing device. In this way, each of the plurality of computing devices may receive an indication of the designation.

Negotiation module 40, in the example of FIG. 2, may also be operable by processors 20 to temporarily deactivate a module for monitoring and/or processing audio input. For instance, responsive to designating a computing device other than computing device 2A as the particular device to monitor for and process audio input, negotiation module 40 may temporarily deactivate monitoring and execution module 38. In a similar fashion, responsive to receiving an indication that one or more others of computing devices 2 have designated one of computing devices 2 other than computing device 2A as the particular computing device, negotiation module 40 may temporarily deactivate monitoring and execution module 38. For example, computing devices 2 may each compare a device identifier included in the indication to their own device identifiers. Responsive to determining that the device identifiers do not match, the respective computing devices (e.g., negotiation modules) may each determine that the designated computing device is a different computing device, and may deactivate a respective module for monitoring for and/or processing audio input (e.g., monitoring and execution module 38). That is, negotiation module 40 may, in some examples, receive an instruction from one or more others of computing devices 2 that causes negotiation module 40 to temporarily deactivate monitoring and execution module 38. When deactivated, monitoring and execution module 38 may not monitor for audio input or process audio input to determine whether the audio input includes a predetermined audio command. That is, when deactivated, monitoring and execution module 38 may temporarily cease monitoring for predetermined audio commands. Responsive to determining that the device identifier included in the indication does match its device identifier, a computing device may not deactivate its module for moni-

19

toring and/or processing audio input, but rather will continue to monitor for audio input and/or process audio input.

In some examples, computing device 2A may designate computing device 2A as the particular computing device to monitor for and/or process audio input and/or receive an indication that one or more others of computing devices 2 have designated computing device 2A as the particular computing device. Responsive to the designation of computing device 2A as the particular computing device to monitor for audio input and process audio input, monitoring and execution module 38 may continue to monitor the environment for audio input and process the audio input to determine whether the audio input includes a predetermined audio command.

Subsequent to the designation of computing device 2A as the particular computing device, monitoring and execution module 38 may receive an indication of audio input (e.g., from one of sensors 26). Monitoring and execution module 38 may process the audio input and compare the processed input to a set of indications corresponding to predetermined audio commands to determine whether the audio input includes a predetermined audio command. In some examples, the set of indications corresponding to predetermined audio commands may represent predetermined audio commands to which computing device 2A is configured to respond. In some such examples, if monitoring and execution module 38 determines that none of the represented predetermined audio commands is included in the audio input, monitoring and execution module 38 may cause audio input module 6A to send (e.g., via communications unit 4A), to others of computing devices 2, the processed input. That is, in some examples the particular computing device may process audio input to determine whether the audio input includes a predetermined audio command to which the particular computing device is configured to respond and, if the audio input does not include a predetermined audio command to which the particular computing device is configured to respond, the particular computing device may send the processed input to others of the plurality of computing devices.

In some examples, the particular computing device may determine whether the audio input includes a predetermined audio command to which any of the plurality of computing devices is configured to respond. For instance, computing device 2A may receive an indication of predetermined audio commands to which others of computing devices 2 are configured to respond. As another example, computing device 2A may receive an indication of a device type for others of computing devices 2 and computing device 2A may be configured with a list associating various predetermined audio commands with various device types. That is, the particular computing device may be operable to determine whether the audio input includes a predetermined audio command to which the particular computing device may not be configured to respond, but to which another computing device is configured to respond.

If monitoring and execution module 38 determines that the audio input includes a predetermined audio command, monitoring and execution module 38 may, in some examples, cause computing device 2A to perform the action or operation associated with the received predetermined audio command. For instance, if monitoring and execution module 38 determines that the audio input includes a predetermined audio command to which computing device 2A is configured to respond, monitoring and execution module 38 may cause computing device 2A to perform one or more actions that are associated with the received predetermined

20

audio command. In some examples, if monitoring and execution module 38 determines that the audio input includes a predetermined audio command to which another of computing devices 2 is configured to respond, monitoring and execution module 38 may cause audio input module 6A to send, to the other computing device, an indication of the included predetermined audio command.

For instance, monitoring and execution module 38 may receive an indication of audio input and process the audio input. In some examples, the predetermined audio command may indicate that additional audio input that indicates a command will follow the predetermined audio command, and in other examples the predetermined audio command may itself represent a command.

In one example, monitoring and execution module 38 may determine that the audio input includes the predetermined audio command “what’s the time,” to which computing device 2C is configured to respond (but to which computing device 2A may not be configured to respond). Responsive to determining that the predetermined audio command “what’s the time” has been used, monitoring and execution module 38 may send an indication of the received predetermined audio command to computing device 2C, thereby causing computing device 2C to perform one or more actions or operations corresponding to the predetermined audio command, such as providing an audio output of the current time. For instance, responsive to receiving the indication of the predetermined audio command, computing device 2C may send information to an application module (e.g., a clock or alarm clock application) instructing the application module to output the audio. In this way, computing device 2A may designate a particular computing device from a plurality of computing devices to monitor for and/or process audio input and, if not designated as the particular computing device, computing device 2A may temporarily cease monitoring for and/or processing of audio input. If designated as the particular computing device, computing device 2A may not cease monitoring for and/or processing of audio input and may monitor for and receive audio input. Computing device 2A may process the audio input to determine whether the audio input includes a predetermined audio command and, if the input includes a predetermined audio command, computing device 2A may perform actions associated with the predetermined audio command and/or send an indication of the included predetermined audio command to one or more other computing devices from the plurality of computing devices.

FIG. 3 is a conceptual diagram illustrating one example of a computing device 2A configured to monitor for and/or process audio input and an example graphical user interface (GUI) 50 for configuring device type priorities, in accordance with one or more techniques of the present disclosure. For purposes of illustration only, the example of FIG. 3 is described below within the context of FIGS. 1 and 2.

In the example of FIG. 3, GUI 50 may be output for display by one of application modules 34 at one or more of output devices 24. For instance, a user configuration or settings application (e.g., application module 34C) may send information to UI module 32 to be output. UI module 32 may receive the data and cause a presence-sensitive display (e.g., output device 24A) to display GUI 50.

GUI 50, as shown in the example of FIG. 3, includes a list of ranked device type entries 52A-52G (collectively, device type entries 52). Each of device type entries 52 may correspond to a type of device. For instance, device type entry 52A may correspond to a defined device type that includes devices such as Bluetooth headsets or other head-mounted,

21

wearable computing devices. Device type entry 52A may, in some examples, correspond to a defined device type that includes in-vehicle navigation systems and/or stereo systems, such as computing device 2B. In some examples, device type entry 52C may correspond to a defined device type that includes any computing devices worn on a user's wrist, such as smartwatches. In some examples, a device type for a device may be user defined by interaction with computing device 2A (e.g., using application module 34C) and/or by interaction with each respective device. In some examples, device types may be preconfigured (e.g., by a manufacturer or retailer of the respective computing device).

In the example of FIG. 3, GUI 50 may represent the list of ranked device types after device type entries 52A-52C and 52E-52G have been added (e.g., by the user of computing device 2A, prepopulated by a device manufacturer or retailer, or in some other way). That is, in some examples, a list or ranked device types on a computing device may initially only include a device type entry for the computing device. In some examples, device type entries may be automatically added as the computing device communicates with new device types (e.g., when pairing via short range communication methods, when receiving an indication of a user's other devices, or in other ways). In some examples, device type entries may additionally or alternatively be manually added. For instance, in the example of FIG. 3, a user of computing device 2A may select graphical element 60 to cause computing device 2A to display a GUI that allows the user to add a new device type entry.

GUI 50, as shown in the example of FIG. 3, includes priority levels 54. Priority levels 54 may represent the relative priority of each of device type entries 52. For instance, device type entry 52A has a relative priority level of 1 (which may indicate a highest priority). Device type entry 52F has a relative priority level of 7. In some examples, a lower priority level may indicate that the respective device type entry indicates the type of devices that are more likely to be designated as the particular computing device to monitor for and/or process audio input in accordance with the techniques described herein. That is, priority levels 54 may, in some examples, represent a ranked ordering of how different device types may be prioritized for designation as the particular computing device to monitor for and/or process audio input. In the example of FIG. 3, for instance, audio input module 6A may designate a computing device having a "headset" device type, if included in the plurality of computing devices and/or detected by computing device 2A, as the particular computing device. If no computing device having the "headset" device type is included in the plurality of computing devices, then audio input module 6A may designate an in-vehicle stereo/navigation system, if included in the plurality of computing devices and/or detected by computing device 2A, as the particular computing device, and so on. In this way, audio input module 6A may use priority levels 54 for device type entries 52 in designating the particular computing device to monitor for and/or process indications of audio input.

In some examples, the list of ranked device types may be user configurable. In the example of FIG. 3, for instance, GUI 50 includes removal elements 56 and arrows 58. Removal elements 56 may be user-selectable to allow the user of computing device 2A to remove one or more of device type entries 52A-52C and/or 52E-52G from the list. In the example of FIG. 3, device type entry 52D may not be removable, because computing device 2A corresponds to the "smartphone" device type. That is, in some examples, a device type entry corresponding to the computing device at

22

which a list of ranked device types is being modified may not be removable. In other examples, only certain device type entries may be removable, all device type entries may be removable, or no device type entries may be removable.

Arrows 58 may, in the example of FIG. 3, be user-selectable elements that cause computing device 2A to modify one of priority levels 54 for a corresponding device type entry. For instance, selecting the up arrow (shown as un-filled in the example of FIG. 3) for device type entry 52D may cause computing device 2A to decrease the priority level of device type entry 52D from 4 to 3, and increase the priority level of device type entry 52C from 3 to 4. As a result, computing devices having a "smartphone" device type may be more likely to be designated as the particular computing device than computing devices having a "smart-watch" device type.

In some examples, the list of ranked device type entries shown in GUI 50 may be stored at computing device 2A. That is, when the list is modified, the changes may be saved at computing device 2A (e.g., at storage devices 28). In such instance, audio input module 6A of computing device 2A may designate the particular computing device to monitor for and/or process audio input. In some examples, the list of ranked device type entries shown in GUI 50 may be stored at another of computing devices 2 (e.g., computing device 2B, computing device 2C), at an external computing device (e.g., a central server or cloud computing system) or at multiple computing devices. That is, when the list is modified, the changes may be sent (e.g., via communication units 4A) from computing device 2A to one or more other computing devices. For instance, in the example of FIG. 3, the list of ranked device type entries 52 may be stored by computing device 2A and be sent by computing device 2A to each of computing devices 2B and 2C for storage. In such instance, each of computing devices 2A, 2B, and 2C may independently designate the particular computing device to monitor for and/or process audio input. That is, because each of computing devices 2 includes the list of ranked device types, each of computing devices 2 may be operable to use the list in designating the particular computing device and thus each designate the correct computing device as the particular computing device.

In some examples, each of device type entries 52 may be associated with a corresponding priority level that is different from every other device type entry. That is, priority levels 54 may be unique. In some examples, two or more of device type entries 52 may be associated with corresponding priority levels that are the same. Additionally or alternatively, in some examples, two computing devices that are associated with the same device type may be included in the plurality of computing devices. In such instance, one or more of computing devices 2 may designate the particular computing device to monitor for and/or process audio input based on an arbitration process, based on a round-robin selection, based on random selection, based on the current context of each of the computing devices, or in other ways. For instance, if computing devices 2 includes two computing devices having the same priority level, the computing device that was most recently used by the user may be designated as the particular computing device. Thus, given the list of ranked device type entries shown in GUI 50, if the user has two computing devices having a "headset" device type, the headset that has more recently been used by the user may be designated as the particular computing device to monitor for and/or process audio input.

By using a list of ranked device types, one or more computing devices may designate a particular computing

23

device from a plurality of computing devices to monitor for and/or process audio input. Thereafter, each of the plurality of computing devices other than the particular computing device may temporarily cease monitoring for and/or processing indications of audio input. The designated computing device may receive audio input, and process the audio input to determine whether the audio input includes a predetermined audio command. The list of ranked device types described in the example of FIG. 3 represents only one example criterion on which designation of the particular computing device may be made. Various alternative or additional criteria may be used in various examples, as described herein.

FIG. 4 is a flow diagram illustrating example operations of computing devices configured to monitor for and/or process audio input, in accordance with one or more techniques of the present disclosure. For purposes of illustration only, the example operations of FIG. 4 are described below within the context of FIGS. 1 and 2. In the example of FIG. 4, a plurality of computing devices (e.g., computing devices 2) may include computing device 2A, computing device 2B, and computing device 2C. Computing devices 2A-2C may be associated with one another. As one example, each of computing devices 2A-2C may be associated with the same user. As another example, each of computing devices 2A-2C may be in physical proximity to one another. As another example, each of computing devices 2A-2C may be communicatively connected to one another via one or more wired or wireless connections.

In the example of FIG. 4, one or more of computing devices 2 may designate one of computing devices 2 to monitor for audio input and/or process audio input (80). For instance, negotiation module 40 of computing device 2A may designate computing device 2B to monitor for and/or process audio input. In various examples, the computing device to monitor for and/or process audio input may be designated based at least in part on a current context of one or more of computing devices 2, based at least in part on the device types associated with each of computing devices 2, and/or on various other criteria.

Computing devices 2B and 2C (e.g., at respective negotiation modules) may, in the example of FIG. 4, receive an indication of the designated particular computing device to monitor for audio input and/or process audio input (82A, 82B). In some examples, computing devices 2B and 2C may receive the indication from computing device 2A, such as when computing devices 2 are communicatively connected via one or more communication links. In some examples, computing devices 2B and 2C may receive the indication from another computing device, such as a remote device (e.g., a server system, “cloud computing” system, etc.) with which each of computing devices 2 is configured to communicate.

In the example of FIG. 4, responsive to the designation of computing device 2B to monitor for and/or process indications of audio input, computing devices 2A and 2C may each deactivate a respective module for monitoring for and/or processing audio input (84A, 84B). For example, computing devices 2A, 2B, and 2C may each compare a device identifier included in the indication to their own device identifiers. Responsive to determining that the device identifiers do not match (as in the case of computing devices 2A and 2C), the respective computing devices may each determine that the designated computing device is a different computing device, and can deactivate a respective module for

24

monitoring for and/or processing audio input. In this way, the respective computing devices may temporarily cease processing of audio input.

Responsive to determining that the device identifier included in the indication does match its device identifier (i.e., that computing device 2B is the computing device designated to monitor for and/or process audio input), computing device 2B will not deactivate its module for monitoring for and/or processing audio input, but rather will continue to monitor for audio input and/or process audio input. That is, computing devices from the plurality of computing devices other than the designated computing device may stop monitoring for predetermined audio commands while the designated computing device may continue to monitor for predetermined audio commands. For instance, computing device 2A may deactivate monitoring and execution module 38, and computing device 2C may deactivate an analogous monitoring and execution module of computing device 2C.

Computing device 2B, in the example of FIG. 4, may receive audio input (86). That is, because computing device 2B did not deactivate its respective module for monitoring for and/or processing audio input, the respective module of computing device 2B (e.g., a monitoring and execution module) may receive an indication of audio input. Computing device 2B may process the audio input and determine that the audio input includes a predetermined audio command (88). For instance, computing device 2B may determine that the audio input includes the predetermined audio command “make a call.”

Computing device 2B may, in the example of FIG. 4, determine an action associated with the predetermined audio command that is included in the audio input (89). For instance, computing device 2B may determine an action based on the predetermined audio command itself and/or on subsequent audio input. As one example of an action determined based on the predetermined audio command itself, the predetermined audio command may be the phrase “go silent” and the action may be to enter a silent mode. That is, the action (e.g., enter silent mode) is based on the predetermined audio command (e.g., “go silent”). As an example of an action determined based on the predetermined audio command and a subsequent audio input, the predetermined audio command may be the word “play” and the subsequent audio input may be the title of a song, TV show, or other media content item. In such instance, the action (e.g., play the specified media item) may be based on both the predetermined audio command and the subsequent audio input. In the example of FIG. 4, computing device 2B may determine an action to execute a telephone application, based on the predetermined audio command, “make a call.”

In the example of FIG. 4, computing device 2B may determine a computing device to perform the action corresponding to the predetermined audio command included in the audio input (90). That is, based at least in part on the predetermined audio command and/or an associated action, computing device 2B may determine which of computing devices 2 should perform the action. In some examples, determining a computing device to perform an action associated with a predetermined audio command may be based on a current context of one or more of computing devices 2. For instance, computing device 2B may determine which of computing devices 2 is operable to perform the corresponding action, respective power sources of each of computing devices 2, or other contextual information. In some examples, determining the computing device to perform the action may be based on additional or alternative criteria. In

25

the example of FIG. 4, computing device 2B may determine that computing device 2A (e.g., the smartphone) will perform the action associated with the predetermined audio command (e.g., “make a call”).

Computing device 2B may, in the example of FIG. 4, send an indication of the action to be performed to computing device 2A (92). That is, after determining which computing device will perform an action, the designated computing device may send an indication of the action to the determined computing device. The determined computing device may receive the indication of the action to be performed (94) and may perform the action (96). In this way, a computing device designated to monitor audio input and/or process indications of audio input does not have to be the computing device to perform actions associated with predetermined audio commands that are included in received audio input. That is, a computing device that monitors for predetermined audio commands may be operable in accordance with the techniques described herein to cause another computing device from the plurality of computing devices to perform actions corresponding to a received command.

FIG. 5 is a flow diagram illustrating example operations of a computing device configured to monitor for and/or process audio input, in accordance with one or more techniques of the present disclosure. For purposes of illustration only, the example operations of FIG. 5 are described below within the context of FIGS. 1 and 2. In the example of FIG. 5, computing device 2A may be one of a plurality of computing devices (e.g., computing devices 2). Each respective computing device from the plurality of computing devices may be configured to process audio input to determine whether the audio input includes a predetermined audio command.

Computing device 2A (e.g., negotiation module 40 of audio input module 6A) may, in the example of FIG. 5, designate a particular computing device from a plurality of computing devices to process audio input (100). In some examples, computing device 2A may perform the designation alone. In some examples, computing device 2A may communicate with others of computing devices 2 to perform the designation. In some examples, two or more of computing devices 2 may independently perform the designation.

Responsive to designating another computing device from the plurality of computing devices as the particular computing device (“NO” branch of operation 102), computing device 2A may, in the example of FIG. 5, temporarily cease processing of audio input (104). That is, if computing device 2A determines that it was not designated as the particular computing device, computing device 2A may temporarily deactivate audio input module 6A. In other words, if the particular computing device is not computing device 2A, computing device 2A may cease processing of audio input. In some examples, computing device 2A may additionally send an instruction to each of computing devices 2 other than the particular computing device that causes the computing devices to temporarily cease processing of audio input.

In the example of FIG. 5, responsive to designating computing device 2A as the particular computing device (“YES” branch of operation 102), computing device 2A may receive first audio input (106). That is, computing device 2A may determine that it is the designated particular computing device. In other words, if the particular computing device is computing device 2A, computing device 2A may receive first audio input. Computing device 2A may process the first audio input to determine whether the first audio input includes a predetermined audio command (108).

26

The example operations of FIG. 5 may further be described by one or more of the examples below.

EXAMPLE 1

A method comprising: designating, by a first computing device from a plurality of computing devices, a particular computing device from the plurality of computing devices to process audio input; if the particular computing device is not the first computing device, ceasing, by the first computing device, processing of audio input; and if the particular computing device is the first computing device: receiving, by the first computing device, first audio input; and processing, by the first computing device, the first audio input to determine whether the first audio input includes a predetermined audio command.

EXAMPLE 2

The method of example 1, further comprising: if the particular computing device is the first computing device, sending, by the first computing device and to a second computing device from the plurality of computing devices, an instruction to cause the second computing device to cease processing of audio input.

EXAMPLE 3

The method of any of examples 1-2, further comprising: determining, by the first computing device, a current context of a second computing device from the plurality of computing devices, wherein designating the particular computing device to process audio input is based at least in part on the current context of the second computing device.

EXAMPLE 4

The method of example 3, wherein determining the current context of the second computing device comprises determining one or more of: a location of the second computing device, a current time as defined by the second computing device, one or more applications installed at the second computing device, one or more applications currently executing at the second computing device, one or more networks available to the second computing device, one or more other computing devices in proximity to the second computing device, an operating mode of the second computing device, an ambient temperature of the location of the second computing device, an ambient noise level of the location of the second computing device, an ambient light level of the location of the second computing device, a movement of the second computing device, a name of a user of the second computing device, a user identification (UID) of the user, a social media network service account associated with the user, one or more calendars associated with the user, or one or more social relationships of the user.

EXAMPLE 5

The method of any of examples 1-4, further comprising: responsive to determining that the first audio input includes the predetermined audio command, determining, by the first computing device, an action associated with the predetermined audio command; and determining, by the first computing device and based at least in part on the action

27

associated with the predetermined audio command, a second computing device to perform the action.

EXAMPLE 6

The method of example 5, wherein the first computing device is different from the second computing device, the method further comprising: sending, by the first computing device and to the second computing device, an indication of the action.

EXAMPLE 7

The method of any of examples 1-6, further comprising determining, by the first computing device, an audio input quality associated with a second computing device from the plurality of computing devices, wherein designating the particular computing device to process audio input is based at least in part on the audio input quality associated with the second computing device.

EXAMPLE 8

The method of any of examples 1-7, wherein designating the particular computing device to process audio input comprises designating the particular computing device to process audio input based at least in part on a configured hierarchy of two or more device types.

EXAMPLE 9

The method of example 8, wherein the configured hierarchy of two or more device types is configured based at least in part on user input.

EXAMPLE 10

The method of any of examples 1-9, wherein designating the particular computing device to process audio input comprises: receiving, by the first computing device and from a second computing device, an indication of a device type associated with the second computing device.

EXAMPLE 11

A computing device comprising: at least one processor; and at least one module operable by the at least one processor to: designate a particular computing device from a plurality of computing devices to process audio input, wherein the computing device comprises a first computing device from the plurality of computing devices, if the particular computing device is not the first computing device, cease processing of audio input, and if the particular computing device is the first computing device: receive first audio input; and process the first audio input to determine whether the first audio input includes a predetermined audio command.

EXAMPLE 12

The computing device of example 11, wherein the at least one module operable to designate the particular computing device to process audio input is operable by the at least one processor to: receive, from a second computing device from the plurality of computing devices, an indication of a device type associated with the second computing device, and designate the particular computing device to process audio

28

input based at least in part on the device type associated with the second computing device.

EXAMPLE 13

The computing device of example 11, wherein the at least one module is further operable by the at least one processor to determine a current context of a second computing device from the plurality of computing devices, wherein the at least one module operable to designate the particular computing device to process audio input is operable by the at least one processor to designate the particular computing device based at least in part on the current context of the second computing device.

EXAMPLE 14

The computing device of example 13, wherein the at least one module operable to determine the current context of the second computing device is operable by the at least one processor to determine one or more of: a location of the second computing device, a current time as defined by the second computing device, one or more applications installed at the second computing device, one or more applications currently executing at the second computing device, one or more networks available to the second computing device, one or more other computing devices in proximity to the second computing device, an operating mode of the second computing device, an ambient temperature of the location of the second computing device, an ambient noise level of the location of the second computing device, an ambient light level of the location of the second computing device, a movement of the second computing device, a name of a user of the second computing device, a user identification (UID) of the user, a social media network service account associated with the user, one or more calendars associated with the user, or one or more social relationships of the user.

EXAMPLE 15

The computing device of example 11, wherein the at least one module is further operable by the at least one processor to: responsive to determining that the first audio input includes the predetermined audio command, determine an action associated with the predetermined audio input; and determine, based at least in part on the action associated with the predetermined audio command, a second computing device to perform the action.

EXAMPLE 16

The computing device of example 15, wherein the first computing device is different from the second computing device, and wherein the at least one module is further operable by the at least one processor to: send, to the second computing device, an indication of the action.

EXAMPLE 17

The computing device of example 11, wherein the at least one module operable to designate the particular computing device to process audio input is operable by the at least one processor to designate the particular computing device to process audio input based at least in part on a configured hierarchy of two or more device types.

29

EXAMPLE 18

The computing device of any of examples 11-17, further comprising means for performing any of the method of examples 1-10.

EXAMPLE 19

A computer-readable storage medium encoded with instructions that, when executed, cause at least one processor of a first computing device to: receive, from a second computing device, a designation of a particular computing device from a plurality of computing devices to process audio input; if the particular computing device is not the first computing device, cease processing of audio input; and if the particular computing device is the first computing device: receive first audio input, and process the first audio input to determine whether the first audio input includes the predetermined audio command.

EXAMPLE 20

The computer-readable storage medium of example 19, further encoded with instructions that, when executed, cause the at least one processor to: determine a current context of the first computing device, and send, to the second computing device, an indication of the current context of the first computing device, wherein receiving the designation of the particular computing device comprises receiving the designation responsive to sending the indication of the current context.

EXAMPLE 21

The computer-readable storage medium of example 19, further encoded with instructions that, when executed, cause the at least one processor to: receive, from a third computing device from the plurality of computing devices, an indication of a device type associated with the third computing device, and send, to the second computing device, the indication of the device type associated with the third computing device, wherein receiving the designation of the particular computing device comprises receiving the designation responsive to sending the indication of the device type associated with the third computing device.

EXAMPLE 22

The computer-readable storage medium of any of examples 19-21, further encoded with instructions that, when executed, cause the at least one processor to perform the method recited by any of examples 1-10.

In one or more examples, the functions described may be implemented in hardware, software, firmware, or any combination thereof. If implemented in software, the functions may be stored on or transmitted over, as one or more instructions or code, a computer-readable medium and executed by a hardware-based processing unit. Computer-readable media may include computer-readable storage media, which corresponds to a tangible medium such as data storage media, or communication media, which includes any medium that facilitates transfer of a computer program from one place to another, e.g., according to a communication protocol. In this manner, computer-readable media generally may correspond to (1) tangible computer-readable storage media, which is non-transitory or (2) a communication medium such as a signal or carrier wave. Data storage media

30

may be any available media that can be accessed by one or more computers or one or more processors to retrieve instructions, code and/or data structures for implementation of the techniques described in this disclosure. A computer program product may include a computer-readable storage medium.

By way of example, and not limitation, such computer-readable storage media can comprise RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage, or other magnetic storage devices, flash memory, or any other medium that can be used to store desired program code in the form of instructions or data structures and that can be accessed by a computer. Also, any connection is properly termed a computer-readable medium. For example, if instructions are transmitted from a website, server, or other remote source using a coaxial cable, fiber optic cable, twisted pair, digital subscriber line (DSL), or wireless technologies such as infrared, radio, and microwave, then the coaxial cable, fiber optic cable, twisted pair, DSL, or wireless technologies such as infrared, radio, and microwave are included in the definition of medium. It should be understood, however, that computer-readable storage media and data storage media do not include connections, carrier waves, signals, or other transient media, but are instead directed to non-transient, tangible storage media. Disk and disc, as used herein, includes compact disc (CD), laser disc, optical disc, digital versatile disc (DVD), floppy disk and Blu-ray disc, where disks usually reproduce data magnetically, while discs reproduce data optically with lasers. Combinations of the above should also be included within the scope of computer-readable media.

Instructions may be executed by one or more processors, such as one or more digital signal processors (DSPs), general purpose microprocessors, application specific integrated circuits (ASICs), field programmable logic arrays (FPGAs), or other equivalent integrated or discrete logic circuitry. Accordingly, the term "processor," as used herein may refer to any of the foregoing structure or any other structure suitable for implementation of the techniques described herein. In addition, in some aspects, the functionality described herein may be provided within dedicated hardware and/or software modules. Also, the techniques could be fully implemented in one or more circuits or logic elements.

The techniques of this disclosure may be implemented in a wide variety of devices or apparatuses, including a wireless handset, an integrated circuit (IC) or a set of ICs (e.g., a chip set). Various components, modules, or units are described in this disclosure to emphasize functional aspects of devices configured to perform the disclosed techniques, but do not necessarily require realization by different hardware units. Rather, as described above, various units may be combined in a hardware unit or provided by a collection of interoperable hardware units, including one or more processors as described above, in conjunction with suitable software and/or firmware.

Various examples have been described. These and other examples are within the scope of the following claims.

What is claimed is:

1. A method comprising:

receiving, by a first computing device of a plurality of computing devices in physical proximity to one another, a spoken audio input that is also received by one or more additional computing devices of the plurality of computing devices;

determining, by the first computing device and based at least in part on information received from the one or

31

more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input; and

responsive to determining that the first computing device should perform speech recognition on the spoken audio input:

sending, by the first computing device and to at least a second computing device from the plurality of computing devices, a command to temporarily refrain from performing speech recognition;

performing, by the first computing device, speech recognition on the spoken audio input;

determining, by the first computing device, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a predetermined spoken audio command; and

responsive to determining that the spoken audio input includes the predetermined spoken audio command, executing, by the first computing device, an action associated with the predetermined spoken audio command.

2. The method of claim 1, further comprising:

determining, by the first computing device, a current context of the first computing device,

wherein determining whether the first computing device should perform speech recognition on the spoken audio input is further based on the current context of the first computing device.

3. The method of claim 2, wherein determining the current context of the first computing device comprises determining one or more of: a location of the first computing device, a current time as defined by the first computing device, one or more applications installed at the sec-end first computing device, one or more applications currently executing at the first computing device, one or more networks available to the first computing device, one or more other computing devices in proximity to the first computing device, an operating mode of the first computing device, an ambient temperature of the location of the first computing device, an ambient noise level of the location of the first computing device, an ambient light level of the location of the first computing device, a movement of the first computing device, a name of a user of the first computing device, a user identification (UID) of the user of the first computing device, a social media network service account associated with the user of the first computing device, one or more calendars associated with the user, or one or more social relationships of the user of the first computing device.

4. The method of claim 1,

wherein the information received from the one or more additional computing devices includes an indication of an audio input quality associated with a second computing device from one or more additional computing devices, and

wherein determining whether the first computing device should perform speech recognition on the spoken audio input is further based on the audio input quality associated with the second computing device and an audio input quality associated with the first computing device.

5. The method of claim 1,

wherein the information received from the one or more additional computing devices includes an indication of a type of a second computing device from the one or more additional computing devices, and

wherein determining whether the first computing device should perform speech recognition on the spoken audio

32

input is further based on a configured hierarchy of two or more device types, a type of the first computing device, and the type of the second computing device.

6. The method of claim 5, wherein the configured hierarchy of two or more device types is configured based at least in part on user input.

7. A first computing device comprising:

at least one processor; and

at least one module operable by the at least one processor to:

receive a spoken audio input that is also received by one or more additional computing devices in physical proximity to the first computing device;

determine, based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input; and

responsive to determining that the first computing device should perform speech recognition on the spoken audio input:

send, to at least a second computing device from a plurality of computing devices, a command to temporarily refrain from performing speech recognition;

perform speech recognition on the spoken audio input;

determine, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a predetermined spoken audio command; and

responsive to determining that the spoken audio input includes the predetermined spoken audio command, execute an action associated with the predetermined spoken audio command.

8. The first computing device of claim 7,

wherein the information received from the one or more additional computing devices includes an indication of a type of a second computing device from the one or more additional computing devices, and

wherein the at least one module is operable by the at least one processor to

determine whether the first computing device should perform speech recognition on the spoken audio input further based on a configured hierarchy of two or more device types, a type of the first computing device, and the type of the second computing device.

9. The computing device of claim 7,

wherein the at least one module is further operable by the at least one processor to determine a current context of the first computing device, and

wherein the at least one module is operable by the at least one processor to determine whether the first computing device should perform speech recognition on the spoken audio input further based on the current context of the first computing device.

10. The computing device of claim 9, wherein the at least one module operable to determine the current context of the first computing device is operable by the at least one processor to determine one or more of: a location of the first computing device, a current time as defined by the first computing device, one or more applications installed at the first computing device, one or more applications currently executing at the first computing device, one or more networks available to the first computing device, one or more other computing devices in proximity to the first computing device, an operating mode of the first computing device, an

33

ambient temperature of the location of the first computing device, an ambient noise level of the location of the first computing device, an ambient light level of the location of the first computing device, a movement of the first computing device, a name of a user of the first computing device, a user identification (UID) of the user of the first computing device, a social media network service account associated with the user of the first computing device, one or more calendars associated with the user of the first computing device, or one or more social relationships of the user of the first computing device.

11. A non-transitory computer-readable storage medium encoded with instructions that, when executed, cause at least one processor of a first computing device to:

receive a spoken audio input that is also received by one or more additional computing devices in physical proximity to the first computing device;

determine, based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input; and responsive to determining that the first computing device should perform speech recognition on the spoken audio input:

send, to at least a second computing device from a plurality of computing devices, a command to temporarily refrain from performing speech recognition; perform speech recognition on the spoken audio input; determine, based on performing speech recognition on the spoken audio input, whether the first audio input includes a predetermined spoken audio command; and responsive to determining that the spoken audio input includes the predetermined spoken audio command, execute an action associated with the predetermined spoken audio command.

12. The non-transitory computer-readable storage medium of claim 11, further encoded with instructions that, when executed, cause the at least one processor to:

determine a current context of the first computing device, wherein the instructions cause the at least one processor to determine whether the first computing device should perform speech recognition on the spoken audio input further based on the current context of the first computing device.

13. The non-transitory computer-readable storage medium of claim 11, wherein the information received from the one or more additional computing devices includes an indication of a type of a second computing device from the one or more additional computing devices, and

wherein the instructions cause the at least one processor to

determine whether the first computing device should perform speech recognition on the spoken audio input further based on a configured hierarchy of two or more device types, a type of the first computing device, and the type of the second computing device.

14. The non-transitory computer-readable storage medium of claim 11,

wherein the information received from the one or more additional computing devices includes an indication of audio input quality associated with a second computing device from the one or more additional computing devices, and

wherein the instructions cause the at least one processor to determine whether the first computing device should perform speech recognition on the spoken audio input

34

further based on the audio input quality associated with the second computing device and an audio input quality associated with the first computing device.

15. The first computing device of claim 7, wherein the information received from the one or more additional computing devices includes an indication of an audio input quality associated with a second computing device from the one or more additional computing devices, and

wherein the at least one module is operable by the at least one processor to determine whether the first computing device should perform speech recognition on the spoken audio input further based at least in part on the audio input quality associated with the second computing device and an audio input quality associated with the first computing device.

16. The first computing device of claim 7, wherein the at least one module is further operable by the at least one processor to:

determine, based on a network connection of the first computing device and a respective network connection of the one or more additional computing devices, that the one or more additional computing devices are in proximity to the first computing device,

wherein the at least one module is operable by the at least one processor to determine whether the first computing device should perform speech recognition on the spoken audio input in response to determining that the one or more additional computing devices are in proximity to the first computing device.

17. The first computing device of claim 7, wherein the at least one module is further operable by the at least one processor to:

determine, based on a signal strength of a wireless communication between the first computing device and the one or more additional computing devices, that the one or more additional computing devices are in proximity to the first computing device,

wherein the at least one module is operable by the at least one processor to determine whether the first computing device should perform speech recognition on the spoken audio input in response to determining that the one or more additional computing devices are in proximity to the first computing device.

18. The method of claim 1, further comprising:

determining, by the first computing device, based on a network connection of the first computing device and a respective network connection of the one or more additional computing devices, that the one or more additional computing devices are in proximity to the first computing device,

wherein determining whether the first computing device should perform speech recognition on the spoken audio input is responsive to determining that the one or more additional computing devices are in proximity to the first computing device.

19. The method of claim 1, further comprising:

determining, by the first computing device, based on a signal strength of a wireless communication between the first computing device and the one or more additional computing devices, that the one or more additional computing devices are in proximity to the first computing device,

wherein determining whether the first computing device should perform speech recognition on the spoken audio

input is responsive to determining that the one or more additional computing devices are in proximity to the first computing device.

20. The method of claim 1, further comprising:
determining, by the first computing device, based on 5
location information from a GPS sensor of the first
computing device and location information from
respective GPS sensors of the one or more additional
computing devices, that the one or more additional
computing devices are within a threshold distance of 10
the first computing device; and
responsive to determining that the one or more additional
computing devices are within a threshold distance of
the first computing device, determining, by the first
computing device, that the one or more additional 15
computing devices are in physical proximity to the first
computing device,
wherein determining whether the first computing device
should perform speech recognition on the spoken audio
input is responsive to determining that the one or more 20
additional computing devices are in proximity to the
first computing device.

* * * * *

Exhibit 2

(12) **United States Patent**
Mixer et al.

(10) **Patent No.:** **US 9,812,128 B2**
(45) **Date of Patent:** **Nov. 7, 2017**

(54) **DEVICE LEADERSHIP NEGOTIATION
AMONG VOICE INTERFACE DEVICES**

(71) Applicant: **GOOGLE INC.**, Mountain View, CA
(US)

(72) Inventors: **Kenneth Mixer**, Los Altos Hills, CA
(US); **Diego Melendo Casado**,
Mountain View, CA (US); **Alexander**
Houston Gruenstein, Mountain View,
CA (US); **Terry Tai**, New York, NY
(US); **Christopher Thaddeus Hughes**,
Redwood City, CA (US); **Matthew**
Nirvan Sharifi, Kilchberg (CH)

(73) Assignee: **GOOGLE INC.**, Mountain View, CA
(US)

(*) Notice: Subject to any disclaimer, the term of this
patent is extended or adjusted under 35
U.S.C. 154(b) by 0 days.

(21) Appl. No.: **15/284,483**

(22) Filed: **Oct. 3, 2016**

(65) **Prior Publication Data**

US 2017/0025124 A1 Jan. 26, 2017

Related U.S. Application Data

(63) Continuation-in-part of application No. 15/088,477,
filed on Apr. 1, 2016, now Pat. No. 9,514,752, which
(Continued)

(51) **Int. Cl.**
G10L 15/00 (2013.01)
G10L 15/22 (2006.01)
(Continued)

(52) **U.S. Cl.**
CPC **G10L 15/22** (2013.01); **G10L 15/32**
(2013.01); **G10L 25/60** (2013.01); **G10L**
2015/088 (2013.01); **G10L 2015/223** (2013.01)

(58) **Field of Classification Search**

USPC 704/231–257, 270–275
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,363,102 A 12/1982 Holmgren
5,659,665 A 8/1997 Whelpley, Jr.
(Continued)

FOREIGN PATENT DOCUMENTS

WO WO/9840875 A1 9/1998

OTHER PUBLICATIONS

International Search Report and Written Opinion in International
App. No. PCT/US2015/052860, dated Dec. 8, 2015, 12 pgs.

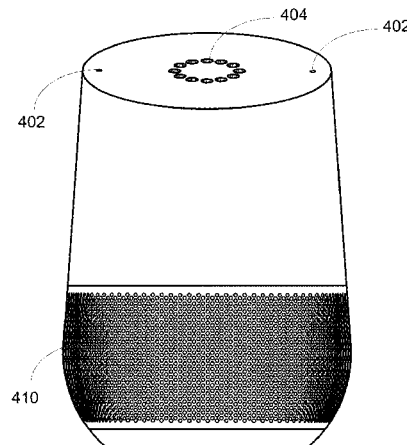
Primary Examiner — Jesse Pullias

(74) *Attorney, Agent, or Firm* — Morgan, Lewis &
Bockius LLP

(57) **ABSTRACT**

A method at a first electronic device of multiple electronic
devices, each electronic device of the plurality of electronic
devices including one or more microphones and a speaker,
includes detecting a voice input; determining a quality score
for the detected voice input; communicating the quality
score to the other devices of the plurality of electronic
devices; receiving quality scores generated by the other
devices for detection of the voice input by the other devices;
if the quality score generated by the first electronic device is
the highest amongst the quality scores, outputting an audible
and/or visual response to the detected voice input, where the
other devices of the plurality of electronic devices forgo
outputting an audible response to the detected voice input;
and if the quality score generated by the first electronic
device is not the highest amongst the quality scores, forgoing
outputting a response to the detected voice input.

15 Claims, 10 Drawing Sheets
(1 of 10 Drawing Sheet(s) Filed in Color)



Related U.S. Application Data

is a continuation of application No. 14/675,932, filed on Apr. 1, 2015, now Pat. No. 9,318,107.

(60) Provisional application No. 62/061,830, filed on Oct. 9, 2014.

(51) **Int. Cl.**

G10L 15/32 (2013.01)
G10L 15/08 (2006.01)
G10L 25/60 (2013.01)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,897,616 A	4/1999	Kanevsky	
6,671,672 B1 *	12/2003	Heck	G10L 17/24
			704/273
6,744,860 B1	6/2004	Schrage	
6,826,159 B1	11/2004	Shaffer et al.	
6,931,375 B1	8/2005	Bossemeyer	
6,973,426 B1	12/2005	Schier	
7,016,833 B2	3/2006	Gable	
7,222,072 B2 *	5/2007	Chang	G10L 17/22
			704/249
7,720,012 B1	5/2010	Borah	
7,904,297 B2 *	3/2011	Mirkovic	G06F 17/28
			704/235
8,099,288 B2	1/2012	Zhang	
8,209,174 B2	6/2012	Al-Telmissani	
8,340,975 B1 *	12/2012	Rosenberger	G10L 15/22
			704/270
8,717,949 B2	5/2014	Crinon	
8,719,018 B2	5/2014	Dinerstein	
8,768,687 B1 *	7/2014	Quasthoff	G10L 15/26
			379/142.14
8,775,191 B1	7/2014	Sharifi et al.	
8,996,372 B1 *	3/2015	Secker-Walker	G10L 15/34
			704/231
9,142,218 B2 *	9/2015	Schroeter	G10L 17/20
2002/0049596 A1 *	4/2002	Burchard	G10L 15/22
			704/270
2002/0072905 A1 *	6/2002	White	G10L 15/30
			704/231
2002/0123890 A1	9/2002	Kopp et al.	
2002/0193991 A1 *	12/2002	Bennett	G10L 15/32
			704/247

2003/0200090 A1	10/2003	Kawazoe	
2003/0231746 A1	12/2003	Hunter	
2004/0101112 A1 *	5/2004	Kuo	H04M 1/67
			379/88.01
2005/0165607 A1 *	7/2005	Di Fabbri	G06F 17/2785
			704/256
2006/0074656 A1 *	4/2006	Mathias	G10L 15/063
			704/243
2006/0184370 A1 *	8/2006	Kwak	G10L 15/1822
			704/275
2007/0100620 A1 *	5/2007	Tavares	G10L 17/06
			704/246
2007/0198262 A1	8/2007	Mindlin	
2009/0258333 A1	10/2009	Yu	
2009/0292541 A1 *	11/2009	Daya	G10L 15/063
			704/251
2010/0070276 A1 *	3/2010	Wasserblat	H04M 3/5232
			704/243
2010/0110834 A1 *	5/2010	Kim	G01S 3/8083
			367/124
2011/0026722 A1	2/2011	Jing et al.	
2011/0054892 A1	3/2011	Jung et al.	
2011/0060587 A1 *	3/2011	Phillips	G10L 15/30
			704/235
2011/0066429 A1	3/2011	Shperling et al.	
2012/0084087 A1	4/2012	Yang	
2012/0232896 A1 *	9/2012	Taleb	G10L 25/78
			704/233
2012/0265528 A1 *	10/2012	Gruber	G10L 15/18
			704/235
2013/0060571 A1 *	3/2013	Soemo	G10L 15/30
			704/251
2013/0132086 A1 *	5/2013	Xu	G10L 15/01
			704/257
2013/0183944 A1 *	7/2013	Mozer	H04L 12/282
			455/414.1
2014/0012573 A1 *	1/2014	Hung	G06F 1/3215
			704/233
2014/0088961 A1 *	3/2014	Woodward	G10L 15/22
			704/235
2014/0222430 A1	8/2014	Rao	
2014/0257821 A1 *	9/2014	Adams	G10L 25/93
			704/275
2014/0278383 A1	9/2014	Fan	
2014/0278435 A1 *	9/2014	Ganong, III	G10L 15/22
			704/275
2015/0154953 A1	6/2015	Bapat et al.	
2015/0262577 A1	9/2015	Nomura	

* cited by examiner

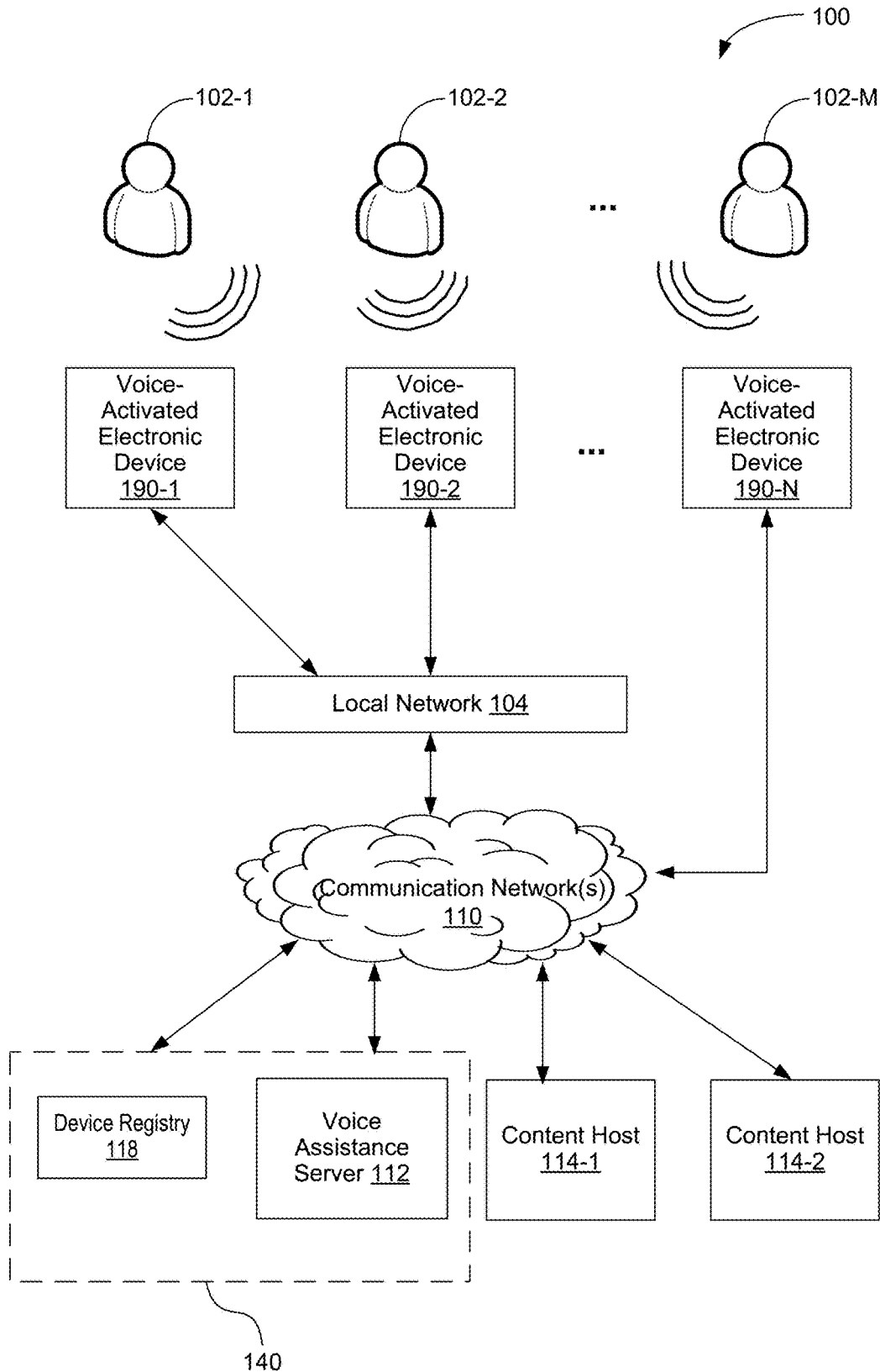


Figure 1

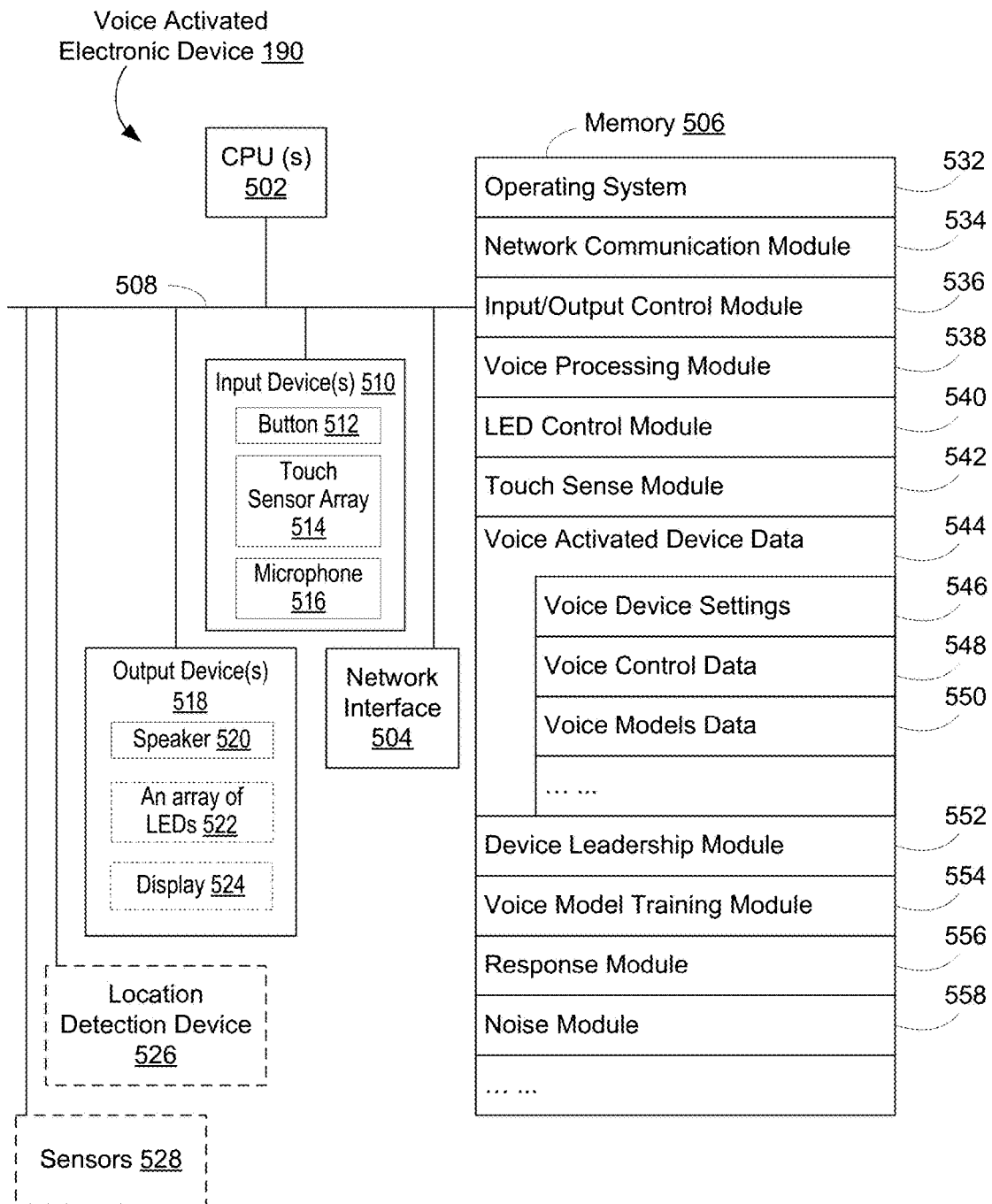


Figure 2A

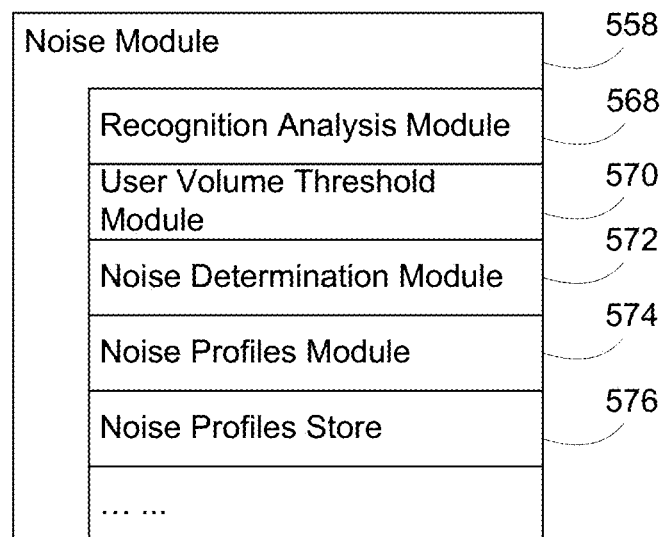
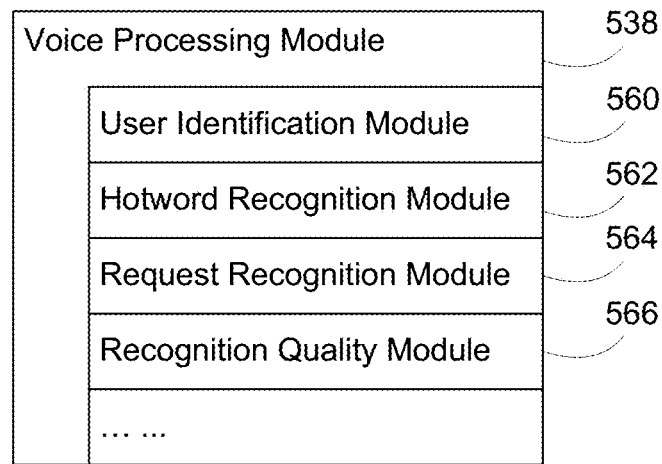


Figure 2B

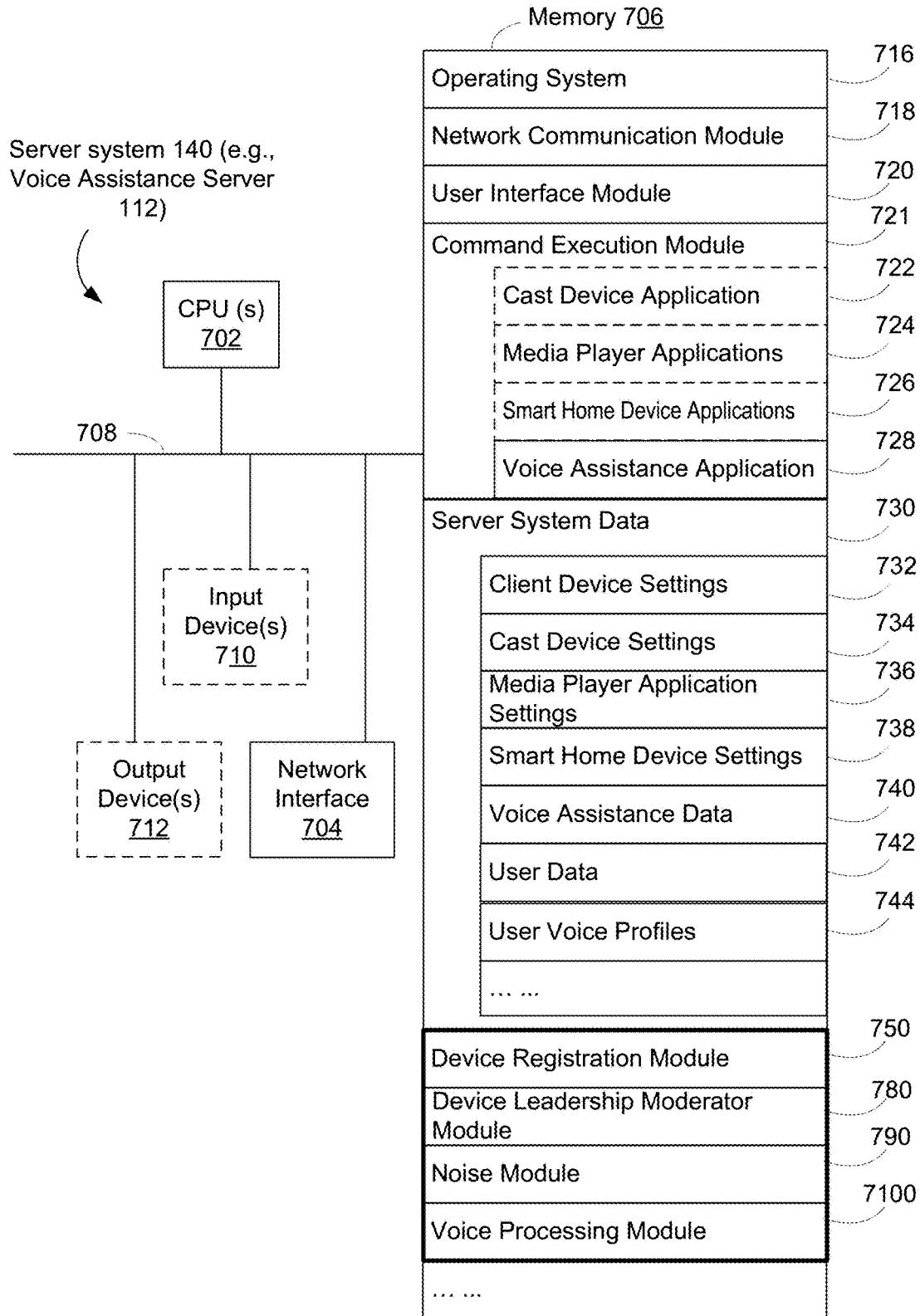


Figure 3

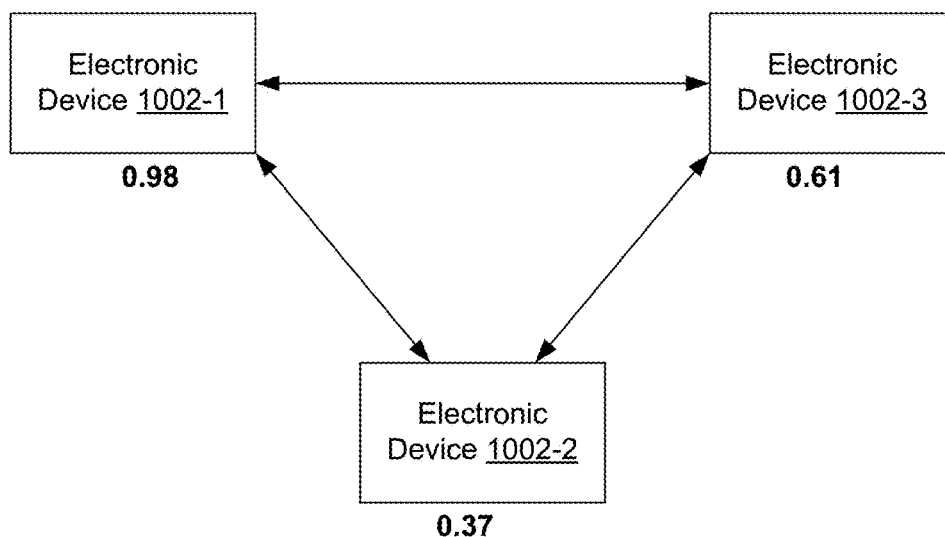


Figure 4A

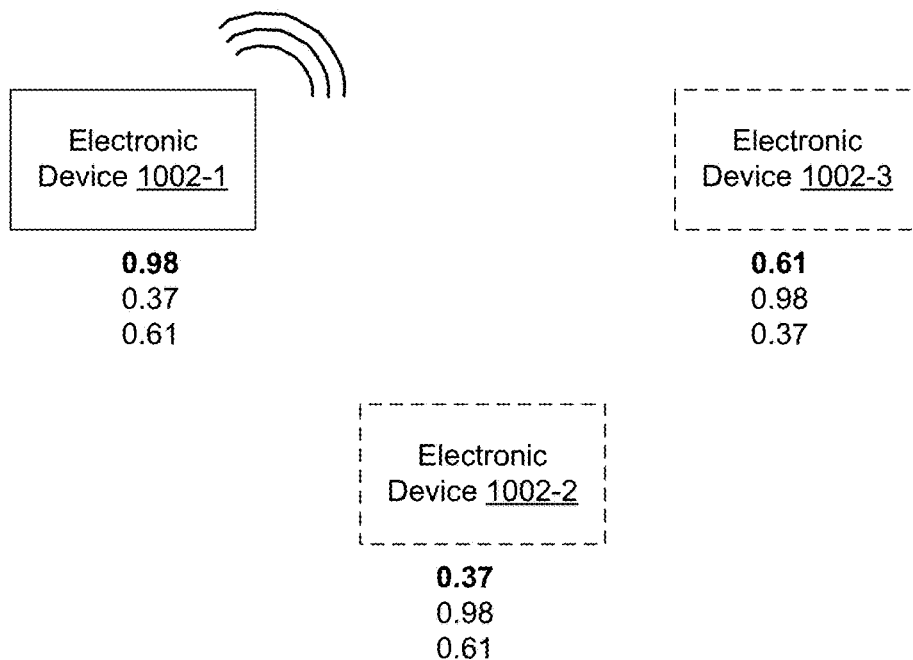


Figure 4B

1100



At an electronic device with one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors:

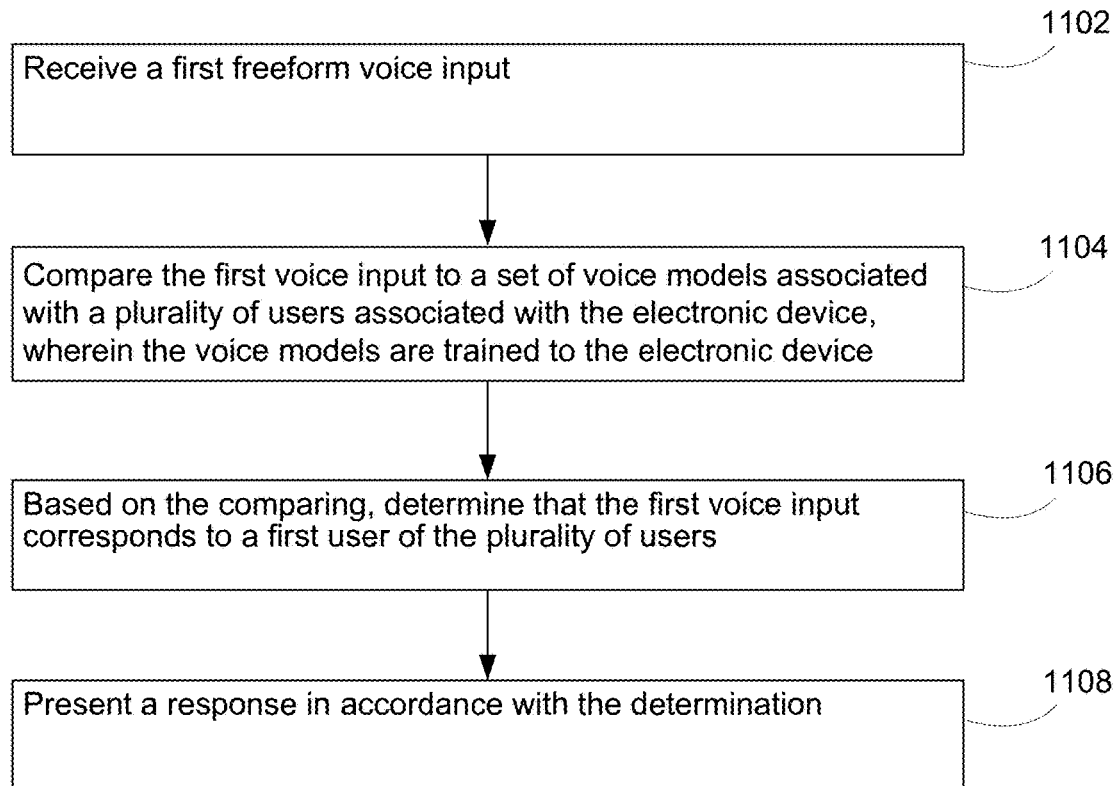
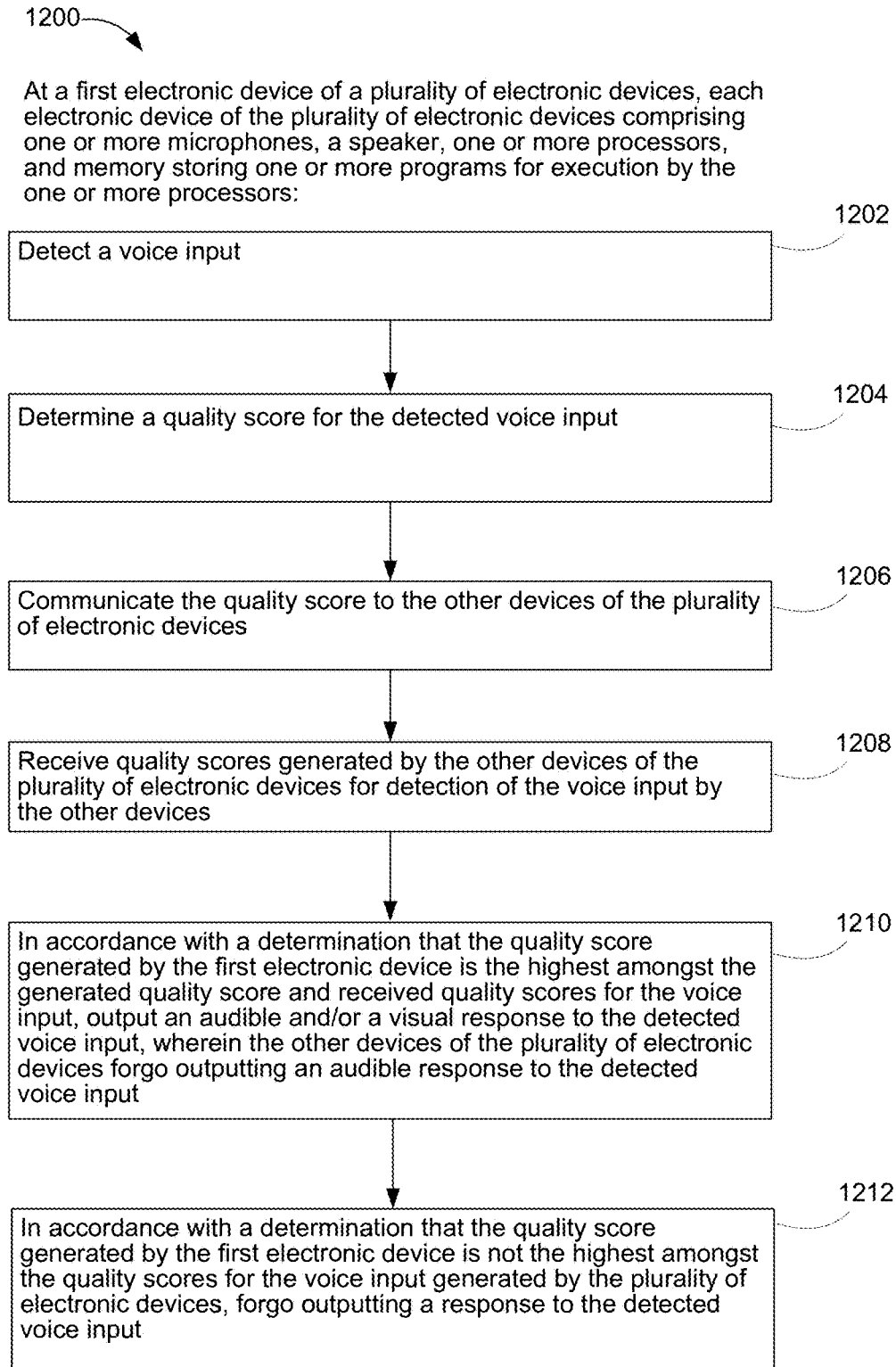
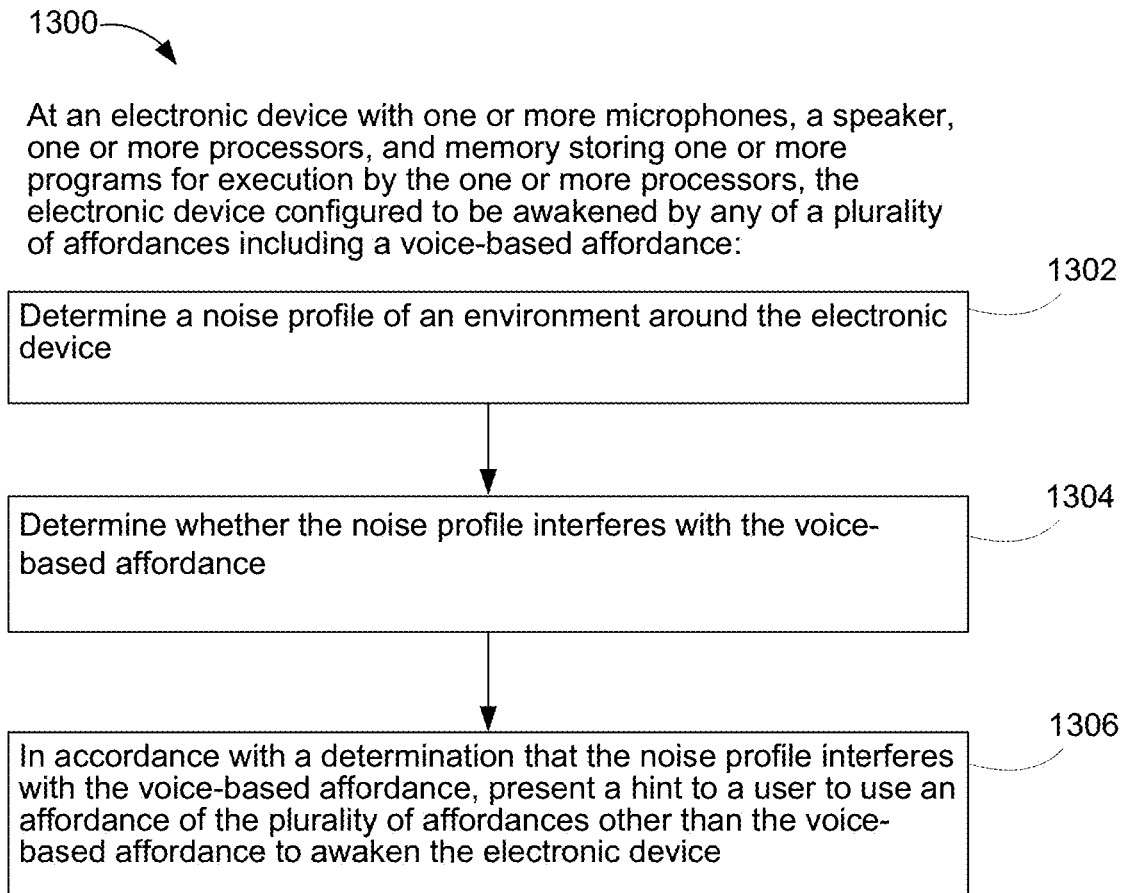


Figure 5

**Figure 6**

**Figure 7**

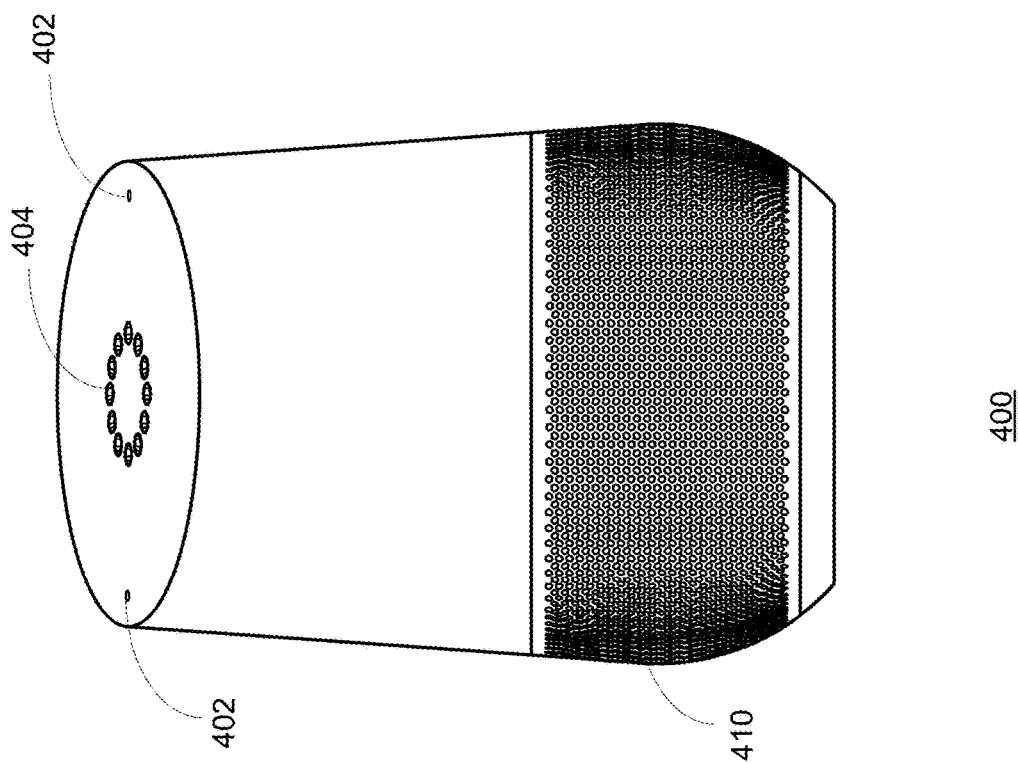


Figure 8A

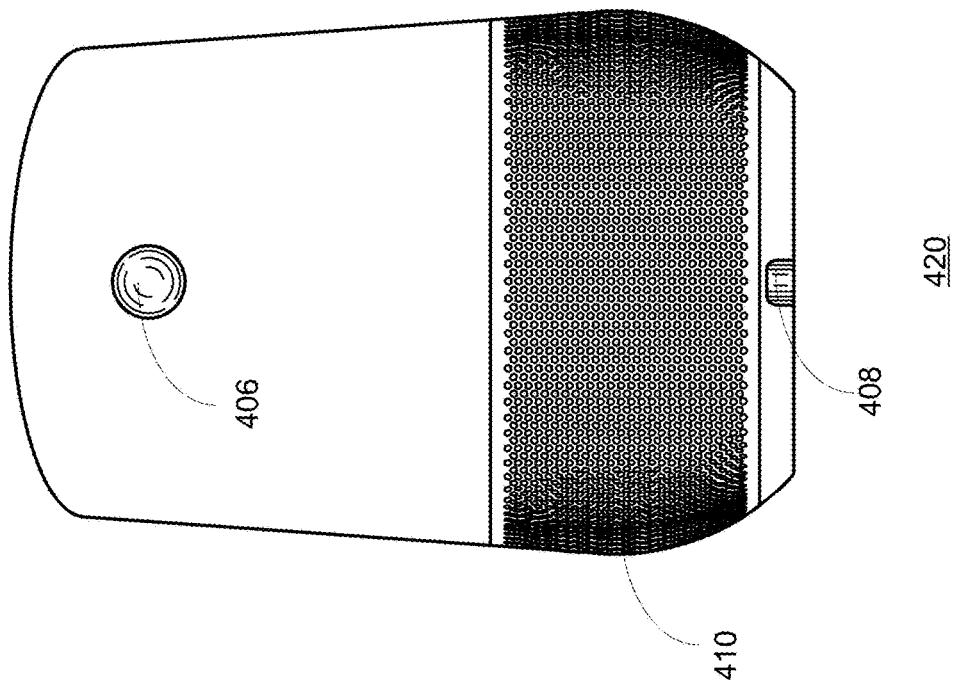


Figure 8B

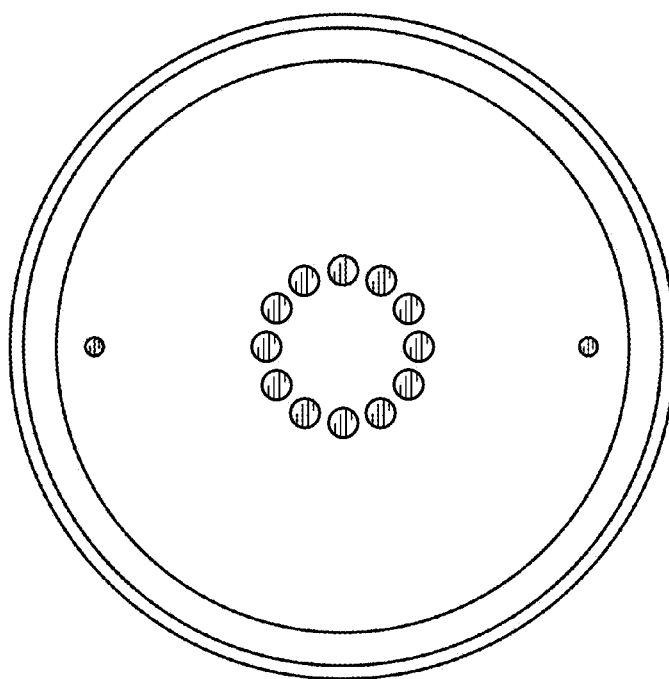


Figure 8C

Motion Sketches

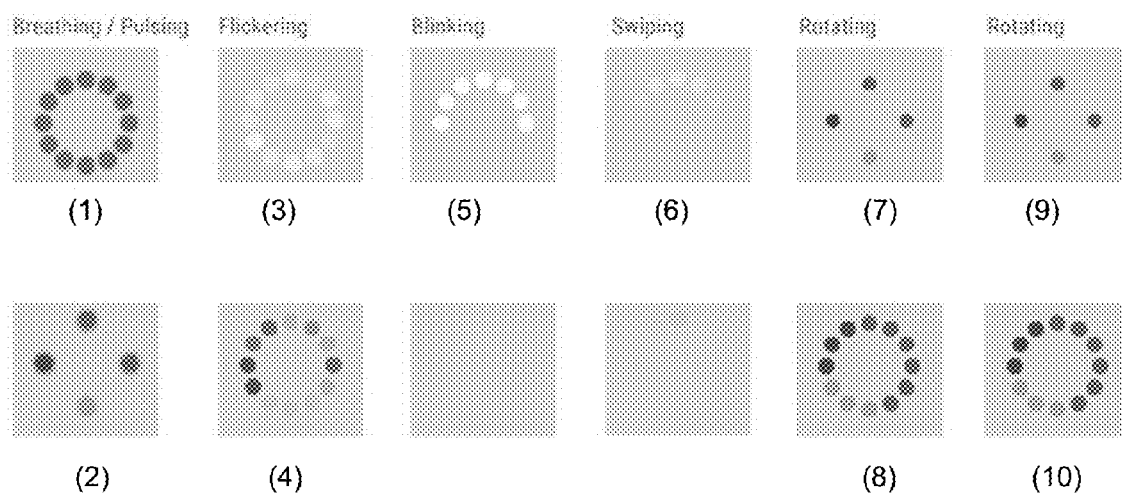


Figure 8D

1

DEVICE LEADERSHIP NEGOTIATION AMONG VOICE INTERFACE DEVICES

RELATED APPLICATIONS

This application is a continuation-in-part of U.S. patent application Ser. No. 15/088,477, titled "Hotword Detection on Multiple Devices," filed Apr. 1, 2016, which is a continuation of U.S. patent application Ser. No. 14/675,932, titled "Hotword Detection on Multiple Devices," filed Apr. 1, 2015, now U.S. Pat. No. 9,318,107, which claims the benefit of U.S. Provisional Application No. 62/061,830, titled "Hotword Detection on Multiple Devices," filed Oct. 9, 2014, all of which are incorporated by reference herein in their entirety.

This application is related to the following applications, which are incorporated by reference herein in their entirety:

U.S. Provisional Patent Application No. 62/334,434, entitled "Implementations for Voice Assistant on Devices," filed May 10, 2016; and

U.S. Provisional Patent Application No. 62/336,566, entitled "LED Design Language for Visual Affordance of Voice User Interfaces," filed May 13, 2016.

TECHNICAL FIELD

The disclosed implementations relates generally to voice interfaces and related devices, including but not limited to methods and systems for coordination amongst multiple voice interface devices.

BACKGROUND

Electronic devices integrated with microphones have been widely used to collect voice inputs from users and implement voice-activated functions according to the voice inputs. For example, many state-of-the-art mobile devices include a voice assistant feature (e.g., Siri, Google Assistant) that is configured to use voice inputs to initiate a phone call, conduct a restaurant search, start routing on a map, create calendar events, add a post to a social network, recognize a song, and complete many other tasks.

A location (e.g., a room or space within a home) may include multiple devices that include voice assistant systems and one or more users of such devices. As the devices are more than ever configured to be capable of accessing and presenting user-specific information, it is desirable for a voice assistant device to be able to handle multiple users gracefully and be able to provide user-specific information to the appropriate user. It is also desirable for there to be a leader amongst the voice assistant devices that is responsible for responding to the user's voice inputs, in order to reduce user confusion. Further, it is desirable for a voice assistant device to include noise mitigation measures in case the voice assistant devices are located at a noisy location.

SUMMARY

Accordingly, there is a need for an electronic device with a voice assistant system and/or a voice assistant server system that incorporates methods and systems for handling multiple users, for multi-device coordination to determine a leader, and for noise mitigation. The device and/or the server system may be configured to identify a speaker of a voice input based on comparisons to voice models, and to personalize responses to the identified speaker. The device and/or the server system may also be configured to negotiate

2

leadership amongst multiple devices for response to a speaker. The device and/or the server system may further be configured to detect that an environment around the device is sufficiently noisy where an alternative method, to a hotword voice input, of waking up the device is advisable.

In accordance with some implementations, a method includes, at an electronic device with one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors: receiving a first freeform voice input, comparing the first voice input to a set of voice models associated with a plurality of users associated with the electronic device, where the voice models are trained to the electronic device, based on the comparing, determining that the first voice input corresponds to a first user of the plurality of users, and presenting a response in accordance with the determination.

In some implementations, an electronic device includes one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors. The one or more programs include instructions for: receiving a first freeform voice input, comparing the first voice input to a set of voice models associated with a plurality of users associated with the electronic device, where the voice models are trained to the electronic device, based on the comparing, determining that the first voice input corresponds to a first user of the plurality of users, and presenting a response in accordance with the determination.

In some implementations, a non-transitory computer readable storage medium stores one or more programs. The one or more programs include instructions, which, when executed by an electronic device with one or more microphones, a speaker, and one or more processors, cause the electronic device to perform operations including: receiving a first freeform voice input, comparing the first voice input to a set of voice models associated with a plurality of users associated with the electronic device, where the voice models are trained to the electronic device, based on the comparing, determining that the first voice input corresponds to a first user of the plurality of users, and presenting a response in accordance with the determination.

In accordance with some implementations, a method includes, at a first electronic device of a plurality of electronic devices, each electronic device of the plurality of electronic devices comprising one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors: detecting a voice input, determining a quality score for the detected voice input, communicating the quality score to the other devices of the plurality of electronic devices, receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices, in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, where the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input, and in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

In accordance with some implementations, a first electronic device of a plurality of electronic devices includes,

3

and each of the plurality of electronic devices includes, one or more microphones, a speaker, one or more processors, and memory storing one or more programs to be executed by the one or more processors. The one or more programs include instructions for: detecting a voice input; determining a quality score for the detected voice input; communicating the quality score to the other devices of the plurality of electronic devices; receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices; in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, where the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

In accordance with some implementations, a non-transitory computer readable storage medium stores one or more programs. The one or more programs include instructions, which, when executed by a first electronic device of a plurality of electronic devices, each of the plurality of electronic device comprising one or more microphones, a speaker, and one or more processors, cause the electronic device to perform operations including: detecting a voice input; determining a quality score for the detected voice input; communicating the quality score to the other devices of the plurality of electronic devices; receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices; in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, where the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

In some implementations, a method includes, at an electronic device with one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors, the electronic device configured to be awakened by any of a plurality of affordances including a voice-based affordance: determining a noise profile of an environment around the electronic device; determining whether the noise profile interferes with the voice-based affordance; and in accordance with a determination that the noise profile interferes with the voice-based affordance, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device.

In some implementations, an electronic device includes one or more microphones, a speaker, one or more processors, and memory storing one or more programs to be executed by the one or more processors. The electronic device is configured to be awakened by any of a plurality of affordances including a voice-based affordance. The one or more programs include instructions for: determining a noise

4

profile of an environment around the electronic device, determining whether the noise profile interferes with the voice-based affordance, and in accordance with a determination that the noise profile interferes with the voice-based affordance, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device.

A non-transitory computer readable storage medium stores one or more programs. The one or more programs include instructions, which, when executed by an electronic device with one or more microphones, a speaker, and one or more processors, the electronic device configured to be awakened by any of a plurality of affordances including a voice-based affordance, cause the electronic device to perform operations including: determining a noise profile of an environment around the electronic device, determining whether the noise profile interferes with the voice-based affordance, and in accordance with a determination that the noise profile interferes with the voice-based affordance, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device.

BRIEF DESCRIPTION OF THE DRAWINGS

The patent or application file contains at least one drawing executed in color. Copies of this patent or patent application publication with color drawing(s) will be provided by the Office upon request and payment of the necessary fee.

For a better understanding of the various described implementations, reference should be made to the Description of Implementations below, in conjunction with the following drawings in which like reference numerals refer to corresponding parts throughout the figures.

FIG. 1 is an example operating environment in accordance with some implementations.

FIGS. 2A-2B are block diagrams illustrating an example electronic device that is applied as a voice interface to collect user voice commands in an operating environment in accordance with some implementations.

FIG. 3 is a block diagram illustrating an example server in the server system of an operating environment in accordance with some implementations. An example server is one of a voice assistance server.

FIGS. 4A-4B are diagrams illustrating an example device leadership negotiation, in accordance with some implementations.

FIG. 5 is a flow diagram illustrating a method of responding to a voice input of a user in accordance with some implementations.

FIG. 6 is a flow diagram illustrating a method of negotiating device leadership among multiple voice interface devices in accordance with some implementations.

FIG. 7 is a flow diagram illustrating a method of mitigating noise around a voice interface device in accordance with some implementations.

FIGS. 8A and 8B are a front view and a rear view of a voice-activated electronic device in accordance with some implementations.

FIG. 8C is a top view of a voice-activated electronic device in accordance with some implementations, and FIG. 8D shows six visual patterns displayed by an array of full color LEDs for indicating voice processing states in accordance with some implementations.

5

Like reference numerals refer to corresponding parts throughout the several views of the drawings.

DESCRIPTION OF IMPLEMENTATIONS

While the digital revolution has provided many benefits ranging from openly sharing information to a sense of global community, emerging new technology often induces confusion, skepticism and fear among consumers, preventing consumers from benefiting from the technology. Electronic devices are conveniently used as voice interfaces to receive voice inputs from users and initiate voice-activated functions, and thereby offer eyes-free and hands-free solutions to approach both existing and emerging technology. Specifically, the voice inputs received at the electronic device can carry instructions and information even if a user's line of sight is obscured and his hands are full. To enable a hands-free and eyes-free experience, the voice-activated electronic device listens to the ambient (i.e., constantly processes audio signals collected from the ambient) constantly or only when triggered. On the other hand, user identities are linked with a user's voice and a language used by the user. To protect the user identities, these voice-activated electronic devices are normally used in non-public places that are protected, controlled and intimate spaces (e.g., home and car).

In accordance with some implementations, a voice-activated electronic device identifies a speaker of a voice input as one of one or more associated or registered users. Responses to the speaker output by the electronic device are personalized to the identified speaker. The speaker is identified by comparing the speaker's voice input to a set of trained voice models or voice fingerprints. In some implementations, words and phrases used by a speaker to train the voice models are not necessarily predefined and do not need to match words later spoken by the speaker in order for the electronic device to identify the speaker.

Further, in accordance with some implementations, multiple voice-activated electronic devices negotiate a leader amongst themselves to respond to voice inputs from a user. The negotiation may be based on which device detected the voice input the best, or which device is the closest to the user. Additionally, if the voice input is particularly relevant to one of the devices, that device may be selected to respond even if it would otherwise not be selected for response; the relevance of the command included in the voice input to a particular device (e.g., "stop music" should refer to a device playing music) and the state of the device receiving the voice input (e.g., "screen on" vs. "screen off") are considered in determining which device will be the leader.

Further, in accordance with some implementations, a voice-activated device determines whether the surrounding noise is sufficiently interfering with detection of spoken hotwords for waking up the device (in some implementations, a "hotword" is a user defined or predefined word or phrase used to "wake-up" or trigger a voice-activated device to attend/respond to a spoken command that is issued subsequent to the hotword) or other spoken voice inputs. If the noise is sufficiently interfering, the device indicates such and gives the user a hint to use an alternative way of waking up the device or other noise mitigation measures.

Voice Assistant Operating Environment

FIG. 1 is an example operating environment in accordance with some implementations. Operating environment 100 includes one or more voice-activated electronic devices

6

190 (e.g., electronic devices 190-1 thru 190-N). The one or more voice-activated electronic devices 190 may be located in one or more locations (e.g., all in a room or space of a structure, spread out throughout multiple spaces within a structure or throughout multiple structures (e.g., one in a house and one in the user's car)). The environment 100 optionally includes one or more devices (e.g., media devices, smart home devices) (not shown) with which the voice-activated electronic devices 190 can communicate.

The electronic devices 190 are communicatively coupled, through communication networks 110, to a voice assistance server 112 of a voice assistant service. One or more of the electronic devices 190 are communicatively coupled to a local network 104, which is communicatively coupled to the communication networks 110 (e.g., electronic devices 190-1 and 190-2). In some implementations, the local network 104 is a local area network implemented at a network interface (e.g., a router). The electronic devices 190 that are communicatively coupled to the local network 104 may also communicate with each other through the local network 104.

Optionally, one or more of the electronic devices 190 are communicatively coupled to the communication networks 110 and are not on the local network 104 (e.g., electronic device 190-N). For example, these electronic devices 190 are not on the Wi-Fi network corresponding to the local network 104 but are connected to the communication networks 110 through a cellular connection. In some implementations, communication between electronic devices 190 that are on the local network 104 and electronic devices 190 that are not on the local network 104 are done through the voice assistance server 112. The electronic devices 190 are registered in a device registry 118 of the voice assistant service and thus known to the voice assistance server 112.

The environment 100 also includes one or more content hosts 114. A content host 114 may be a remote content source from which content is streamed or otherwise obtained in accordance with a user voice request. A content host 114 may be an information source from which the voice assistance server 112 retrieves information in accordance with a user voice request.

In some implementations, an electronic device 190 may be associated with multiple users having respective user accounts in the user domain. Any of these users, as well as users not associated with the device, may make voice inputs to the electronic device 190. The electronic device 190 receives these voice inputs from these users 102-1 thru 102-M (including associated and non-associated users), and the electronic device 190 and/or the voice assistance server 112 proceeds to identify, for a voice input, the user making the voice input. With the user identification, a response to that voice input may be personalized to the identified user.

In some implementations, the environment 100 includes multiple electronic devices 190 (e.g., devices 190-1 thru 190-N). The devices 190 are located throughout the environment 100 (e.g., all within a room or space in a structure, spread throughout the structure, some within the structure and some without). When a user 102 makes a voice input, each of the devices 190 either receives the input or does not receive the input (e.g., if the device was too far away from the user). The devices that receive the input receive the input at varying degrees of quality; the quality of the sample of the voice input at a device is based on multiple factors, including but not limited to distance of the user from the device and the noise around the device. The multiple devices 190 negotiate a leader amongst themselves to respond to the user and to receive further voice input from the user 102 based on the quality of the samples of the voice inputs.

7

In some implementations, an electronic device **190** determines a level of noise around the device and determines whether the determined noise level is sufficiently high to interfere with recognition of the hotword in voice inputs, and thus interfere with awakening of the device by voice, or with recognition of voice requests. If the noise level is determined to be sufficiently high to be interfering, the electronic device **190** indicates to the user that the noise level is interfering and gives the user a hint that the user should use another way to wake up the electronic device **190** (e.g., activate a button). The indication of the interfering noise level and the hint to use another way to wake up the device may be done in the same presentation (e.g., illuminating the wake-up button).

In some implementations, one or more media devices are disposed in the operating environment **100** to provide to one or more occupants media content, news and/or other information. In some implementations, the content provided by the media devices is stored at a local content source, streamed from a remote content source (e.g., content host(s) **114**), or generated locally (e.g., through a local text to voice processor that reads a customized news briefing, emails, texts, a local weather report, etc. to one or more occupants of the operating environment **100**). In some implementations, the media devices include media output devices that directly output the media content to an audience (e.g., one or more users), and cast devices that are networked to stream media content to the media output devices. Examples of the media output devices include, but are not limited to television (TV) display devices and music players. Examples of the cast devices include, but are not limited to, set-top boxes (STBs), DVD players, TV boxes, and media streaming devices, such as Google's Chromecast™ media streaming device.

In some implementations, the media devices include one or more voice-activated electronic devices **190** that receive, process and respond to voice commands of occupants. In some implementations, the voice-activated electronic devices **190** respond to voice commands by: generating and providing a spoken response to a voice command (e.g., speaking the current time in response to the question, "what time is it?"); streaming media content requested by a user (e.g., "play a Beach Boys song"); reading a news story or a daily news briefing prepared for the user; playing a media item stored on the personal assistant device or on the local network; changing a state or operating one or more other connected devices within the operating environment **100** (e.g., turning lights, appliances or media devices on/off, locking/unlocking a lock, opening windows, etc.); or issuing a corresponding request to a server via a network **110**.

In some implementations, the one or more electronic devices **190** are disposed in the operating environment **100** to collect audio inputs for initiating various functions, including media play functions of the media devices. In some implementations, these voice-activated electronic devices **190** (e.g., devices **190-1** thru **190-N**) are disposed in proximity to a media device, for example, in the same room with the cast devices and the media output devices. Alternatively, in some implementations, a voice-activated electronic device **190** is disposed in a room having one or more smart home devices but not any media device. Alternatively, in some implementations, a voice-activated electronic device **190** is disposed in a location having no networked electronic device. Further, in some implementations, a room or space in the structure may have multiple electronic devices **190**.

In some implementations, the electronic device **190** includes at least one or more microphones, a speaker, a

8

processor and memory storing at least one program for execution by the processor. The speaker is configured to allow the electronic device **190** to deliver voice messages and other audio (e.g., audible tones) to a location where the electronic device **190** is located in the operating environment **100**, thereby broadcasting music, reporting a state of audio input processing, having a conversation with or giving instructions to a user of the electronic device **190**. As an alternative to the voice messages, visual signals could also be used to provide feedback to the user of the electronic device **190** concerning the state of audio input processing. When the electronic device **190** is a mobile device (e.g., a mobile phone or a tablet computer), its display screen is configured to display a notification concerning the state of audio input processing.

In some implementations, the electronic device **190** is a voice interface device that is network-connected to provide voice recognition functions with the aid of a voice assistance server **112** and/or optionally a cloud cast service server (not shown). For example, the electronic device **190** includes a smart speaker that provides music to a user and allows eyes-free and hands-free access to a voice assistant service (e.g., Google Assistant). Optionally, the electronic device **190** is one of a desktop or laptop computer, a tablet, a mobile phone that includes a microphone, a cast device that includes a microphone and optionally a speaker, an audio system (e.g., a stereo system, a speaker system, a portable speaker) that includes a microphone and a speaker, a television that includes a microphone and a speaker, and a user interface system in an automobile that includes a microphone and a speaker and optionally a display. Optionally, the electronic device **190** is a simple and low cost voice interface device. Generally, the electronic device **190** may be any device that is capable of network connection and that includes a microphone, a speaker, and programs, modules, and data for interacting with voice assistant service. Given simplicity and low cost of the electronic device **190**, the electronic device **190** includes an array of light emitting diodes (LEDs) rather than a full display screen, and displays a visual pattern on the LEDs to indicate the state of audio input processing. In some implementations, the LEDs are full color LEDs, and the colors of the LEDs may be employed as a part of the visual pattern to be displayed on the LEDs. Multiple examples of using LEDs to display visual patterns in order to convey information or device status are described in U.S. Provisional Patent Application No. 62/336,566, entitled "LED Design Language for Visual Affordance of Voice User Interfaces," filed May 13, 2016, which is incorporated by reference in its entirety. In some implementations, visual patterns indicating the state of voice processing operations are displayed using characteristic images shown on conventional displays associated with electronic devices that are performing the voice processing operations.

In some implementations, LEDs or other visual displays are used to convey a collective voice processing state of multiple participating electronic devices. For example, in an operating environment where there are multiple voice processing or voice interface devices (e.g., multiple electronic devices **400** as shown in FIG. 4A of the '566 application, duplicated as FIG. 8A below; multiple electronic devices **190**), groups of color LEDs (e.g., LEDs **404** as shown in FIG. 4A of the '566 application) associated with respective electronic devices can be used to convey which of the electronic devices is listening to a user, and which of the listening devices is the leader (where the "leader" device generally takes the lead in responding to a spoken request issued by the user).

More generally, the '566 application describes (e.g., see paras. [0087]-[0100]) a "LED Design Language" for indicating visually using a collection of LEDs a variety of voice processing states of an electronic device, such as a "Hot word detection state and listening state," a "Thinking mode or working mode," and a "Responding mode or speaking mode." In some implementations, unique states of voice processing operations described herein are represented using a group of LEDs in accordance with one or more aspects of the "LED Design Language" of the '566 application. These visual indicators can also be combined with one or more audible indicators generated by electronic devices that are performing voice processing operations. The resulting audio and/or visual indicators will enable users in a voice-interactive environment to understand the state of various voice processing electronic devices in the environment and to effectively interact with those devices in a natural, intuitive manner.

When voice inputs from the electronic device **190** are used to control the media output devices via the cast devices, the electronic device **190** effectively enables a new level of control of cast-enabled media devices. In a specific example, the electronic device **190** includes a casual enjoyment speaker with far-field voice access and functions as a voice interface device for the voice assistant service. The electronic device **190** could be disposed in any area in the operating environment **100**. When multiple electronic devices **190** are distributed in multiple rooms, they become cast audio receivers that are synchronized to provide voice inputs from all these rooms.

Specifically, in some implementations, the electronic device **190** includes a Wi-Fi speaker with a microphone that is connected to a voice-activated voice assistant service (e.g., Google Assistant). A user can issue a media play request via the microphone of electronic device **190**, and ask the voice assistant service to play media content on the electronic device **190** itself or on another connected media output device. For example, the user can issue a media play request by saying to the Wi-Fi speaker "OK Google, play cat videos on my Living room TV." The voice assistant service then fulfils the media play request by playing the requested media content on the requested device using a default or designated media application.

In some implementations, a user can issue a voice request, via the microphone of the electronic device **190**, concerning media content that has already been played or is being played on a display device (e.g., the user can ask for information about the media content, buy the media content through an online store, or compose and issue a social post about the media content).

In some implementations, a user may want to take a current media session with them as they move through the house and can request such a service from one or more of the electronic devices **190**. This requires the voice assistant service to transfer the current media session from a first cast device to a second cast device that is not directly connected to the first cast device or has no knowledge of the existence of the first cast device. Subsequent to the media content transfer, a second output device coupled to the second cast device continues to play the media content previously a first output device coupled to the first cast device from the exact point within a music track or a video clip where play of the media content was forgone on the first output device. In some implementations, the electronic device **190** that receives the request to transfer the media session can satisfy the request. In some implementations, the electronic device **190** that receives the request to transfer the media session

relays the request to another device or system (e.g., a hub device, voice assistance server **112**) for handling.

Further, in some implementations, a user may issue, via the microphone of electronic device **190**, a request for information or for performance of an action or operation. The information requested may be personal (e.g., the user's emails, the user's calendar events, the user's flight information, etc.), non-personal (e.g., sports scores, news stories, etc.) or somewhere in between (e.g., scores for teams or sports preferred by the user, news stories from the user's preferred sources, etc.). The requested information or action/operation may involve access to personal information (e.g., purchasing a digital media item with payment information provided by the user, purchasing a physical good). The electronic device **190** responds to the request with voice message responses to the user, where the response may include, for example, requests for additional information to fulfill the request, confirmation that the request has been fulfilled, notice that the request cannot be fulfilled, and so forth.

In some implementations, in addition to the voice-activated electronic devices **190** and the media devices (e.g., the output devices and the cast devices), the operating environment **100** may also include one or more smart home devices (not shown). The integrated smart home devices include intelligent, multi-sensing, network-connected devices that integrate seamlessly with each other in a smart home network and/or with a central server or a cloud-computing system to provide a variety of useful smart home functions. In some implementations, a smart home device is disposed at the same location of the operating environment **100** as a cast device and/or an output device, and therefore, is located in proximity to or with a known distance with respect to the cast device and the output device.

The smart home devices in the operating environment **100** may include, but are not limited to, one or more intelligent, multi-sensing, network-connected thermostats, one or more intelligent, network-connected, multi-sensing hazard detectors, one or more intelligent, multi-sensing, network-connected entryway interface devices and (hereinafter referred to as "smart doorbells" and "smart door locks"), one or more intelligent, multi-sensing, network-connected alarm systems, one or more intelligent, multi-sensing, network-connected camera systems, and one or more intelligent, multi-sensing, network-connected wall switches, and one or more intelligent, multi-sensing, network-connected power sockets. In some implementations, the smart home devices in the operating environment **100** of FIG. 1 includes a plurality of intelligent, multi-sensing, network-connected appliances (hereinafter referred to as "smart appliances"), such as refrigerators, stoves, ovens, televisions, washers, dryers, lights, stereos, intercom systems, garage-door openers, floor fans, ceiling fans, wall air conditioners, pool heaters, irrigation systems, security systems, space heaters, window AC units, motorized duct vents, and so forth. In some implementations, any one of these smart home device types can be outfitted with microphones and one or more voice processing capabilities as described herein so as to in whole or in part respond to voice requests from an occupant or user.

In some implementations, each of the cast devices and the voice-activated electronic devices **190** is capable of data communications and information sharing with other cast devices, voice-activated electronic devices **190**, smart home devices, a central server or cloud-computing system, and/or other devices (e.g., the client device) that are network-connected. Data communications may be carried out using any of a variety of custom or standard wireless protocols

11

(e.g., IEEE 802.15.4, Wi-Fi, ZigBee, 6LoWPAN, Thread, Z-Wave, Bluetooth Smart, ISA100.11a, WirelessHART, MiWi, etc.) and/or any of a variety of custom or standard wired protocols (e.g., Ethernet, HomePlug, etc.), or any other suitable communication protocol, including communication protocols not yet developed as of the filing date of this document.

Through the communication networks (e.g., the Internet) **110**, the cast devices, the electronic devices **190** and the smart home devices may communicate with a server system (also called a central server system and/or a cloud-computing system herein). Optionally, the server system may be associated with a manufacturer, support entity, or service provider associated with the cast devices and the media content displayed to the user. Accordingly, the server system includes the voice assistance server **112** that processes audio inputs collected by voice-activated electronic devices **190**, one or more content hosts **114** that provide the displayed media content, a cloud cast service server creating a virtual user domain based on distributed device terminals, and the device registry **118** that keeps a record of the distributed device terminals in the virtual user environment. Examples of the distributed device terminals include, but are not limited to the cast devices, the media output devices, the electronic devices **190** and the smart home devices. In some implementations, these distributed device terminals are linked to a user account (e.g., a Google user account) in the virtual user domain. It should be appreciated that processing of audio inputs collected by electronic devices **190** can be performed locally at an electronic device **190**, at a voice assistance server **112**, at another smart home device (e.g., a hub device) or at some combination of all or subset of the above.

It will be appreciated that in some implementations the electronic device(s) **190** also function in an environment without smart home devices. For example, an electronic device **190** can, even in the absence of smart home devices, respond to user requests for information or performance of an action, and/or to initiate or control various media play functions. An electronic device **190** can also function in a wide range of environments, including, without limitation, a vehicle, a ship, a business, or a manufacturing environment.

In some implementations, an electronic device **190** is “awakened” (e.g., to activate an interface for the voice assistant service on the electronic device **190**, to put the electronic device **190** into a state where the electronic device **190** is ready to receive voice requests to the voice assistant service) by a voice input that includes a hotword (also called a “wake word”). In some implementations, the electronic device **190** requires awakening if the electronic device **190** has been idle with respect to receipt of voice inputs for at least a predefined amount of time (e.g., 5 minutes); the predefined amount of time corresponds to an amount of idle time allowed before a voice interface session or conversation times out. The hotword may be a word or phrase, and may be a predefined default and/or may be customized by a user (e.g., a user may set a nickname for a particular electronic device **190** as the device’s hotword). In some implementations, there may be multiple hotwords that can awaken an electronic device **190**. A user may speak the hotword, wait for an acknowledgement response from the electronic device **190** (e.g., the electronic device **190** outputs a greeting), and then make a first voice request. Alternatively, the user may combine the hotword and the first voice request in one voice input (e.g., the voice input includes the hotword followed by the voice request).

12

In some implementations, the voice inputs spoken by the user to an electronic device **190** may generally be freeform or natural language speech. That is, the voice input need not be strictly limited to a predefined set of words and phrases within a predefined syntax, with possibly certain exceptions (e.g., user needs to speak the hotword first to awaken the device).

In some implementations, an electronic device **190** includes one or more additional ways or affordances to wake up the device besides speaking the hotword to the electronic device **190**. The additional ways or affordances may include, for example, activating a predefined button or touch sensor (e.g., a touch sense array) on the electronic device **190**.

In some implementations, a voice-activated electronic device **190** interacts with a cast device, a client device or a server system of an operating environment **100** in accordance with some implementations. The voice-activated electronic device **190** is configured to receive audio inputs from an environment in proximity to the voice-activated electronic device **190**. Optionally, the electronic device **190** stores the audio inputs and at least partially processes the audio inputs locally. Optionally, the electronic device **190** transmits the received audio inputs or the partially processed audio inputs to a voice assistance server **112** via the communication networks **110** for further processing. The cast device is configured to obtain media content or Internet content from one or more content hosts **114** for display on an output device coupled to the cast device. In some implementations, the cast device and the voice-activated electronic device **190** are linked to each other in a user domain, and more specifically, associated with each other via a user account in the user domain. Information of the cast device and information of the electronic device **190** are stored in the device registry **118** in association with the user account. In some implementations, there is a device registry for cast devices and a registry for voice-activated electronic devices **190**. In some implementations, a cloud cast service server manages the cast devices registry and the voice assistance server **112** manages the voice-activated electronic devices registry. The cast devices registry may reference devices in the voice-activated electronic devices registry that are associated in the user domain, and vice versa.

In some implementations, one or more of the electronic devices **190** (and one or more cast devices) are commissioned to the voice assistant service via a client device (not shown). In some implementations, the voice-activated electronic device **190** does not include any display screen, and relies on the client device to provide a user interface during a commissioning process, and similarly for a cast device as well. Specifically, the client device is installed with an application that enables a user interface to facilitate commissioning of a new voice-activated electronic device **190** disposed in proximity to the client device. A user may send a request on the user interface of the client device to initiate a commissioning process for the new electronic device **190** that needs to be commissioned. After receiving the commissioning request, the client device establishes a short range communication link with the new electronic device **190** that needs to be commissioned. Optionally, the short range communication link is established based near field communication (NFC), Bluetooth, Bluetooth Low Energy (BLE) and the like. The client device then conveys wireless configuration data associated with a wireless local area network (WLAN) (e.g., local network **104**) to the new or electronic device **190**. The wireless configuration data includes at least a WLAN security code (i.e., service set identifier (SSID) password), and optionally includes a SSID, an Internet

13

protocol (IP) address, proxy configuration and gateway configuration. After receiving the wireless configuration data via the short range communication link, the new electronic device **190** decodes and recovers the wireless configuration data, and joins the WLAN based on the wireless configuration data.

In some implementations, additional user domain information is entered on the user interface displayed on the client device, and used to link the new electronic device **190** to an account in a user domain. Optionally, the additional user domain information is conveyed to the new electronic device **190** in conjunction with the wireless communication data via the short range communication link. Optionally, the additional user domain information is conveyed to the new electronic device **190** via the WLAN after the new device has joined the WLAN.

Once the electronic device **190** has been commissioned into the user domain, other devices and their associated activities may be controlled via multiple control paths. In accordance with one control path, an application installed on the client device is used to control the other device and its associated activities (e.g., media play activities). Alternatively, in accordance with another control path, the electronic device **190** is used to enable eyes-free and hands-free control of the other device and its associated activities.

Voice Activated Media Play on a Media Output Device

In some implementations, after the cast device and the voice-activated electronic device **190** are both commissioned and linked to a common user domain, the voice-activated electronic device **190** can be used as a voice user interface to enable eyes-free and hands-free control of media content streaming to the cast device involving no remote control, client device or other second screen device. For example, the user may give voice commands such as "Play Lady Gaga on Living Room speakers." A Lady Gaga music track or video clip is streamed to a cast device associated with the "Living Room speakers." The client device is not involved, nor is any cast device application or media play application loaded on the client device.

In some implementations, a proxy service (e.g., voice assistant service, a cloud cast service) communicatively links the voice-activated electronic device **190** to the cast device and makes casting to the cast device possible without involving any applications on the client device. Specifically, a voice message is captured and recorded by an electronic device **190**, and the voice message is configured to request media play on a media output device. Optionally, the electronic device **190** partially processes the voice message locally. Optionally, the electronic device **190** transmits the voice message or the partially processed voice message to a voice assistance server **112** via the communication networks **110** for further processing. The voice assistance server **112** or a cloud cast service server determines that the voice message includes a first media play request, and that the first media play request includes a user voice command to play media content on a media output device and a user voice designation of the media output device. The user voice command further includes at least information of a first media play application (e.g., YouTube and Netflix) and the media content (e.g., Lady Gaga music) that needs to be played.

In accordance with the voice designation of the media output device, the voice assistance server **112** or the cloud cast service server identifies in a device registry **118** a cast

14

device associated in the user domain with the electronic device **190** and coupled to the media output device. The cast device is configured to execute one or more media play applications for controlling the media output device to play media content received from one or more media content hosts **114**. Then, the cloud cast service server sends to the cast device a second media play request including the information of the first media play application and the media content that needs to be played. Upon receiving the information sent by the cloud cast service server, the cast device executes the first media play application and controls the media output device to play the requested media content.

In some implementations, the user voice designation of the media output device includes a description of the destination media output device. The voice assistance server **112** or the cloud cast service server identifies in the device registry **118** the destination media output device among a plurality of media output devices according to the description of the destination media output device. In some implementations, the description of the destination media output device includes at least a brand ("Samsung TV") or a location of the media output device ("my Living Room TV").

Voice Activated Media Transfer Among Media Output Devices

In some implementations, the operating environment **100** includes a first cast device (not shown) and a first output device (not shown) coupled to the first cast device. The operating environment **100** also includes a second cast device (not shown) and a second output device (not shown) coupled to the second cast device. The cast devices are optionally located in the same location (e.g., the living room) or two distinct locations (e.g., two rooms) in the operating environment **100**. Each of the cast devices is configured to obtain media content or Internet content from content hosts **114** for display on the output device coupled to the respective cast device. Both the first and second cast devices are communicatively coupled to the voice assistance server **112**, optionally a cloud cast service server, and the content hosts **114**.

The operating environment **100** further includes one or more voice-activated electronic devices **190** that are communicatively coupled to the voice assistance server **112** and optionally the cloud cast service server. In some implementations, the voice-activated electronic devices **190** are disposed independently of the cast devices and the output devices. For example, an electronic device **190** is disposed in a room where no cast device or output device is located. In some implementations, a first electronic device **190** is disposed in proximity to the first cast device and the first output device, e.g., the first electronic device **190**, the first cast device and the first output device are located in the same room. Optionally, a second electronic device **190** is disposed independently of or in proximity to the second cast device and the second output device.

When media content is being played on the first output device, a user may send a voice command to any of the electronic devices **190** to request play of the media content to be transferred to the second output device. The voice command includes a media play transfer request. In one situation, the user can issue the voice command to the electronic device **190** disposed in proximity to the first cast device before the user moves to a destination location. Alternatively, in another situation, the user can issue the

15

voice command to the electronic device **190** disposed in proximity to the second device after the user reaches the destination location.

The voice command is transmitted to the voice assistance server **112**. The voice assistance server **112** sends a media display information request to the first cast device to request instant media play information of the media content that is currently being played on the first output device coupled to the first cast device. The first cast device then returns to the voice assistance server **112** the requested instant play information including at least information of a first media play application (e.g., YouTube), the media content that is currently being played (e.g., “Lady Gaga—National Anthem—Super Bowl 2016”), and a temporal position related to playing of the media content. The second cast device then receives a media display request including the instant play information from the voice assistance server **112**, and in accordance with the instant play information, executes the first media play application that controls the second output device to play the media content from the temporal location.

For example, when a music playlist is played on the first output device, the user says “Play on my living room speakers.” The first output device stops playing the currently played song, and the stopped song resumes on the living room speakers. When the song is completed, the living room speakers continue to play the next song on the music playlist previously played on the first output device. As such, when the user is moving around in the operating environment **100**, the play of the media content would seamlessly follow the user while only involving limited user intervention (i.e., giving the voice command).

Individual Devices Involved in the Operating Environment

FIGS. 2A-2B are block diagrams illustrating an example electronic device **190** that is applied as a voice interface to collect user voice commands in an operating environment (e.g., operating environment **100**) in accordance with some implementations. The electronic device **190**, typically, includes one or more processing units (CPUs) **502**, one or more network interfaces **504**, memory **506**, and one or more communication buses **508** for interconnecting these components (sometimes called a chipset). The electronic device **190** includes one or more input devices **510** that facilitate user input, such as a button **512**, a touch sense array **514**, and one or more microphones **516**. The electronic device **190** also includes one or more output devices **518**, including one or more speakers **520**, optionally an array of LEDs **522**, and optionally a display **524**. In some implementations, the array of LEDs **522** is an array of full color LEDs. In some implementations, an electronic device **190**, depending on the type of device, has either the array of LEDs **522**, or the display **524**, or both. In some implementations, the electronic device **190** also includes a location detection device **526** (e.g., a GPS module) and one or more sensors **528** (e.g., accelerometer, gyroscope, light sensor, etc.).

Memory **506** includes high-speed random access memory, such as DRAM, SRAM, DDR RAM, or other random access solid state memory devices; and, optionally, includes non-volatile memory, such as one or more magnetic disk storage devices, one or more optical disk storage devices, one or more flash memory devices, or one or more other non-volatile solid state storage devices. Memory **506**, optionally, includes one or more storage devices remotely located from one or more processing units **502**. Memory **506**, or alternatively the non-volatile memory within memory **506**,

16

includes a non-transitory computer readable storage medium. In some implementations, memory **506**, or the non-transitory computer readable storage medium of memory **506**, stores the following programs, modules, and data structures, or a subset or superset thereof:

Operating system **532** including procedures for handling various basic system services and for performing hardware dependent tasks;

Network communication module **534** for connecting the electronic device **190** to other devices (e.g., the server system **140**, one or more cast devices, one or more client devices, one or more smart home devices, and other electronic device(s) **190**) via one or more network interfaces **504** (wired or wireless) and one or more networks **110**, such as the Internet, other wide area networks, local area networks (e.g., local network **104**), metropolitan area networks, and so on;

Input/output control module **536** for receiving inputs via one or more input devices and enabling presentation of information at the electronic device **190** via one or more output devices **518**, including:

Voice processing module **538** for processing audio inputs or voice messages collected in an environment surrounding the electronic device **190**, or preparing the collected audio inputs or voice messages for processing at a voice assistance server **112**;

LED control module **540** for generating visual patterns on the LEDs **522** according to device states of the electronic device **190**; and

Touch sense module **542** for sensing touch events on a top surface (e.g., on touch sensor array **514**) of the electronic device **190**;

Voice activated device data **544** for storing at least data associated with the electronic device **190**, including:

Voice device settings **546** for storing information associated with the electronic device **190** itself, including common device settings (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.), information of one or more user accounts in a user domain, settings regarding restrictions when dealing with a non-registered user, and display specifications associated with one or more visual patterns displayed by the LEDs **522**;

Voice control data **548** for storing audio signals, voice messages, response messages and other data related to voice interface functions of the electronic device **190**;

Voice models data **550** for storing voice models or voice fingerprints corresponding to users in the user domain that are associated with the electronic device **190**;

Device leadership module **552** for determining a leader amongst multiple electronic devices **190** in response to user voice inputs;

Voice model training module **554** for training and generating voice models or voice fingerprints **550** that can be used to identify and disambiguate users in the user domain that are associated with the electronic device **190**;

Response module **556** for performing instructions included in voice request responses generated by the voice assistance server **112**, and in some implementations, generating responses to certain voice inputs; and Noise module **558** for determining noise levels around the electronic device **190** and provide indications of inter-

17

fering noise levels and hints to use alternative ways to wake up the electronic device.

In some implementations, the voice processing module 538 includes the following modules:

User identification module 560 for identifying and disambiguating users who provide voice inputs to the electronic device 190;

Hotword recognition module 562 for determining whether voice inputs include a hotword for waking up the electronic device 190 and recognizing such in the voice inputs;

Request recognition module 564 for determining a user request included in a voice input; and

Recognition quality module 566 for determining a measure of the quality (e.g., a score) of recognition of hotwords and/or requests in voice inputs.

In some implementations, the noise module 558 includes the following modules:

Recognition analysis module 568 for analyzing how different levels of noise affect the operations of the voice processing module 538 (e.g., in particular the hotword recognition module 562 and/or the request recognition module 564) with respect to the particular electronic device 190;

User volume threshold module 568 for determining, for a given distance from the electronic device 190, highest comfortable volume levels at which users will utter voice inputs to the electronic device, 190;

Noise determination module 572 for determining a noise level and a noise profile around the electronic device 190;

Noise profiles module 574 for training and generating noise profiles of noises that interfere with recognition of hotwords and/or requests in voice inputs; and

Noise profiles store 576 for storing noise profiles that have been determined to be interfering.

Each of the above identified elements may be stored in one or more of the previously mentioned memory devices, and corresponds to a set of instructions for performing a function described above. The above identified modules or programs (i.e., sets of instructions) need not be implemented as separate software programs, procedures, modules or data structures, and thus various subsets of these modules may be combined or otherwise re-arranged in various implementations. In some implementations, memory 506, optionally, stores a subset of the modules and data structures identified above. Furthermore, memory 506, optionally, stores additional modules and data structures not described above. In some implementations, a subset of the programs, modules, and/or data stored in the memory 506 can be stored on and/or executed by the server system 140/voice assistance server 112.

In some implementations, one or more of the modules in memory 506 described above are part of a voice processing library of modules. The voice processing library may be implemented and embedded on a wide variety of devices. An example of a voice processing library is described in U.S. Provisional Patent Application No. 62/334,434, entitled "Implementations for Voice Assistant on Devices," filed May 10, 2016, which is incorporated by reference herein in its entirety.

FIG. 3 is a block diagram illustrating an example server in the server system 140 of an operating environment (e.g., operating environment 100) in accordance with some implementations. An example server is one of a voice assistance server 112. The server 140, typically, includes one or more processing units (CPUs) 702, one or more network inter-

18

faces 704, memory 706, and one or more communication buses 708 for interconnecting these components (sometimes called a chipset). The server 140 could include one or more input devices 710 that facilitate user input, such as a keyboard, a mouse, a voice-command input unit or microphone, a touch screen display, a touch-sensitive input pad, a gesture capturing camera, or other input buttons or controls. Furthermore, the server 140 could use a microphone and voice recognition or a camera and gesture recognition to supplement or replace the keyboard. In some implementations, the server 140 includes one or more cameras, scanners, or photo sensor units for capturing images, for example, of graphic series codes printed on the electronic devices. The server 140 could also include one or more output devices 712 that enable presentation of user interfaces and display content, including one or more speakers and/or one or more visual displays.

Memory 706 includes high-speed random access memory, such as DRAM, SRAM, DDR RAM, or other random access solid state memory devices; and, optionally, includes non-volatile memory, such as one or more magnetic disk storage devices, one or more optical disk storage devices, one or more flash memory devices, or one or more other non-volatile solid state storage devices. Memory 706, optionally, includes one or more storage devices remotely located from one or more processing units 702. Memory 706, or alternatively the non-volatile memory within memory 706, includes a non-transitory computer readable storage medium. In some implementations, memory 706, or the non-transitory computer readable storage medium of memory 706, stores the following programs, modules, and data structures, or a subset or superset thereof:

Operating system 716 including procedures for handling various basic system services and for performing hardware dependent tasks;

Network communication module 718 for connecting the server system 140 to other devices (e.g., various servers in the server system 140, client devices, cast devices, electronic devices 190, and smart home devices) via one or more network interfaces 704 (wired or wireless) and one or more networks 110, such as the Internet, other wide area networks, local area networks, metropolitan area networks, and so on;

User interface module 720 for enabling presentation of information (e.g., a graphical user interface for presenting application(s) 826-830, widgets, websites and web pages thereof, and/or games, audio and/or video content, text, etc.) at a client device;

Command execution module 721 for execution on the server side (e.g., games, social network applications, smart home applications, and/or other web or non-web based applications for controlling a client device, a cast device, an electronic device 190 and a smart home devices and reviewing data captured by such devices), including one or more of:

a cast device application 722 that is executed to provide server-side functionalities for device provisioning, device control, and user account management associated with cast device(s);

one or more media player applications 724 that is executed to provide server-side functionalities for media display and user account management associated with corresponding media sources;

one or more smart home device applications 726 that is executed to provide server-side functionalities for

19

device provisioning, device control, data processing and data review of corresponding smart home devices; and

a voice assistance application **728** that is executed to arrange voice processing of a voice message received from the electronic device **190** or directly process the voice message to extract a user voice command and one or more parameters for the user voice command (e.g., a designation of a cast device or another electronic device **190**); and

Server system data **730** storing at least data associated with automatic control of media display (e.g., in an automatic media output mode and a follow-up mode), including one or more of:

Client device settings **732** for storing information associated with one or more client device, including common device settings (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.), and information for automatic media display control;

Cast device settings **734** for storing information associated with user accounts of the cast device application **722**, including one or more of account access information, information for device settings (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.), and information for automatic media display control;

Media player application settings **736** for storing information associated with user accounts of one or more media player applications **724**, including one or more of account access information, user preferences of media content types, review history data, and information for automatic media display control;

Smart home device settings **738** for storing information associated with user accounts of the smart home applications **726**, including one or more of account access information, information for one or more smart home devices (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.);

Voice assistance data **740** for storing information associated with user accounts of the voice assistance application **728**, including one or more of account access information, information for one or more electronic device **190** (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.);

User data **742** for storing information associated with users in the use domain, including users' subscriptions (e.g., music streaming service subscriptions, video streaming service subscriptions, newsletter subscriptions), user devices (e.g., devices registered in the device registry **118** associated with respective users), user accounts (e.g., users' email accounts, calendar accounts, financial accounts), and other user data; and

User voice profiles **744** for storing voice profiles of the users in the user domain, including, for example, voice models or voice fingerprints of the users and comfortable volume level thresholds of the users.

Device registration module **750** for managing the device registry **118** coupled to the voice assistance server **112**;

Device leadership moderator module **780** for moderating device leadership determinations between the electronic devices **190** in the operating environment;

Noise module **790** for determining noise levels around the electronic device **190**; and

20

Voice processing module **7100** for processing audio inputs or voice messages collected in an environment surrounding the electronic device **190**.

In some implementations, the voice assistance server **112** is primarily responsible for processing of voice inputs and for noise mitigation, and thus one or more of the programs, modules, and data structures in memory **506** described above with reference to FIGS. **2A-2B** are included in respective modules in memory **706** (e.g., the programs, modules, and data structures included with voice processing module **538** are included in voice processing module **7100**, and the programs, modules, and data structures included with noise module **558** are included in noise module **790**). The electronic device **190** either transmits captured voice inputs to the voice assistance server **112** for processing, or first pre-processes the voice inputs and transmits the pre-processed voice inputs to the voice assistance server **112** for processing. In some implementations, the voice assistance server **112** and the electronic device **190** has some shared and some divided responsibilities regarding processing of voice inputs and noise mitigation, and the programs, modules, and data structures shown in FIGS. **2A-2B** may be included in both or divided amongst the voice assistance server **112** and the electronic device **190**. Other programs, modules, and data structures shown in FIGS. **2A-2B** (e.g., voice models data **550**, voice model training module **554**), or analogues thereof, may also be included in the voice assistance server **112**.

Each of the above identified elements may be stored in one or more of the previously mentioned memory devices, and corresponds to a set of instructions for performing a function described above. The above identified modules or programs (i.e., sets of instructions) need not be implemented as separate software programs, procedures, modules or data structures, and thus various subsets of these modules may be combined or otherwise re-arranged in various implementations. In some implementations, memory **706**, optionally, stores a subset of the modules and data structures identified above. Furthermore, memory **706**, optionally, stores additional modules and data structures not described above.

Example Device Leadership Negotiation

FIGS. **4A-4B** illustrate an example of a device leadership negotiation between multiple electronic devices (e.g., electronic devices **190**) in accordance with some implementations. In an operating environment with multiple electronic devices **190** (e.g., operating environment **100**), when a user in the environment makes a voice input, one or more of the multiple electronic devices **190** detect and receive the voice input. The electronic devices **190** determine a leader amongst themselves to respond to the user's voice input and to detect further voice inputs from the user.

FIG. **4A** shows three voice-activated electronic devices **1002-1** thru **1002-3** (e.g., electronic devices **190**) in an operating environment (e.g., operating environment **100**). They may all be in the same space as a user (not shown), e.g., in the same room. The user makes a voice input (e.g., an utterance) (not shown), and the three electronic devices **1002-1** thru **1002-3** detect and capture a sample of the voice input at varying degrees or levels of quality. The quality of a captured sample of the voice input at an electronic device may be affected by one or more factors, including but not limited to surrounding noise, capabilities of the device, distance and position of the device relative to the user, position of the device within the space, and so on.

21

In some implementations, each of the electronic devices determines a level of quality of the respective sample of the voice input captured by the electronic device. The quality level may be expressed as a score or some other value. For example, in FIG. 4A, each of the electronic devices **1002** has captured a respective sample of a user's voice input and has determined a score for its respective sample; the sample at electronic device **1002-1** is scored as 0.98 (on a scale of 0 to 1, with 1 corresponding to the highest quality), the sample at electronic device **1002-2** is scored as 0.37, and the sample at electronic device **1002-3** is scored as 0.61. In some implementations, the recognition quality module **566** (FIG. 2B) at the electronic device **1002** determines the quality level of the respective captured sample. In some implementations, criteria for evaluating the quality of a voice sample include, for example, loudness or amplitude of the sample, presence or absence of noise (e.g., cross-talk, hissing) and corresponding level, presence or absence of distortion (e.g., echo) and corresponding level, frequency profile of the sample, and so forth.

The electronic devices **1002** communicate their quality scores with each other. Thus, for example, electronic device **1002-1** receives the quality scores from electronic devices **1002-2** and **1002-3**. Electronic device **1002-2** receives the quality scores from electronic devices **1002-1** and **1002-3**. Electronic device **1002-3** receives the quality scores from electronic devices **1002-1** and **1002-2**. If all of the electronic devices **1002** are on the same local network (e.g., local network **104**, FIG. 1), the electronic devices **1002** exchange scores through the local network. For example, an electronic device **1002** may send its quality score to the other electronic devices **1002** by sending multicast messages within the local network. If at least one of the electronic devices **1002** is not on the local network, scores may be sent to server system **140**, and the device leadership moderator module **780** of the voice assistance server **112** distributes the scores to the electronic devices **1002**. Alternatively, the device leadership moderator module **780** determines which of the devices is to be the leader based on the scores received, and sends the result of the determination to the devices. In some implementations, the electronic devices **1002** discover each other and their locations through the voice assistance server **112** and device registry **118** (e.g., a device **1002** periodically checks in with the voice assistance server **112** and is informed of other devices associated with the same user account based on information in the device registry **118**).

In some implementations, if just one of the multiple electronic devices **1002** captured a sample of the voice input (e.g., the other electronic devices **1002** were too far away from the user to detect the voice input), the device that captured the voice input sample is determined to be the leader. In some implementations, if an electronic device **1002** failed to capture a sample of the voice input (e.g., the device was too far away from the user to detect the voice input), the quality score for that device for the voice input is 0.00 or a null score; a device with a "0.00" or null score is eliminated from contention to be the leader.

When an electronic device **1002** receives the scores from the other electronic devices **1002**, the electronic device compares its quality score with the scores from the other devices. If its score is not the best (e.g., highest) amongst the scores, the electronic device forgoes outputting responses (e.g., responses generated by the server system **140**) to the voice input (e.g., by powering down or "turning off" the speaker **520**). If its score is the best amongst the scores, the electronic device outputs responses to the voice input and subsequent voice inputs; this device is determined to be the

22

"leader." In some implementations, the leader also proceeds to detect and process or pre-process subsequent voice inputs from the user (whether it be the leader processing the voice inputs and generating the responses, the leader pre-processing the voice inputs for transmission to the voice assistance server **112** which generates the responses, or the leader simply transmitting the voice inputs to the voice assistance server **112** which generates the response), and the non-leaders forgo receiving, detecting, and processing subsequent voice inputs from the user.

In some implementations, the leader determination is performed in response to a user's voice input to awaken the device (e.g., a voice input that includes the hotword). In some implementations, the leader determination may be performed in response to each voice input from the user or at intervals of a predefined number of voice inputs from the user (e.g., at every 5th voice input from the user). In this way, the leadership determination may be updated to account for changing conditions (e.g., the user moving around, the noise level around the devices change, etc.).

In some implementations, the leader determination based on quality scores may be overridden based on additional criteria. That is, a device may be determined to be the leader even if it would otherwise not be determined to be the leader based merely on the quality scores. The additional criteria may include, for example, a type of request or command in the voice input and a state of the device. For example, if the voice input includes a request to play or stop playing a video, then the request is of particular relevance to an electronic device **1002** that is capable of playing videos (e.g., a smart TV, a cast device) and is not particularly relevant to an electronic device **1002** that is an audio-only device (e.g., sound system, portable speaker). When the request is of particular relevance to an electronic device **1002**, then that electronic device **1002** may be determined to be the leader even if the quality score for that device is not the highest. In some implementations, if the request is of particular relevance to multiple electronic devices in the space, then the one with the best quality score amongst the devices to which the request has particular relevance is determined to be the leader. As another example, if the command in the voice input is a command to change device state (e.g., "screen off"), then a device in a state to which the command is applicable (e.g., a smart TV whose screen is currently on) may be determined to be the leader despite a lower quality score, and a device in a state to which the command is not applicable (e.g., a smart TV whose screen is already off) is passed over for leadership. Further, if the request/command is not of particular relevance to any of the devices **1002** (e.g., the devices **1002** are audio-only voice interface devices that can communicate with a non-voice-interface cast device through the voice interface server **112**, and the command is a request to play video via the cast device), the leadership determination is determined based on quality scores as described above. In some implementations, multiple leadership determination criteria (e.g., quality scores, relevance of command, and state of the device, as described above) may be implemented as a multi-step leadership determination (e.g., determine relevance of the command and the device state to narrow leadership candidates, then determine leader based on quality scores; determine device with highest score, then check if command relevance or device state criteria apply to the other devices) or as a weighted determination (e.g., each criterion is accounted for and weighted in a weighted score).

FIG. 4B illustrates a result of the device leadership negotiation amongst the electronic devices **1002** in FIG. 4A.

Electronic device **1002-1** determines that its quality score is the best amongst the three electronic devices **1002**, and it assumes leadership for outputting responses to the user. Electronic devices **1002-2** and **1002-3** determine that their quality scores are not the best amongst the three electronic devices **1002**, and power down their speakers **520** and forgo outputting responses to the user.

In some implementations, the electronic devices **1002** that are not the leader also power down their microphones **516**, and thus do not detect further voice inputs from the user; the leader device also is the leader with respect to detection and processing of further voice inputs from the user. In some implementations, the electronic devices **1002** that are not the leader maintain power to their microphones **516** and provide assistance re detection of further voice inputs from the user. For example, the leader device and the non-leader devices receive and record respective samples of a voice input and transmit them to the voice assistance server **112**, which may use the multiple samples together to better process the voice input (e.g., aggregate the samples, choose the best sample for processing, compare the samples to get the best parts of each).

In some implementations, the electronic devices **1002** repeat the leadership determination process as described above after each voice input from the user. In some implementations, the electronic devices **1002** repeat the leadership determination process as described above after intervals of a predefined number of voice inputs (e.g., determine a leader again at every 5th voice inputs or at every 10th voice input).

In some implementations, one or more of the electronic devices **1002** output a visual response after a leader is determined (e.g., along with the audio response to the voice input, the determined leader displays a notification or a predefined LED light pattern indicating that it is the leader).

In some implementations, each of the electronic devices **1002** outputs a visual indication (e.g., a predefined LED light pattern) that it is listening whenever it is listening to a user (e.g., receiving and detecting the user's voice input), and then just the leader device outputs the response to the voice input per the leadership determination. For example, when a user is speaking a voice input, the electronic devices **1002-1** thru **1002-3** display the visual indication to indicate that they are listening while receiving the voice input, and afterwards just the leader device **1002-1** outputs the response per the leadership determination.

In some implementations, an electronic device **1002** indicates that it is listening by displaying using a group of LEDs a characteristic visual pattern, such as one of the patterns shown in FIG. 4G of the '566 application (duplicated as FIG. 8D below). The visual patterns can be static (unchanging) or dynamic (giving the impression of movement through changes in the color, brightness, duty cycle, etc. of individual LEDs). In some implementations, an electronic device **1002** indicates that it is the leader among multiple listening devices by displaying another characteristic visual pattern using its LEDs. Similarly, multiple listening devices engaged in a negotiation to determine which device should be the leader can indicate that the negotiation is ongoing by displaying another characteristic visual pattern using their respective LEDs.

Example Process for Multi-User Personalization

FIG. 5 is a flow diagram illustrating a method **1100** of responding to a voice input of a user in accordance with some implementations. The method **1100** is implemented at an electronic device **190** with one or more microphones, a

speaker, one or more processors and memory storing one or more programs for execution by the one or more processors.

The electronic device **190** receives **(1102)** via the one or more microphones **516** a first voice input from a user. The electronic device **190** captures and records a sample of the first voice input via the microphone **516**. In some implementations, the first voice input is a freeform voice input, in the sense that the voice input need not be strictly limited to a predefined set of words and phrases within a predefined syntax; the voice input may be a natural language voice input, for example.

The electronic device **190** compares **(1104)** the first voice input to a set of voice models associated with a plurality of users associated with the electronic device **190**, where the voice models are trained to the electronic device. The user identification module **560** compares the recorded sample of the first voice input to voice models in voice models data **550**. The voice models data **550** include voice models or voice fingerprints of one or more users in the user domain that are associated with the electronic device **190** (e.g., users registered to the electronic device **190**). In some implementations, the voice models are trained by the voice model training module **554** to the electronic device **190**.

Based on the comparing, the electronic device **190** determines **(1106)** that the first voice input corresponds to a first user of the plurality of users. For example, the user identification module **560** identifies a voice model in voice models data **550** that best matches the first voice input, and in accordance with the identification of the match determines that the user speaking the first voice input is the user to which the matching voice model corresponds. In some implementations, the user identification module **560** also determines a confidence level or some other similar measure of the quality or closeness of the match between a voice model and the voice input, and identifies a match only if the match is the best and the confidence level is above a predefined threshold.

The electronic device **190** presents **(1108)** a response in accordance with the determination. In some implementations, the electronic device **190** generates the response in accordance with the determination and with the first voice input. The response includes an indication of the identity of the identified first user, to let the first user know that he has been identified. The response also includes a response to any voice request in the first voice input. For example, in response to a hotword voice input to awaken the electronic device **190**, the electronic device **190** generates and presents (e.g., outputs) a voice greeting that includes the first user's name (e.g., "Hello David," "Good morning Mary"). The response may include a voice output and/or instructions to another device to perform an operation. For instructions to perform an operation, the indication of the identified user's identity is included, so that the operation is performed with respect to the identified user. For example, if the operation is playing music from a music subscription, the device plays music from the identified user's music subscription.

In some implementations, the response is an audible voice output, and presenting the response includes outputting the audible voice output. The electronic device **190** outputs the audible voice output through the speaker **520**.

In some implementations, the electronic device determines a user request based on the first voice input, and generates the response in accordance with the determined user request. In some implementations, generating the response in accordance with the determination and with the first voice input includes these determining and generating operations. In addition to identifying the speaker of the voice

25

input, the electronic device 190 determines the user request in the voice input (e.g., a media playback request, a request for news headlines, a request for emails to be read, etc.), and generates a response that is responsive to the request (e.g., an audible acknowledgement of the request, an audible inquiry to the user for more information needed to fulfill the request).

In some implementations, the determined user request includes a request to the electronic device to awaken, and the response includes an audible greeting, where the greeting includes an indication of an identity of the determined first user. If the voice input includes a hotword (i.e., the voice input is a request to the electronic device 190 to awaken), the generated response includes an audible greeting that includes an identity of the first user (e.g., “Good morning Dave”). The greeting indicates that the electronic device 190 has awakened and indicates the identity of the user as identified by the electronic device 190.

In some implementations, the determined user request includes a request for information, and generating the response in accordance with the determined request includes obtaining the requested information with respect to the first user. If the request is a request for information (e.g., a request to read the user’s emails, request to say out loud the user’s upcoming calendar events), the electronic device 190 accesses the identified user’s information (e.g., the user’s emails, the user’s calendar) to generate the response. In some implementations, obtaining the requested information with respect to the determined first user comprises accessing an account associated with the first user; in order to obtain the identified user’s information for the response, the electronic device 190 accesses the user’s accounts (e.g., the user’s email account, the user’s calendar account). At the electronic device 190, the user account information may be stored in the voice device settings 546.

In some implementations, the electronic device 190 receives a second voice input, compares the second voice input to the set of voice models, based on the comparing, determines that the second voice input corresponds to a second user of the plurality of users, and presents a response in accordance with the determination that the second voice input corresponds to the second user. In some implementations, the electronic device generates the response in accordance with the determination and with the second voice input. If a different user associated with the electronic device 190 speaks a voice input to the electronic device 190, the electronic device 190 identifies the different user in a similar manner as the identification of the first user—comparing the different user’s voice input to the voice models and finding a best match. With the identification of the different user, the electronic device 190 is able to generate a response based on this new identification. Thus, the electronic device 190 is able to switch between associated users by comparing the voice inputs to the voice models and identifying the speaking user based on the comparing, negating a need for the users to explicitly speak a command to switch the focus of the electronic device 190 from one user to another user.

In some implementations, the electronic device 190 receives a third voice input, compares the third voice input to the set of voice models, based on the comparing, determines that the third voice input corresponds to none of the plurality of users, and in accordance with the determination that the third voice input corresponds to none of the plurality of users, restricts functionality of the electronic device. If a user who is not associated with the electronic device 190 speaks a voice input to the electronic device 190, the electronic device 190 receives the voice input and attempts

26

to identify the user. As this user is not associated with the electronic device 190 and thus does not have an associated voice model in the voice models data 550, the electronic device 190 determines that this user is an unidentified user. For this unidentified user, the electronic device 190 may restrict its functionality in order to prevent personal or private information from being accessed. For example, the electronic device 190 may act on only requests from the unidentified user that involve non-private or non-personal information (e.g., request for sports scores, news, stocks, weather). In some implementations, the electronic device 190 may deny the unidentified user access to any functionality of the electronic device 190 and the voice assistant service (e.g., by ignoring the unidentified user’s voice inputs or responding with a response indicating that the user is not authorized). In some implementations, the restriction of functionality to unidentified users is based on settings for the device (e.g., voice device settings 546). The device owner can set, for example, whether to restrict functionality, for unidentified users, entirely or just to functionality that does not involve personal or private information.

In some implementations, the electronic device includes a display, and displays on the display an indication of an identity of the determined first user. If the electronic device 190 includes a display 524, the electronic device 190 may display the identified user’s identify (e.g., display the identified user’s name, display a greeting with the identified user’s name, displaying the identified user’s picture, avatar, or other image associated with the identified user) on the display 524 to indicate that the electronic device 190 has identified a speaker of the voice input and to give the user an opportunity to verify the identification.

In some implementations, the electronic device includes a display, and displays on the display a visual user interface associated with the determined first user. If the electronic device 190 includes a display 524, the electronic device 190 may display a visual user interface identified with the identified user (e.g., a user interface with a theme associated with the identified user applied). In some implementations, the electronic device 190 displays on the LEDs 522 a pattern associated with the identified user.

In some implementations, the electronic device 190 receives the voice input (e.g., capture a sample of the voice input), as in step 1102, and sends the voice input to the voice assistance server 112. The voice assistance server 112 compares the voice input to a set of voice models (e.g., voice models in user voice profiles 744) associated with a plurality of users associated with the electronic device 190, and based on the comparing, determines the identity of the user who spoke the voice input, similar to steps 1104 and 1106. The voice assistance server 112 generates the response to the voice input, and transmits the generated response to the electronic device 190, which outputs the response. In some implementations, the voice assistance server 112 identifies the user, and transmits the user identification result to the electronic device 190, which generates and outputs a response in accordance with the identification result received from the voice assistance server 112.

In some implementations, the electronic device 190 receives the voice input (e.g., capture a sample of the voice input), as in step 1102, performs pre-processing on the voice input, and sends the pre-processed voice input to the voice assistance server 112. The pre-processing may include identification of the speaking user, as in steps 1104 and 1106, and the identification result accompanies the voice input to the voice assistance server 112. Alternatively, the user identification is performed by the voice assistance server 112. The

27

voice assistance server 112 receives the pre-processed voice input, identifies the speaking user if not already performed by the electronic device 190 (similar to steps 1104 and 1106), generates the response to the voice input, and transmits the generated response to the electronic device 190, which outputs the response.

In some implementations, the electronic device 190 receives the voice input (e.g., capture a sample of the voice input) and either sends the voice input to the voice assistance server 112 or performs pre-processing and sends the pre-processed voice input to the voice assistance server 112. The voice assistance server 112 compares the voice input to a set of voice models (e.g., voice models in user voice profiles 744) associated with a plurality of users associated with the electronic device 190, and based on the comparing, determines the identity of the user who spoke the voice input, as in steps 1104 and 1105. The voice assistance server 112 generates the response to the voice input and transmits the generated response to the electronic device 190, which outputs the response. In some implementations, the voice assistance server 112 identifies the user, and transmits the user identification result to the electronic device 190, which generates and outputs a response in accordance with the identification result received from the voice assistance server 112.

As described above, the speaking user is identified by matching a voice model or voice fingerprint to the voice input. In some implementations, the electronic device 190 is configured to perform a process for capturing and training voice models or voice fingerprints. For example, when a user is to be associated with the electronic device 190 (e.g., registered to the device), the electronic device 190 (e.g., voice model training module 554) may initiate and guide the user to be associated with the electronic device 190 through a step-by-step process to capture and train a voice model of the user. During the process, the electronic device 190 may prompt the user to speak one or more words or phrases (e.g., the hotword, a phrase or sentence that includes multiple distinct phonemes) in order to generate and train the voice model. The voice model training module 554 may direct the LED control module 540 to illuminate various visual patterns on the LEDs 522 to indicate to the user a status of the voice model capture and training process. In some implementations, the electronic device 190 stores the trained voice models in voice models data 550, and optionally transmits the trained voice models to the voice assistance server 112 for storage (e.g., in user voice profiles 744).

In some implementations, a given user has voice models for each electronic device 190 with which the user is associated. The voice models at the respective devices 190 are trained to the particular device, thus accounting for the particular devices and the environments surrounding the respective devices.

In some implementations, the voice model training process is performed by the voice assistance server 112 in conjunction with the electronic device 190. For example, when a user is to be associated with an electronic device 190, a voice model training module (not shown) at the voice assistance server 112 transmits instructions and audio outputs to the electronic device 190. The electronic device 190 executes the instructions and output the audio outputs to guide the user through the training process similar to the training process performed by the voice model training module 554 described above. The electronic device 190 captures the user's spoken voice inputs to the training process and transmits the captured voice inputs to the voice assistance server 112, which generates the voice models and

28

stores them in user voice profiles 744. In some implementations, the voice assistance server 112 transmits to the electronic device 190 voice models of users associated with the electronic device 190 for storage (e.g., in voice models data 550) and local use.

In some implementations, when a voice model is generated and trained for a user, an authentication token is also generated for the user. The authentication token authenticates the user to the voice assistance server 112. The speaker's voice input is matched to voice models associated with different users. When a matching voice model is found for the speaker's voice input, the authentication token for the user account associated with the matching voice model is used. In some implementations, authentication tokens are pre-generated but initially not associated with any voice model. The training process associates a voice model trained as a result of the training process to a pre-generated authentication token. In some implementations, after the electronic device 190 identifies the speaker of a voice input, the voice input (or the data containing the user request and associated parameters determined from the voice input) is transmitted to the voice assistance server 112 along with the authentication token associated with the speaker; the authentication token authenticates the speaker to the voice assistance server 112. In some implementations, a speaker may be associated with an authentication token of a user who has granted permission to any speaker. For example, a user associated with an electronic device 190 may configure his account to allow anyone to use his music service login credentials, so that other users can access any connected output device (e.g., speakers) near the user's device 190.

In some implementations, an electronic device 190 identifies the speaking user by detecting a nearby key device associated with the speaking user (e.g., using Bluetooth low energy or ultrasonic RF).

In some implementations, a content host 114 generates the audible response to a voice input. For example, the voice input may include a request for the balance in a bank account associated with the user (e.g., "what is the balance in my Chase account?"). For such a voice input, the electronic device 190 or the voice assistance server 112 identifies the speaking user, as described above. After identifying the user and obtaining the bank account information (from, e.g., user data 742), the electronic device 190 or the voice assistance server 112 transmits a query for the account balance (e.g., through an application programming interface (API)) to a content host 114 associated with the bank where the bank account is located (e.g., the bank's database system). The bank content host obtains the balance from its databases and generates an audible response responsive to the request. The bank content host transmits the audible response to the electronic device 190 or to the voice assistance server 112 (which then transmits the audible response to the electronic device 190), and the electronic device 190 outputs the audible response. In this way, transmission and storage of account data in numerical form over networks and on the device is reduced, which promotes security and privacy of user data.

Example Process for Device Leadership Negotiation

FIG. 6 is a flow diagram illustrating a method 1200 of negotiating device leadership among multiple voice interface devices, in accordance with some implementations. The method 1200 is implemented at an electronic device (e.g., device 1002-1, 1002-2, or 1002-3) of two or more electronic

29

devices **190** (electronic devices **1002**), where each of the two or more electronic devices includes one or more microphones, a speaker, one or more processors and memory storing one or more programs for execution by the one or more processors.

The electronic device **190** detects (**1202**) a voice input. The electronic device **190** captures and records a sample of the voice input from a user via the microphone **516**.

The electronic device **190** determines (**1204**) a quality score for the detected voice input. The electronic device **190** determines a score, rating, or some other measure of the quality of the captured voice input sample. The quality of the sample may be affected by multiple factors, including the user's distance from the device, the volume at which the user spoke the voice input, the surrounding environment and noise, the capabilities of the device, and so on. In some implementations, criteria for evaluating the quality of a voice sample include, for example, loudness or amplitude of the sample, presence or absence of noise (e.g., cross-talk, hissing) and corresponding level, presence or absence of distortion (e.g., echo) and corresponding level, frequency profile of the sample, and so forth.

The electronic device **190** communicates (**1206**) the quality score to the other devices of the plurality of electronic devices, and receives (**1208**) quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices. The electronic device **190** transmits the score for its voice input sample to the other devices of the plurality of electronic devices, and receives scores from the other devices for their respective samples of the voice input.

In accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, the electronic device outputs (**1210**) an audible response and/or a visual response (e.g., a notification or a LED light pattern) to the detected voice input, and the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input. In some implementations, the device with the highest score (or more generally, the best score) amongst the devices with respect to the voice input optionally pre-processes its respective voice input sample, and transmits the voice input sample or the pre-processed voice input sample to the voice assistance server **112**, which generates a response, which includes audible output (e.g., voice confirmation of the user request, reading out of the user-requested information), to the voice input and transmits the response to the devices. Alternatively, the device with the highest score generates the response to the voice input. In either situation, the device with the highest score (e.g., device **1002-1**) outputs the response, and the other devices (e.g., devices **1002-2** and **1002-3**), which have the lower scores, do not output the response. In some implementations, the electronic device that is the leader (e.g., the device with the highest score in the example here) also is the leader with respect to processing or pre-processing subsequent voice inputs from the user.

In accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, the electronic device **190** forgoes outputting a response to the detected voice input. If a device determines that its score is not the highest amongst the devices with respect to the voice input (e.g., devices **1002-2** and **1002-3**), the device does not output a response to the voice input. For example, the devices with the lower

30

scores power down their speakers **520**. In some implementations, the devices with the lower scores also power down their microphones **516** so that just the device with the highest score is detecting further voice inputs. In some implementations, in accordance with the determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, the electronic device **190** forgoes outputting an audible response to the detected voice input, and may output a visual response (e.g., a visual indication that the device was not determined to be the leader).

In some implementations, the plurality of electronic devices is communicatively coupled through a local network, and the communicating and receiving are performed through the local network. The electronic devices **190** are communicatively coupled through a local network (e.g., local network **104**), and may transmit their scores to each other through the local network. For example, a device may multi-cast or broadcast its score through the local network.

If at least one of the devices is not on the local network, then the devices may transmit their scores to the voice assistance server **112**. In some implementations, the device leadership moderator module **780** transmits the received scores to the devices, and the devices determine whether their respective scores are the highest. In some implementations, the device leadership moderator module determines which score is the highest and transmits a result of the determination to the devices. In some implementations, a hub device may include a module similar to the device leadership moderator module **780** and serve in a similar role.

In some implementations, the quality score includes a confidence level of detection of the voice input; the quality score is a confidence level value for the voice input sample. In some implementations, the quality score includes a signal-to-noise rating of detection of the voice input; the quality score is a signal-to-noise rating or value (e.g., signal-to-noise ratio) for the voice input sample.

In some implementations, the electronic device **190** recognizes a command in the voice input, and in accordance with a determination that a type of the command is related to the first electronic device, outputs an audible and/or a visual response to the detected voice input. If the command or request in the voice input is recognized as one with particular relevance to a device of the multiple devices that detected the voice input, then that device outputs a response to the voice input and the other devices forgo outputting the response, even if the device to which the command has particular relevance does not have the highest score among the devices with respect to the voice input. Examples of commands or requests that have particular relevance include video playback requests (e.g., for cast or TV devices) and music playback requests (e.g., for audio systems or speaker systems). Additionally, a command or request may have particular relevance to a device if the command or request identifies the particular device that the user wishes to fulfill the request (e.g., the command "Play video on my bedroom TV" has particular relevance to the TV device in the bedroom because the command explicitly identifies that device), or the command/request includes a device state parameter that is applicable to the current state of the device (e.g., a "screen off" command is relevant to a device whose screen is on and not relevant to a device without a screen or whose screen is off).

In this way, the multiple devices capture respective samples of the voice input, and a determination as to which sample is best is made. Based on this determination, one of

31

the devices respond to the voice input and further voice inputs from the user. In some implementations, the multiple devices repeat the process described above for each voice input or at intervals of a predefined number of voice inputs (e.g., determine the leader again at every 5th voice input or at every 10th voice input).

In some implementations, the multiple devices detect the voice input and send their captured voice input samples to the voice assistance server 112. The voice assistance server 112 aggregates the samples or selects the best quality sample for processing or compares the samples to get the best parts of each. In some implementations, when the multiple devices are receiving and detecting the voice input, the devices each display a visual indication (e.g., a predefined LED pattern) that it is listening to the user.

Example Process for Noise Mitigation

FIG. 7 is a flow diagram illustrating a method 1300 of mitigating noise around a voice interface device in accordance with some implementations. The method 1300 is implemented at an electronic device 190 with one or more microphones, a speaker, one or more processors and memory storing one or more programs for execution by the one or more processors. The electronic device 190 is configured to be awaked by any of a plurality of affordances, including a voice-based affordance (e.g., a voice input that includes a hotword).

The electronic device determines (1302) a current noise profile of an environment around the electronic device. The electronic device 190, while it is idle, may run the noise module 558 (e.g., noise determination module 572) to determine a noise profile for its surroundings. In some implementations, determining the current noise profile includes determining a noise level (e.g., volume) of the noise, and capturing and processing the ambient noise. The processing may include, for example, determining frequencies and other parameters of the ambient noise.

The electronic device determines (1304) whether the noise profile interferes with the voice-based affordance. The electronic device 190 compares the determined noise profile with noise profiles in the noise profile store 576 and evaluates the parameters of the noise profile to determine if the noise profile (i.e., the current noise surrounding the electronic device 190) interferes with the voice-based affordance.

In accordance with a determination that the noise profile interferes with the voice-based affordance, the electronic device 190 presents (1306) a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device. The hint directs the user to use another affordance other than the voice-based affordance to awaken the electronic device 190. In some implementations, the hint is also or otherwise includes indication that the surrounding noise interferes with hotword recognition. For example, the electronic device 190 may display a predefined pattern on the LEDs 522 that indicates surrounding noise interference with the voice-based affordance and hints to the user that the user should use another affordance to wake up the device (e.g., pressing a button 512, touching a touch sensor array 514, use another device communicatively coupled to the electronic device 190, such as a smartphone). The indication/hint may be displayed on the display 524 or output as sounds or tones through speaker 520.

In some implementations, wherein the voice-based affordance includes a spoken hotword. The voice-based afford-

32

dance includes the user speaking the hotword to awaken the electronic device 190, as described above.

In some implementations, determining whether the noise profile interferes with the voice-based affordance includes determining whether the noise profile negatively affects recognition of the hotword in voice inputs detected by the electronic device. The electronic device 190 determines if the surrounding noise (represented by the noise profile) would negatively affect detection/recognition of the hotword by the electronic device 190 (e.g., is the noise sufficiently loud to drown out the user's voice input with the hotword, does the noise have a frequency profile that would interfere with the user's voice input with the hotword, would the noise distort the user's voice input with the hotword).

In some implementations, determining whether the noise profile negatively affects recognition of the hotword in voice inputs includes comparing the noise profile to a noise level threshold of a hotword recognition process associated with the electronic device. The recognition analysis module 568 of the electronic device 190 (or the voice assistance server 112) analyzes the hotword recognition module 562 of the electronic device 190 (e.g., when the electronic device 190 is idle) to determine noise thresholds at which noise would cause problems for the algorithm or process used by the module. These thresholds are saved and may be compared to current noise profiles to determine if the current noise profile would pose a problem for the hotword recognition module 562. The analysis of the hotword recognition module is with respect to the electronic device 190, and thus the threshold may be different between devices.

In some implementations, determining whether the noise profile negatively affects recognition of the hotword in voice inputs includes comparing the noise profile to a voice volume level threshold, where the voice volume level threshold is defined based on a predefined distance from the electronic device. The user volume threshold module 570 of the electronic device 190 (or the voice assistance server 112) performs analyses of the user's voice inputs to determine a maximum comfortable volume level at which the user will speak voice inputs, with the determination based on a predefined distance (e.g., a predefined distance that is based on a "typical user distance" beyond which voice inputs attenuate).

In some implementations, determining whether the noise profile negatively affects recognition of the hotword in voice inputs includes comparing the noise profile to a predetermined set of noise profiles associated with the electronic device, wherein the noise profiles of the predetermined set of noise profiles are known to negatively affect recognition of the hotword by the electronic device in voice inputs. The electronic device 190 compares the noise profiles to previous noise profiles (e.g., in noise profiles store 576) that interfered with hotword detection.

In some implementations, the electronic device includes a button, and the affordance of the plurality of affordances other than the voice-based affordance includes activating the button. The electronic device 190 includes a physical push button 512 and the user can awaken the device by activating the button 512 in lieu of speaking a voice input with the hotword.

In some implementations, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device includes illuminating the button on the electronic device corresponding to a button-based affordance of the plurality of affordances. The button 512 may include lighting (e.g., an embedded LED light), and the electronic device 190 may

33

hint the user to use the button **512** instead of speaking the hotword by lighting up the button **512**.

In some implementations, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device includes outputting a predefined audible sound. The electronic device **190** may hint to the user to use an alternate affordance by outputting a predefined sound or tone (or set of sounds or tones) through the speaker **520**.

In some implementations, the electronic device includes a display, and presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device includes displaying the hint on a display. The electronic device **190** may hint to the user to use an alternate affordance by displaying a hint message on the display **524**.

In some implementations, the electronic device includes one or more LED lights, and presenting a hint to a user to use an affordance of the plurality of affordances different from the voice-based affordance to awaken the electronic device includes displaying the hint using the one or more LED lights. The electronic device **190** may hint to the user to use an alternate affordance by displaying a predefined light pattern on the LEDs **522**.

FIGS. **8A** and **8B** are a front view **400** and a rear view **420** of a voice-activated electronic device **190** in accordance with some implementations. The electronic device **190** is designed as warm and inviting, and fits naturally in many areas of a home. The electronic device **190** includes one or more microphones **402** and an array of full color LEDs **404**. The full color LEDs **404** could be hidden under a top surface of the electronic device **190** and invisible to the user when they are not lit. In some implementations, the array of full color LEDs **404** is physically arranged in a ring. Further, the rear side of the electronic device **190** optionally includes a power supply connector **408** configured to couple to a power supply.

In some implementations, the electronic device **190** presents a clean look having no visible button, and the interaction with the electronic device **190** is based on voice and touch gestures. Alternatively, in some implementations, the electronic device **190** includes a limited number of physical buttons (e.g., a button **406** on its rear side), and the interaction with the electronic device **190** is further based on press on the button in addition to the voice and touch gestures.

In some implementations, given simplicity and low cost of the electronic device **190**, the electronic device **190** includes an array of full color light emitting diodes (LEDs) rather than a full display screen. A LED design language is adopted to configure illumination of the array of full color LEDs and enable different visual patterns indicating different voice processing state of the electronic device **190**. The LED Design Language consists of a grammar of colors, patterns, and specific motion applied to a fixed set of full color LEDs. The elements in the language are combined to visually indicate specific device states during the use of the electronic device **190**. In some implementations, illumination of the full color LEDs aims to clearly delineate the passive listening and active listening states of the electronic device **190** among other important states. Placement of the full color LEDs complies with physical constraints of the electronic device **190**, and the array of full color LEDs could be used in a speaker that is made by a third party original equipment manufacturer (OEM) based on specific technology (e.g., Google Assistant).

34

When the array of full color LEDs is used in a speaker that is made by a third party OEM based on specific technology, the full color LEDs and the LED design language are configured to fit a corresponding physical user interface of the OEM speaker. In this situation, device states of the OEM speaker remain the same, while specific visual patterns representing the device states could be varied (for example, the colors of the full color LEDs could be different but are displayed with similar animation effects).

In a voice-activated electronic device **190**, passive listening occurs when the electronic device **190** processes audio inputs collected from its surrounding environment but does not store the audio inputs or transmit the audio inputs to any remote server. In contrast, active listening occurs when the electronic device **190** stores the audio inputs collected from its surrounding environment and/or shares the audio inputs with a remote server. In accordance with some implementations of this application, the electronic device **190** only passively listens to the audio inputs in its surrounding environment without breaching privacy of users of the electronic device **190**.

FIG. **8C** is a top view of a voice-activated electronic device **190** in accordance with some implementations, and FIG. **8D** shows six visual patterns displayed by an array of full color LEDs for indicating voice processing states in accordance with some implementations. In some implementations, the electronic device **190** does not include any display screen, and the full color LEDs provide a simple and low cost visual user interface compared with the a full display screen. The full color LEDs could be hidden under a top surface of the electronic device and invisible to the user when they are not lit. Referring to FIGS. **8C** and **8D**, in some implementations, the array of full color LEDs are physically arranged in a ring.

Reference has been made in detail to implementations, examples of which are illustrated in the accompanying drawings. In the detailed description above, numerous specific details have been set forth in order to provide a thorough understanding of the various described implementations. However, it will be apparent to one of ordinary skill in the art that the various described implementations may be practiced without these specific details. In other instances, well-known methods, procedures, components, circuits, and networks have not been described in detail so as not to unnecessarily obscure aspects of the implementations.

It will also be understood that, although the terms first, second, etc. are, in some instances, used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first device could be termed a second device, and, similarly, a second device could be termed a first device, without departing from the scope of the various described implementations. The first device and the second device are both types of devices, but they are not the same device.

The terminology used in the description of the various described implementations herein is for the purpose of describing particular implementations only and is not intended to be limiting. As used in the description of the various described implementations and the appended claims, the singular forms “a”, “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term “and/or” as used herein refers to and encompasses any and all possible combinations of one or more of the associated listed items. It will be further understood that the terms “includes,” “including,” “comprises,” and/or “comprising,”

35

when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

As used herein, the term “if” is, optionally, construed to mean “when” or “upon” or “in response to determining” or “in response to detecting” or “in accordance with a determination that,” depending on the context. Similarly, the phrase “if it is determined” or “if [a stated condition or event] is detected” is, optionally, construed to mean “upon determining” or “in response to determining” or “upon detecting [the stated condition or event]” or “in response to detecting [the stated condition or event]” or “in accordance with a determination that [a stated condition or event] is detected,” depending on the context.

For situations in which the systems discussed above collect information about users, the users may be provided with an opportunity to opt in/out of programs or features that may collect personal information (e.g., information about a user’s preferences or usage of a smart device). In addition, in some implementations, certain data may be anonymized in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user’s identity may be anonymized so that the personally identifiable information cannot be determined for or associated with the user, and so that user preferences or user interactions are generalized (for example, generalized based on user demographics) rather than associated with a particular user.

Although some of various drawings illustrate a number of logical stages in a particular order, stages that are not order dependent may be reordered and other stages may be combined or broken out. While some reordering or other groupings are specifically mentioned, others will be obvious to those of ordinary skill in the art, so the ordering and groupings presented herein are not an exhaustive list of alternatives. Moreover, it should be recognized that the stages could be implemented in hardware, firmware, software or any combination thereof.

The foregoing description, for purpose of explanation, has been described with reference to specific implementations. However, the illustrative discussions above are not intended to be exhaustive or to limit the scope of the claims to the precise forms disclosed. Many modifications and variations are possible in view of the above teachings. The implementations were chosen in order to best explain the principles underlying the claims and their practical applications, to thereby enable others skilled in the art to best use the implementations with various modifications as are suited to the particular uses contemplated.

What is claimed is:

1. A method, comprising:

at a first electronic device of a plurality of electronic devices, each electronic device of the plurality of electronic devices comprising one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors:

- detecting a voice input;
- determining a quality score for the detected voice input;
- communicating the quality score to the other devices of the plurality of electronic devices;
- receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices;

36

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and

in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

2. The method of claim 1, wherein the plurality of electronic devices is communicatively coupled through a local network; and

the communicating and receiving are performed through the local network.

3. The method of claim 1, wherein the quality score comprises a confidence level of detection of the voice input.

4. The method of claim 1, wherein the quality score comprises a signal-to-noise rating of detection of the voice input.

5. The method of claim 1, further comprising:

recognizing a command in the voice input; and

in accordance with a determination that a type of the command is related to the first electronic device, outputting an audible and/or a visual response to the detected voice input.

6. A first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising:

one or more microphones;

a speaker;

one or more processors; and

memory storing one or more programs to be executed by the one or more processors, the one or more programs comprising instructions for:

detecting a voice input;

determining a quality score for the detected voice input;

communicating the quality score to the other devices of the plurality of electronic devices;

receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices;

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and

in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

7. The device of claim 6, wherein the plurality of electronic devices is communicatively coupled through a local network; and

the communicating and receiving are performed through the local network.

8. The device of claim 6, wherein the quality score comprises a confidence level of detection of the voice input.

37

9. The device of claim 6, wherein the quality score comprises a signal-to-noise rating of detection of the voice input.

10. The device of claim 6, further comprising instructions for:

recognizing a command in the voice input; and
in accordance with a determination that a type of the command is related to the first electronic device, outputting an audible and/or a visual response to the detected voice input.

11. A non-transitory computer readable storage medium storing one or more programs, the one or more programs comprising instructions, which, when executed by a first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more microphones, a speaker, and one or more processors, cause the electronic device to perform operations comprising:

detecting a voice input;
determining a quality score for the detected voice input;
communicating the quality score to the other devices of the plurality of electronic devices;
receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices;

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input,

38

wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and

in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

12. The computer readable storage medium of claim 11, wherein the plurality of electronic devices is communicatively coupled through a local network; and the communicating and receiving are performed through the local network.

13. The computer readable storage medium of claim 11, wherein the quality score comprises a confidence level of detection of the voice input.

14. The computer readable storage medium of claim 11, wherein the quality score comprises a signal-to-noise rating of detection of the voice input.

15. The computer readable storage medium of claim 11, further comprising instructions which, when executed by the electronic device, cause the electronic device to perform operations comprising:

recognizing a command in the voice input; and
in accordance with a determination that a type of the command is related to the first electronic device, outputting an audible and/or a visual response to the detected voice input.

* * * * *

Exhibit 3



US011024311B2

(12) **United States Patent**
Mixter et al.

(10) **Patent No.:** **US 11,024,311 B2**

(45) **Date of Patent:** ***Jun. 1, 2021**

(54) **DEVICE LEADERSHIP NEGOTIATION
AMONG VOICE INTERFACE DEVICES**

(71) Applicant: **GOOGLE LLC**, Mountain View, CA
(US)

(72) Inventors: **Kenneth Mixter**, Los Altos Hills, CA
(US); **Diego Melendo Casado**,
Mountain View, CA (US); **Alexander**
Houston Gruenstein, Mountain View,
CA (US); **Terry Tai**, New York, NY
(US); **Christopher Thaddeus Hughes**,
Redwood City, CA (US); **Matthew**
Nirvan Sharifi, Kilchberg (CH)

(73) Assignee: **GOOGLE LLC**, Mountain View, CA
(US)

(*) Notice: Subject to any disclaimer, the term of this
patent is extended or adjusted under 35
U.S.C. 154(b) by 12 days.

This patent is subject to a terminal dis-
claimer.

(21) Appl. No.: **16/786,943**

(22) Filed: **Feb. 10, 2020**

(65) **Prior Publication Data**

US 2020/0211556 A1 Jul. 2, 2020

Related U.S. Application Data

(63) Continuation of application No. 16/159,339, filed on
Oct. 12, 2018, now Pat. No. 10,559,306, which is a
(Continued)

(51) **Int. Cl.**
G10L 15/22 (2006.01)
G10L 15/32 (2013.01)

(Continued)

(52) **U.S. Cl.**

CPC **G10L 15/22** (2013.01); **G10L 15/32**
(2013.01); **G10L 25/60** (2013.01); **G10L**
2015/088 (2013.01); **G10L 2015/223** (2013.01)

(58) **Field of Classification Search**

CPC **G10L 15/22**; **G10L 15/32**; **G10L 25/60**;
G10L 2015/088; **G10L 2015/223**
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,455,889 A * 10/1995 Bahl **G10L 15/142**
704/236
6,961,704 B1 * 11/2005 Phillips **G10L 13/10**
704/267

(Continued)

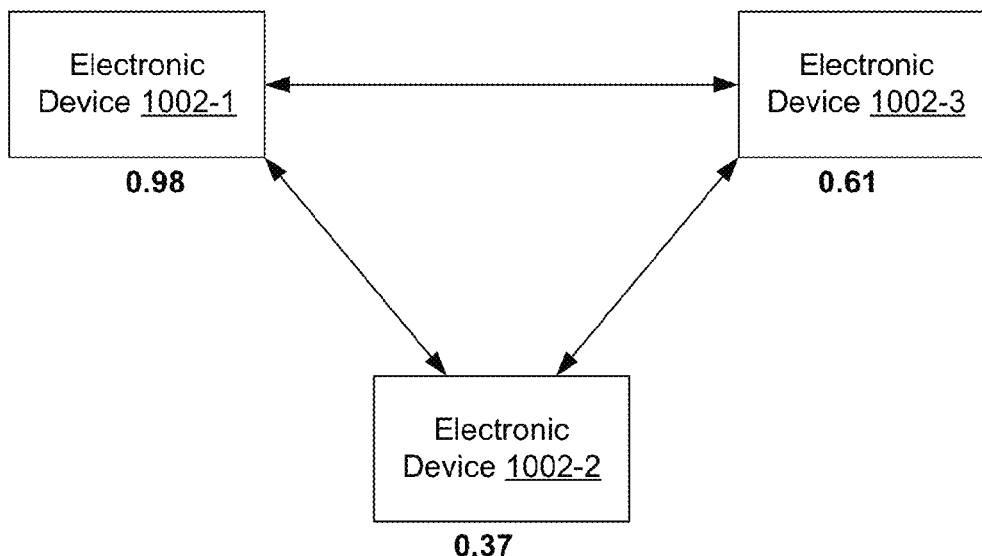
Primary Examiner — Michael Colucci

(74) *Attorney, Agent, or Firm* — Marshall, Gerstein &
Borun LLP

(57) **ABSTRACT**

The various implementations described herein include meth-
ods and systems for determining device leadership among
voice interface devices. In one aspect, a method is per-
formed at a first electronic device of a plurality of electronic
devices, each having microphones, a speaker, processors,
and memory storing programs for execution by the proces-
sors. The first device detects a voice input. It determines a
device state and a relevance of the voice input. It identifies
a subset of electronic devices from the plurality to which the
voice input is relevant. In accordance with a determination
that the subset includes the first device, the first device
determines a first score of a criterion associated with the
voice input and receives second scores of the criterion from
other devices in the subset. In accordance with a determi-
nation that the first score is higher than the second scores, the
first device responds to the detected input.

20 Claims, 10 Drawing Sheets
(1 of 10 Drawing Sheet(s) Filed in Color)



Related U.S. Application Data

continuation of application No. 15/788,658, filed on Oct. 19, 2017, now Pat. No. 10,102,857, which is a continuation of application No. 15/284,483, filed on Oct. 3, 2016, now Pat. No. 9,812,128, which is a continuation-in-part of application No. 15/088,477, filed on Apr. 1, 2016, now Pat. No. 9,514,752, which is a continuation of application No. 14/675,932, filed on Apr. 1, 2015, now Pat. No. 9,318,107.

(60) Provisional application No. 62/061,830, filed on Oct. 9, 2014.

(51) **Int. Cl.**

G10L 15/08 (2006.01)
G10L 25/60 (2013.01)

(56) **References Cited**

U.S. PATENT DOCUMENTS

7,778,816 B2 * 8/2010 Reynar G06K 9/2054
704/9
8,244,534 B2 * 8/2012 Qian G10L 13/06
704/256.3
8,340,975 B1 * 12/2012 Rosenberger G10L 15/22
704/275

2004/0078191 A1 * 4/2004 Tian G06F 40/263
704/9
2004/0158464 A1 * 8/2004 Baker G10L 15/083
704/231
2004/0236570 A1 * 11/2004 Tato G10L 15/02
704/206
2005/0171766 A1 * 8/2005 Albesano G10L 15/16
704/202
2008/0288256 A1 * 11/2008 Agapi G10L 13/04
704/260
2011/0246181 A1 * 10/2011 Liang G06F 40/289
704/9
2013/0238326 A1 * 9/2013 Kim G10L 15/22
704/231
2014/0163978 A1 * 6/2014 Basye G10L 15/28
704/233
2014/0278435 A1 * 9/2014 Ganong, III G10L 15/22
704/275
2015/0032443 A1 * 1/2015 Karov G06F 16/36
704/9
2015/0248885 A1 * 9/2015 Koulomzin G10L 15/08
704/251
2016/0012465 A1 * 1/2016 Sharp G06Q 20/386
705/14.17
2018/0061419 A1 * 3/2018 Melendo Casado G10L 15/22
2018/0182390 A1 * 6/2018 Hughes G10L 15/02
2018/0196776 A1 * 7/2018 Hershko G06F 13/4291
2019/0304443 A1 * 10/2019 Bhagwan G10L 15/22

* cited by examiner

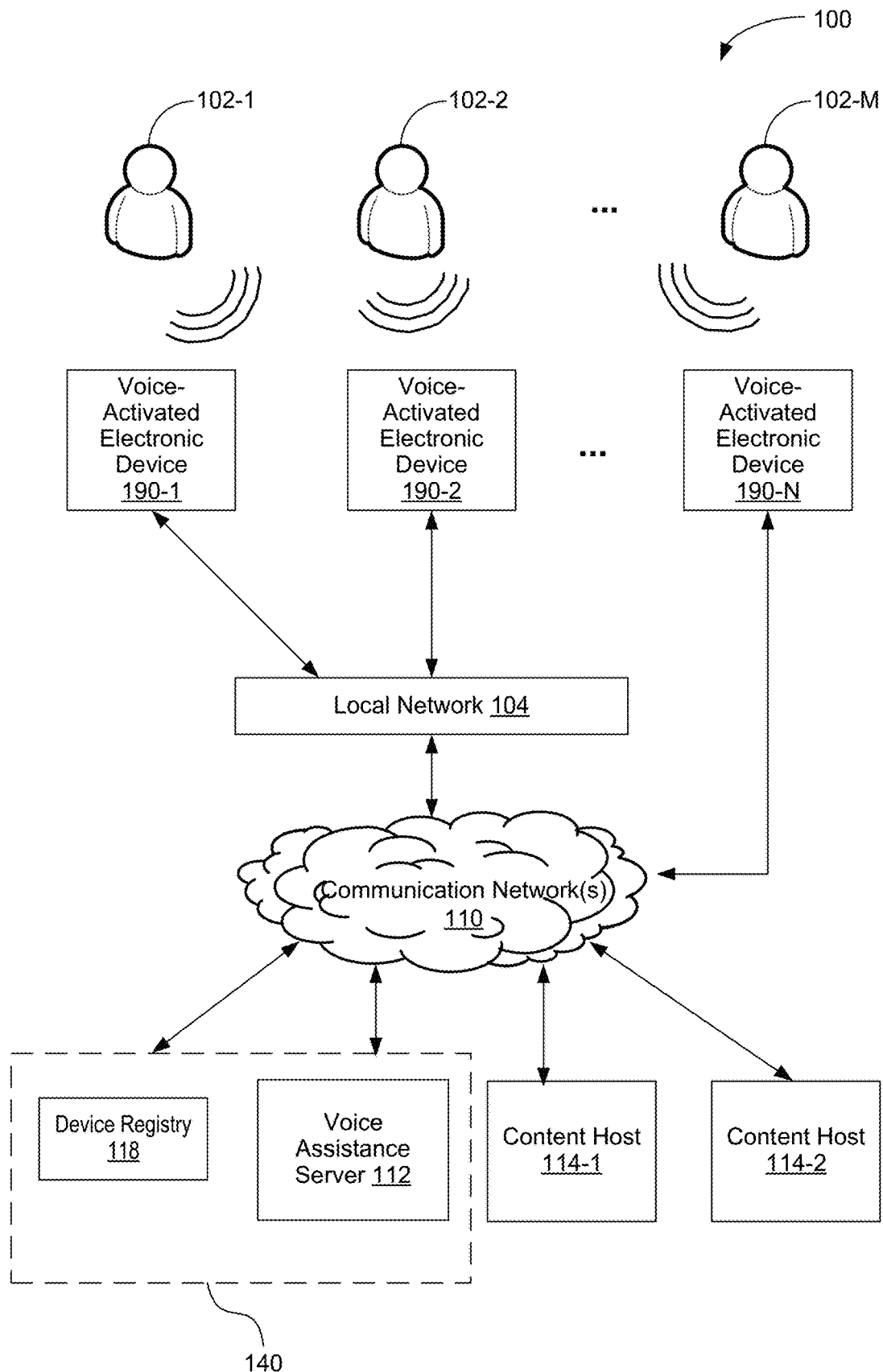


Figure 1

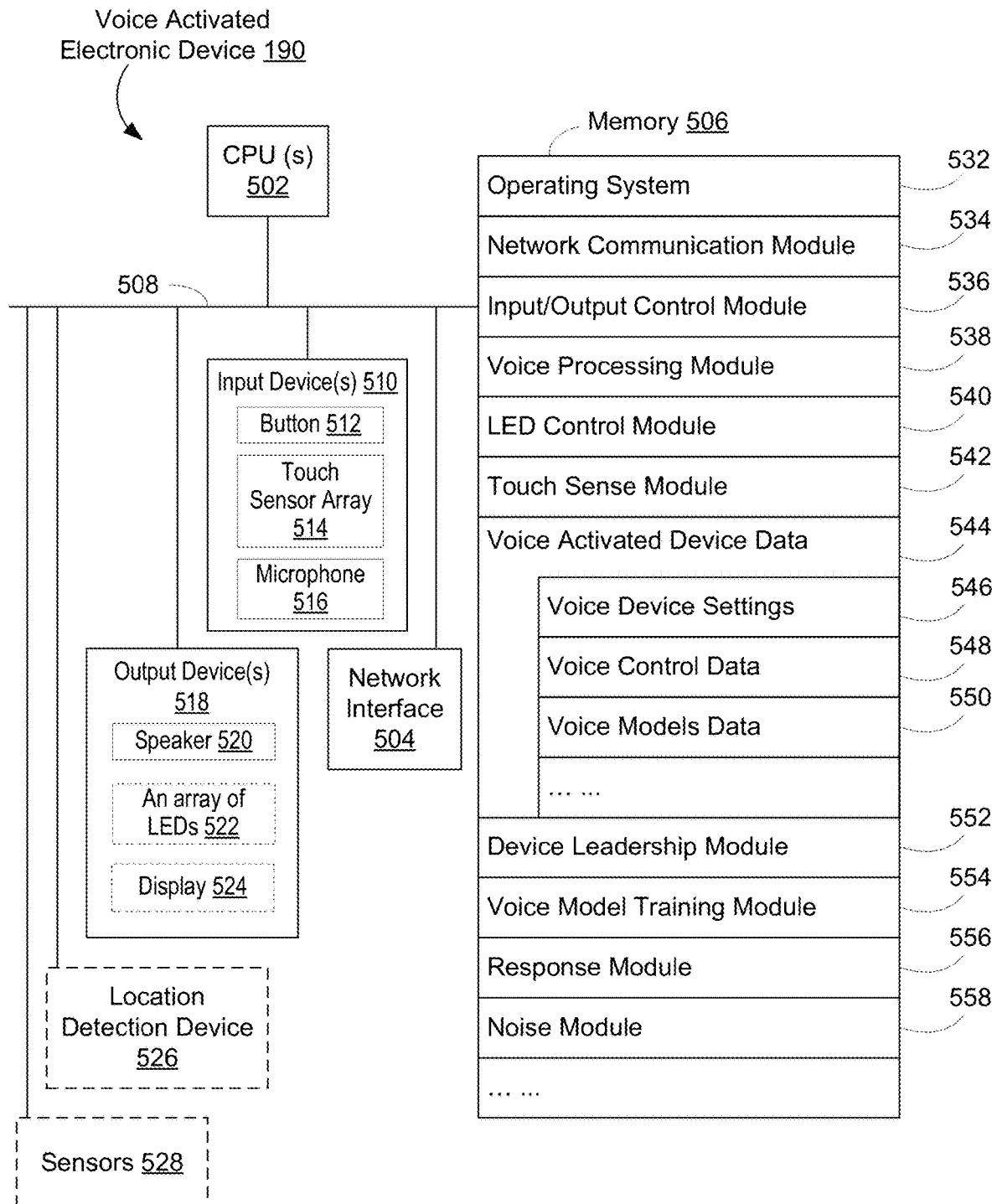
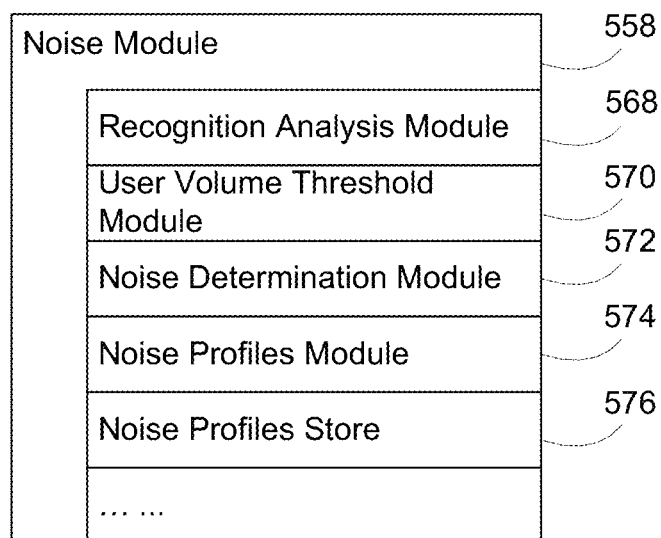
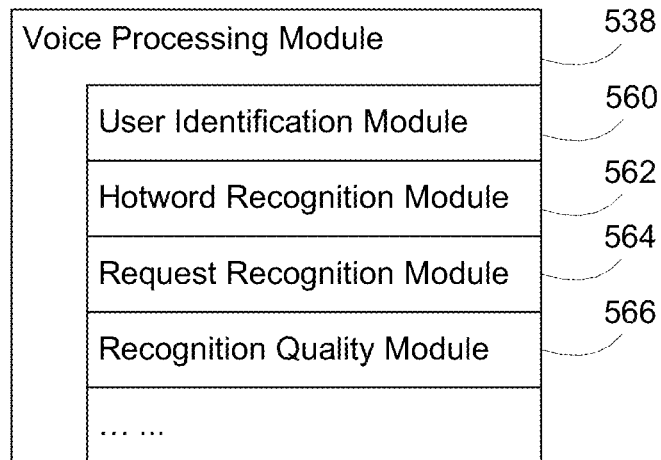


Figure 2A

**Figure 2B**

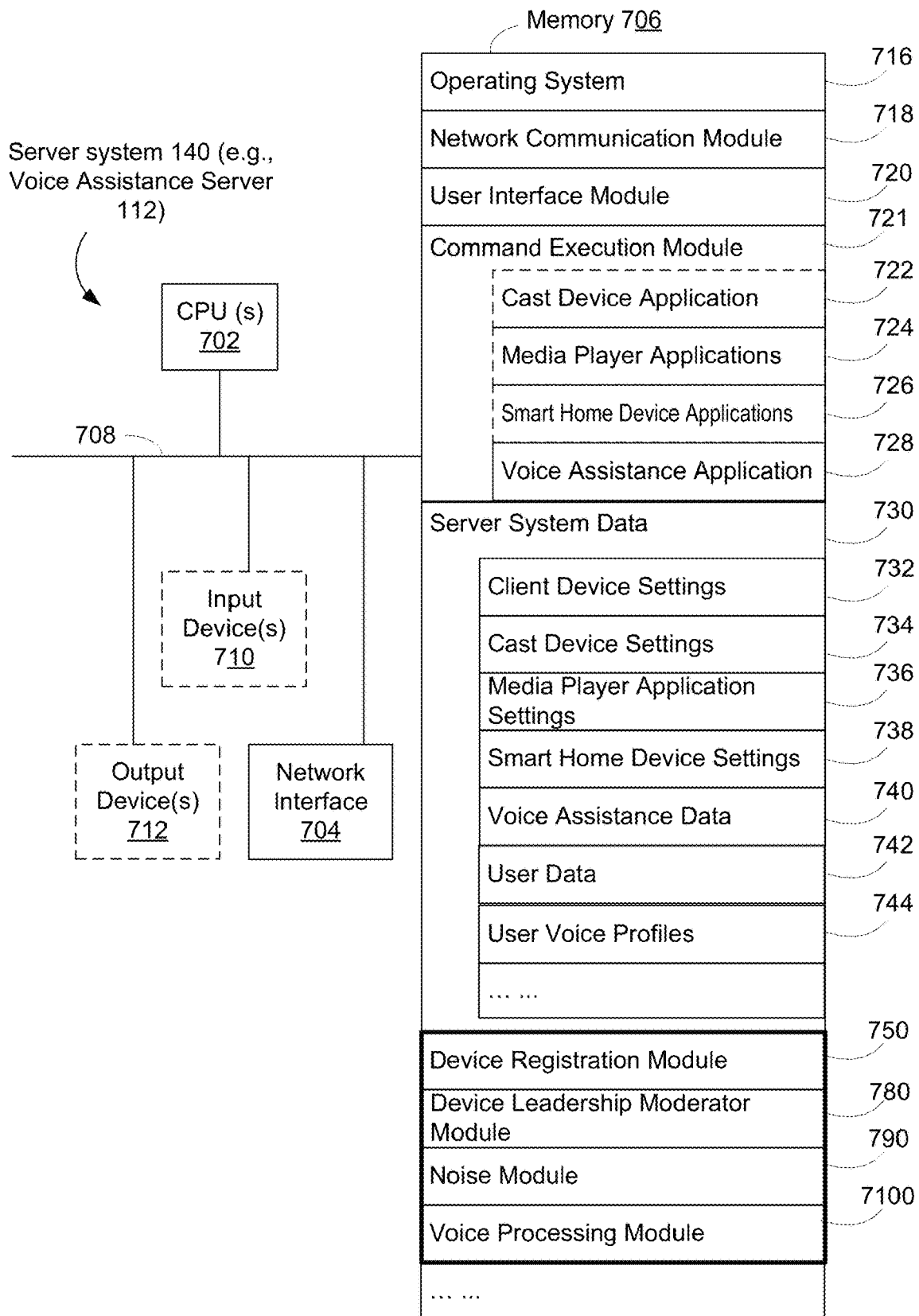


Figure 3

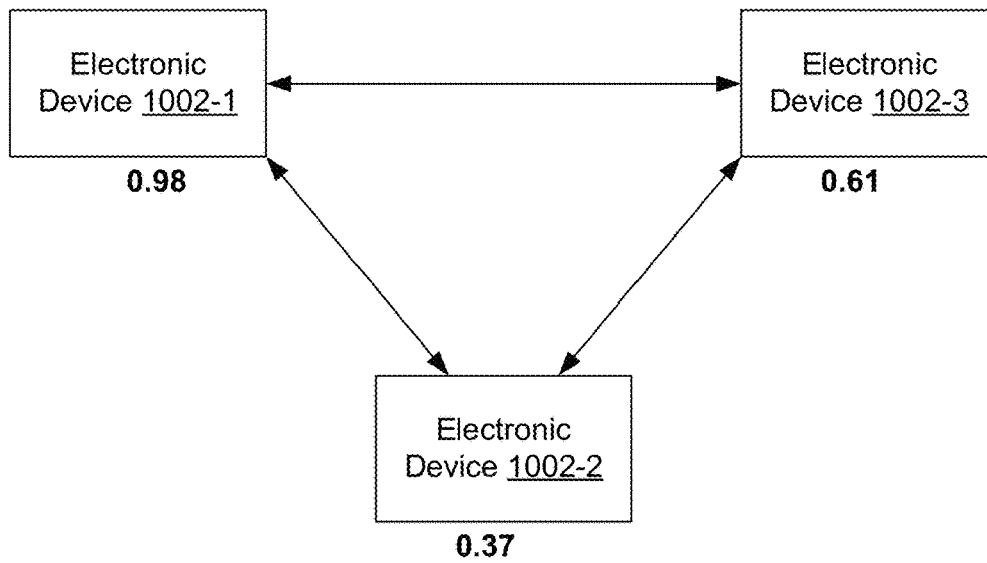


Figure 4A

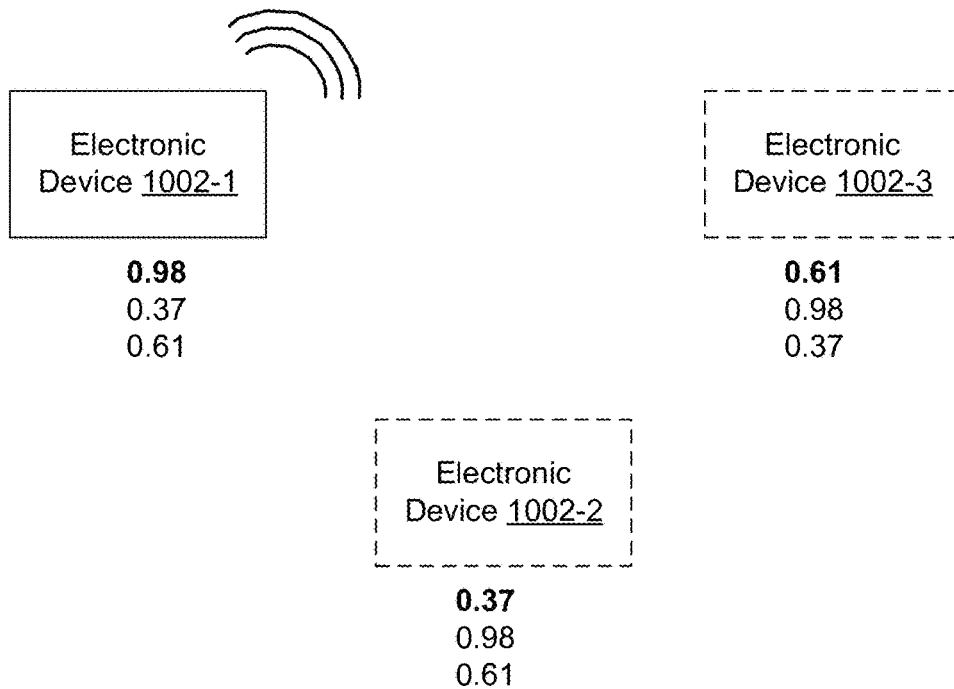


Figure 4B

1100

At an electronic device with one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors:

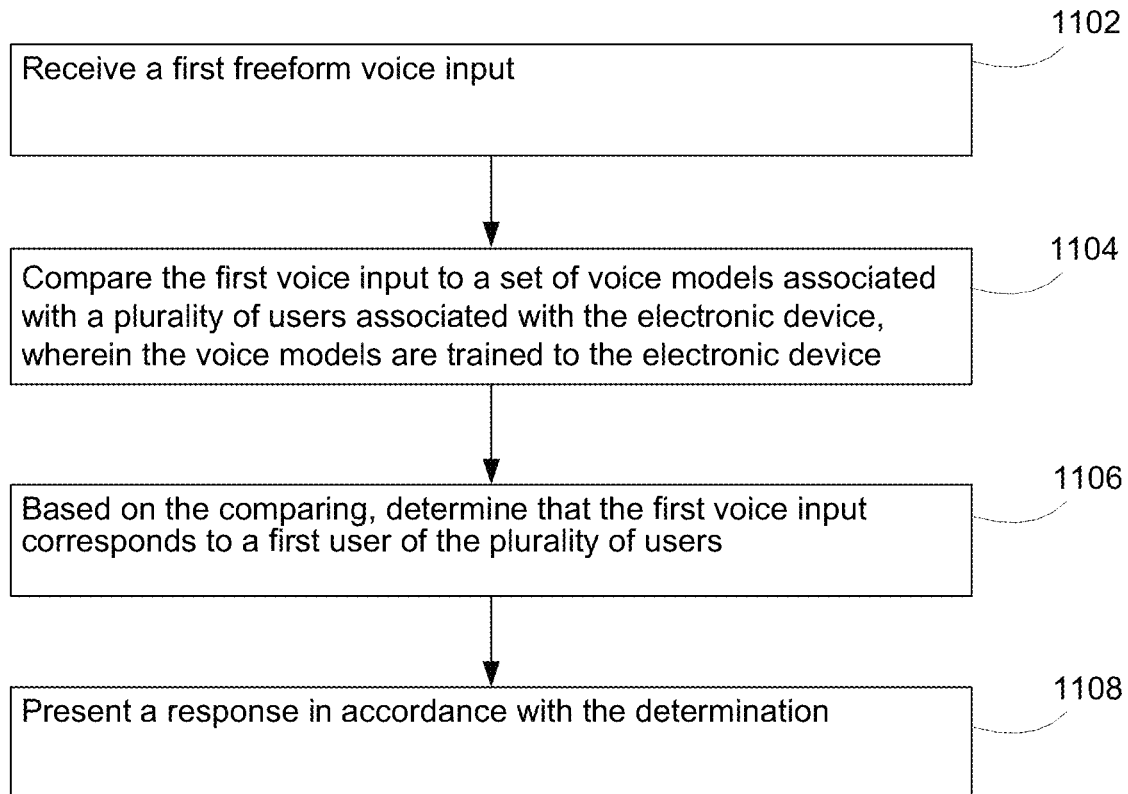
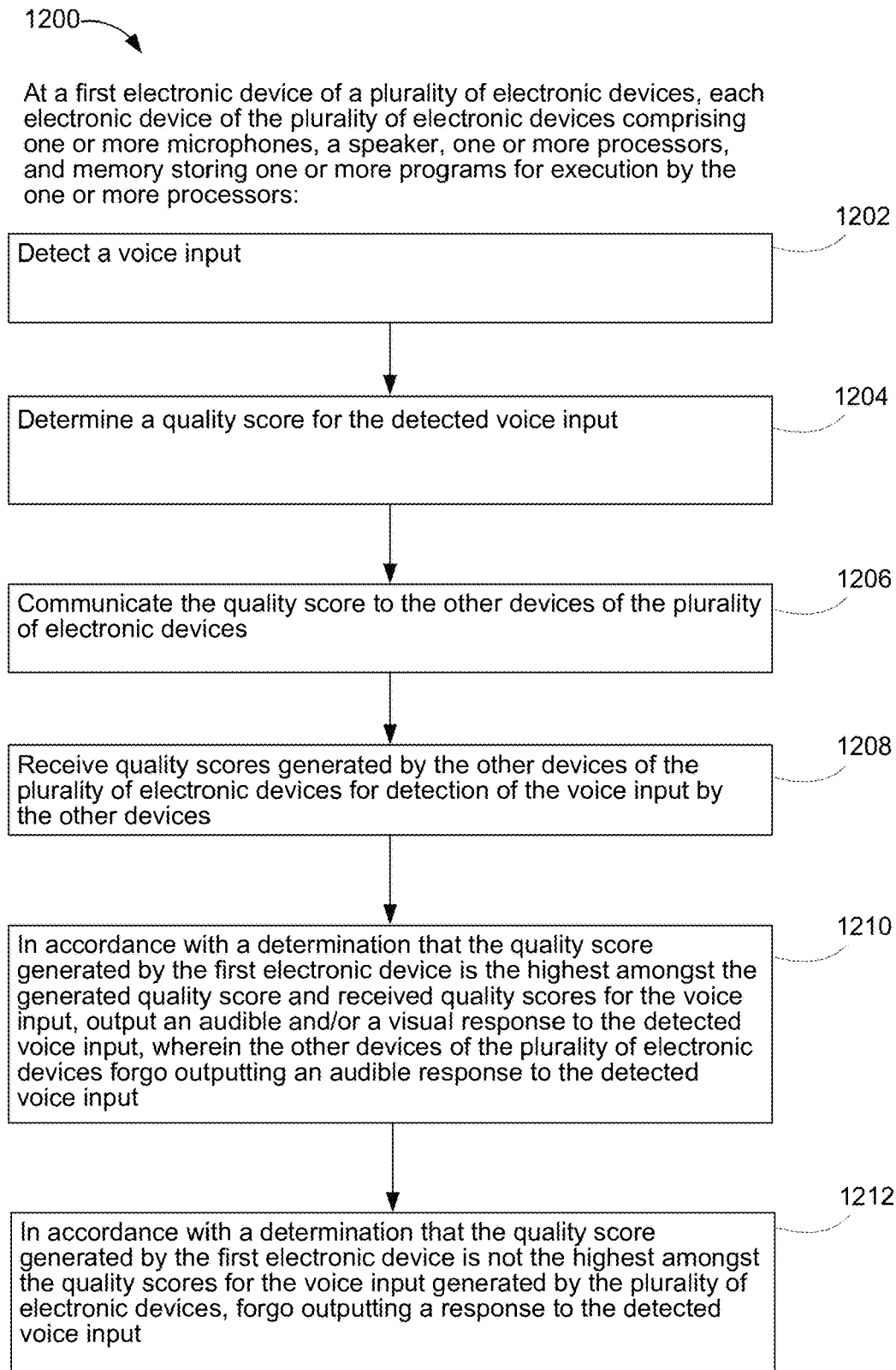
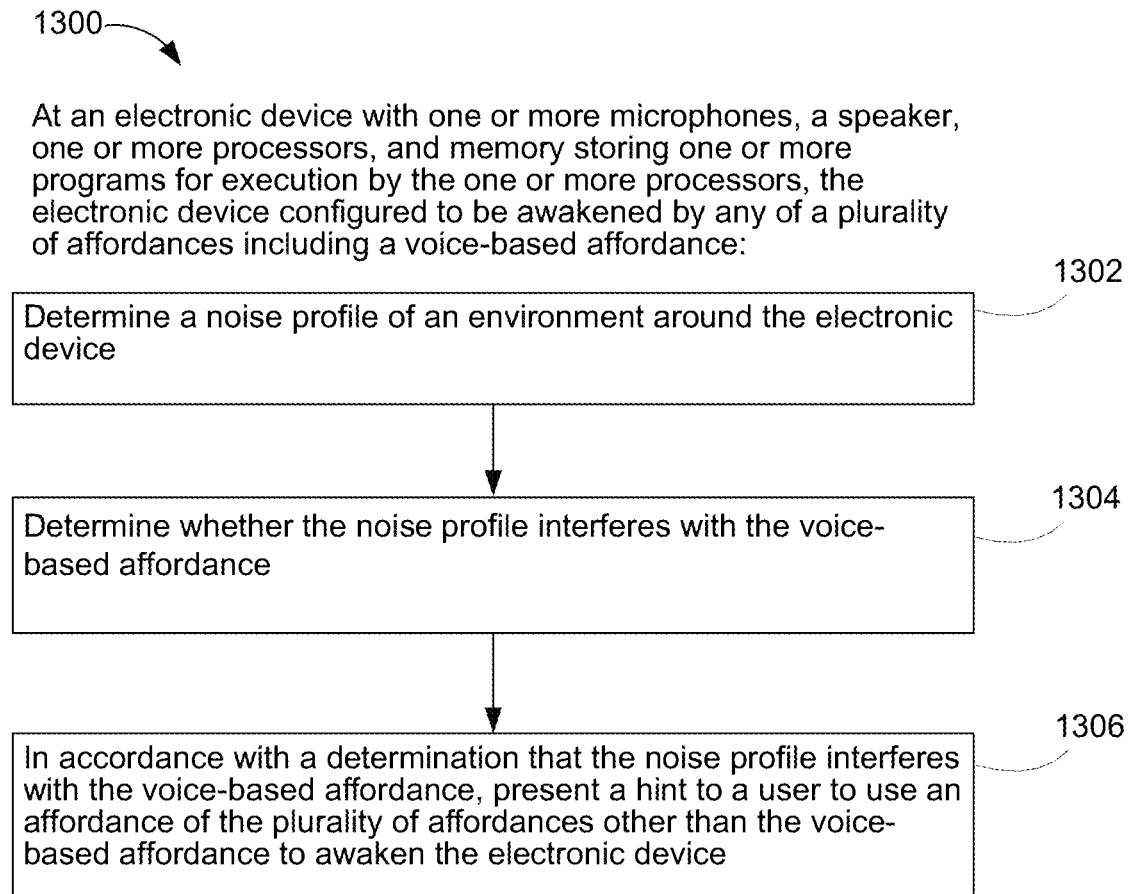


Figure 5

**Figure 6**

**Figure 7**

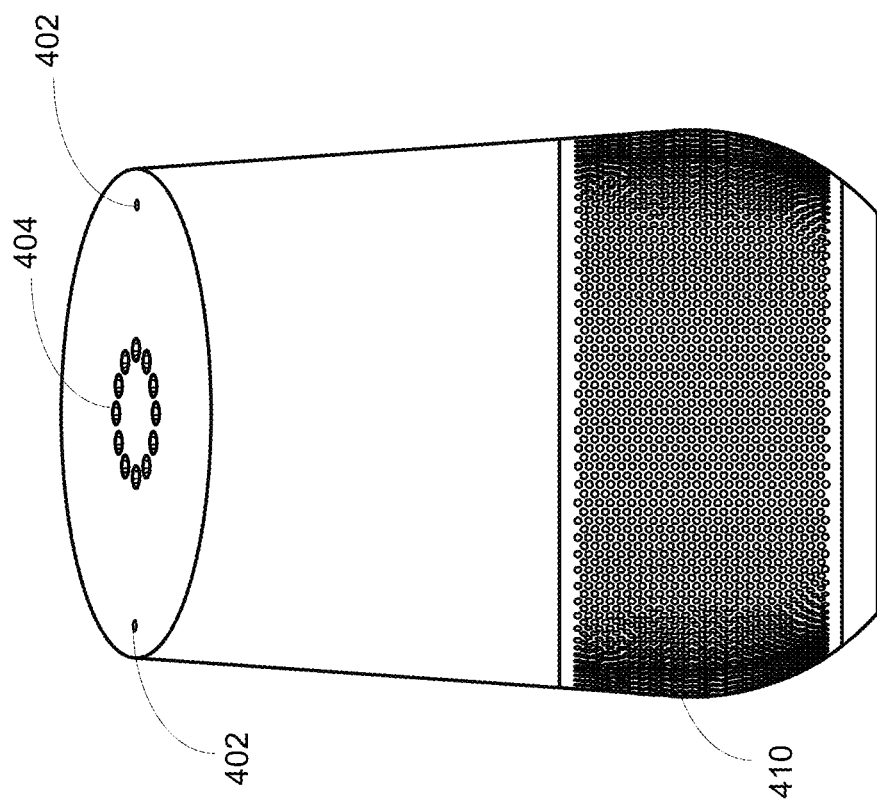


Figure 8A

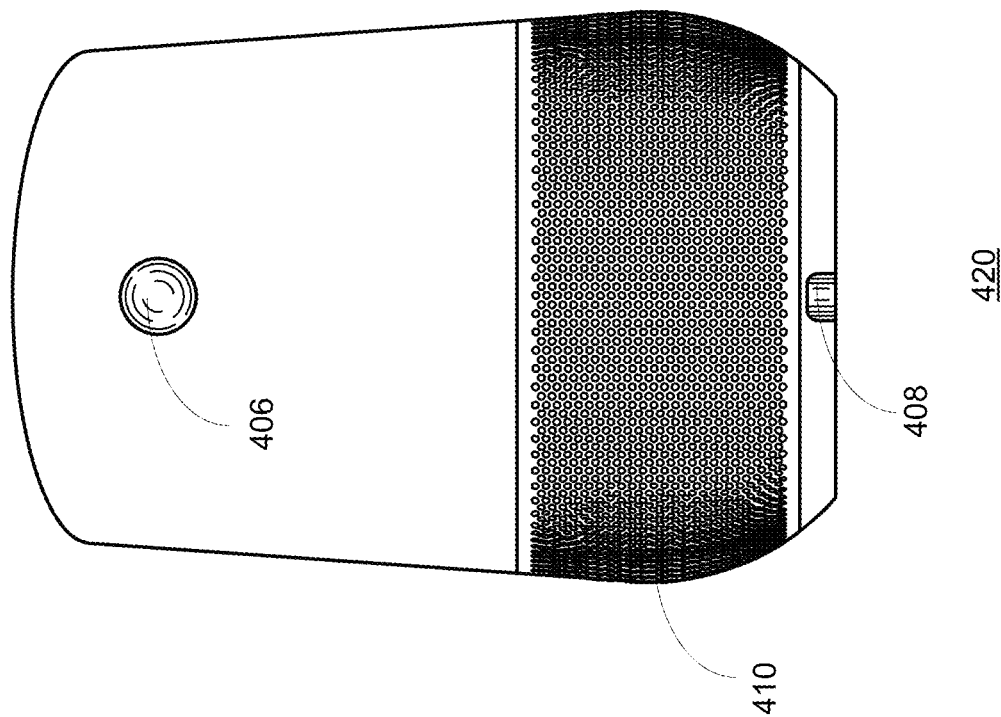


Figure 8B

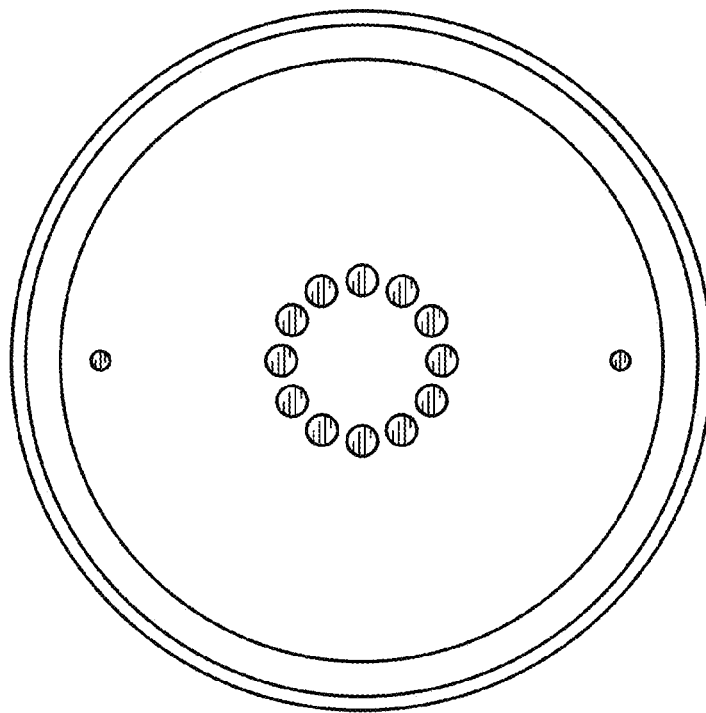


Figure 8C

Motion Sketches

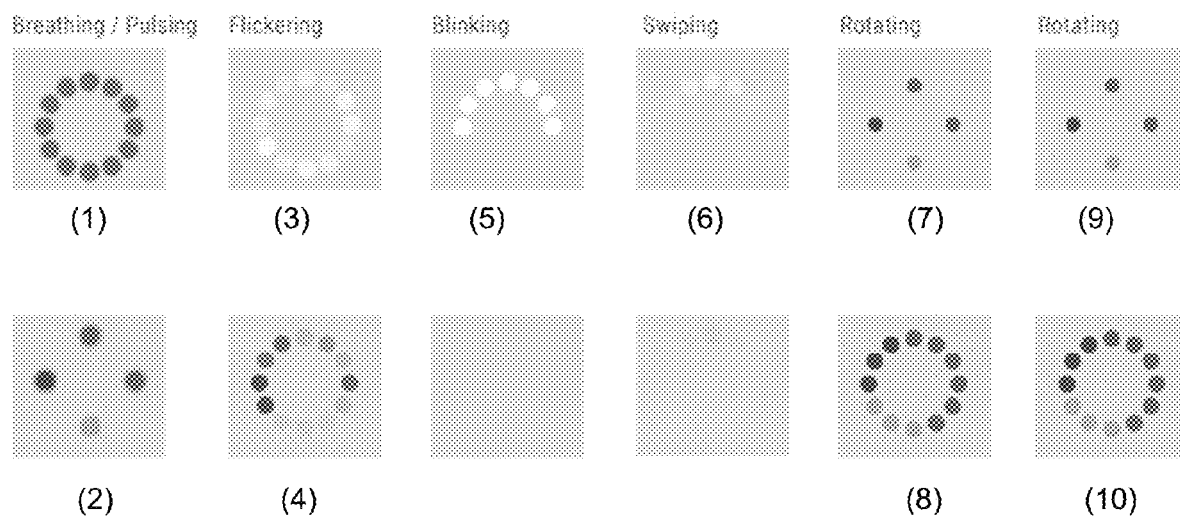


Figure 8D

1

**DEVICE LEADERSHIP NEGOTIATION
AMONG VOICE INTERFACE DEVICES****RELATED APPLICATIONS**

This application is a continuation of U.S. application Ser. No. 16/159,339, filed Oct. 12, 2018, titled "Device Leadership Negotiation Among Voice Interface Devices," which is a continuation of U.S. application Ser. No. 15/788,658, filed Oct. 19, 2017, titled "Device Leadership Negotiation Among Voice Interface Devices," now U.S. Pat. No. 10,102,857, issued Oct. 16, 2018, which is a continuation of U.S. application Ser. No. 15/284,483, filed Oct. 3, 2016, titled "Device Leadership Negotiation Among Voice Interface Devices," now U.S. Pat. No. 9,812,128, issued Nov. 7, 2017, which is a continuation-in-part of U.S. patent application Ser. No. 15/088,477, filed Apr. 1, 2016, titled "Hotword Detection on Multiple Devices," now U.S. Pat. No. 9,514,752, issued Dec. 6, 2016, which is a continuation of U.S. patent application Ser. No. 14/675,932, filed Apr. 1, 2015, titled "Hotword Detection on Multiple Devices," now U.S. Pat. No. 9,318,107, issued Apr. 19, 2016, which claims the benefit of U.S. Provisional Application No. 62/061,830, titled "Hotword Detection on Multiple Devices," filed Oct. 9, 2014, all of which are incorporated by reference herein in their entirety.

This application is related to the following applications, which are incorporated by reference herein in their entirety:

U.S. Provisional Patent Application No. 62/334,434, entitled "Implementations for Voice Assistant on Devices," filed May 10, 2016;

U.S. Provisional Patent Application No. 62/336,566, entitled "LED Design Language for Visual Affordance of Voice User Interfaces," filed May 13, 2016;

U.S. patent application Ser. No. 15/284,476, entitled "Multi-User Personalization at a Voice Interface Device," filed Oct. 3, 2016; and

U.S. patent application Ser. No. 15/284,493, entitled "Noise Mitigation For A Voice Interface Device," filed Oct. 3, 2016.

TECHNICAL FIELD

The disclosed implementations relates generally to voice interfaces and related devices, including but not limited to methods and systems for coordination amongst multiple voice interface devices.

BACKGROUND

Electronic devices integrated with microphones have been widely used to collect voice inputs from users and implement voice-activated functions according to the voice inputs. For example, many state-of-the-art mobile devices include a voice assistant feature (e.g., Siri, Google Assistant) that is configured to use voice inputs to initiate a phone call, conduct a restaurant search, start routing on a map, create calendar events, add a post to a social network, recognize a song, and complete many other tasks.

A location (e.g., a room or space within a home) may include multiple devices that include voice assistant systems and one or more users of such devices. As the devices are more than ever configured to be capable of accessing and presenting user-specific information, it is desirable for a voice assistant device to be able to handle multiple users gracefully and be able to provide user-specific information to the appropriate user. It is also desirable for there to be a

2

leader amongst the voice assistant devices that is responsible for responding to the user's voice inputs, in order to reduce user confusion. Further, it is desirable for a voice assistant device to include noise mitigation measures in case the voice assistant devices are located at a noisy location.

SUMMARY

Accordingly, there is a need for an electronic device with a voice assistant system and/or a voice assistant server system that incorporates methods and systems for handling multiple users, for multi-device coordination to determine a leader, and for noise mitigation. The device and/or the server system may be configured to identify a speaker of a voice input based on comparisons to voice models, and to personalize responses to the identified speaker. The device and/or the server system may also be configured to negotiate leadership amongst multiple devices for response to a speaker. The device and/or the server system may further be configured to detect that an environment around the device is sufficiently noisy where an alternative method, to a hotword voice input, of waking up the device is advisable.

In accordance with some implementations, a method includes, at an electronic device with one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors: receiving a first freeform voice input, comparing the first voice input to a set of voice models associated with a plurality of users associated with the electronic device, where the voice models are trained to the electronic device, based on the comparing, determining that the first voice input corresponds to a first user of the plurality of users, and presenting a response in accordance with the determination.

In some implementations, an electronic device includes one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors. The one or more programs include instructions for: receiving a first freeform voice input, comparing the first voice input to a set of voice models associated with a plurality of users associated with the electronic device, where the voice models are trained to the electronic device, based on the comparing, determining that the first voice input corresponds to a first user of the plurality of users, and presenting a response in accordance with the determination.

In some implementations, a non-transitory computer readable storage medium stores one or more programs. The one or more programs include instructions, which, when executed by an electronic device with one or more microphones, a speaker, and one or more processors, cause the electronic device to perform operations including: receiving a first freeform voice input, comparing the first voice input to a set of voice models associated with a plurality of users associated with the electronic device, where the voice models are trained to the electronic device, based on the comparing, determining that the first voice input corresponds to a first user of the plurality of users, and presenting a response in accordance with the determination.

In accordance with some implementations, a method includes, at a first electronic device of a plurality of electronic devices, each electronic device of the plurality of electronic devices comprising one or more microphones, a speaker, one or more processors, and memory storing one or more programs for execution by the one or more processors: detecting a voice input, determining a quality score for the detected voice input, communicating the quality score to the

3

other devices of the plurality of electronic devices, receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices, in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, where the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input, and in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

In accordance with some implementations, a first electronic device of a plurality of electronic devices includes, and each of the plurality of electronic devices includes, one or more microphones, a speaker, one or more processors, and memory storing one or more programs to be executed by the one or more processors. The one or more programs include instructions for: detecting a voice input; determining a quality score for the detected voice input; communicating the quality score to the other devices of the plurality of electronic devices; receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices; in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, where the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

In accordance with some implementations, a non-transitory computer readable storage medium stores one or more programs. The one or more programs include instructions, which, when executed by a first electronic device of a plurality of electronic devices, each of the plurality of electronic device comprising one or more microphones, a speaker, and one or more processors, cause the electronic device to perform operations including: detecting a voice input; determining a quality score for the detected voice input; communicating the quality score to the other devices of the plurality of electronic devices; receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices; in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, where the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting a response to the detected voice input.

In some implementations, a method includes, at an electronic device with one or more microphones, a speaker, one or more processors, and memory storing one or more

4

programs for execution by the one or more processors, the electronic device configured to be awakened by any of a plurality of affordances including a voice-based affordance: determining a noise profile of an environment around the electronic device; determining whether the noise profile interferes with the voice-based affordance; and in accordance with a determination that the noise profile interferes with the voice-based affordance, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device.

In some implementations, an electronic device includes one or more microphones, a speaker, one or more processors, and memory storing one or more programs to be executed by the one or more processors. The electronic device is configured to be awakened by any of a plurality of affordances including a voice-based affordance. The one or more programs include instructions for: determining a noise profile of an environment around the electronic device, determining whether the noise profile interferes with the voice-based affordance, and in accordance with a determination that the noise profile interferes with the voice-based affordance, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device.

A non-transitory computer readable storage medium stores one or more programs. The one or more programs include instructions, which, when executed by an electronic device with one or more microphones, a speaker, and one or more processors, the electronic device configured to be awakened by any of a plurality of affordances including a voice-based affordance, cause the electronic device to perform operations including: determining a noise profile of an environment around the electronic device, determining whether the noise profile interferes with the voice-based affordance, and in accordance with a determination that the noise profile interferes with the voice-based affordance, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device.

BRIEF DESCRIPTION OF THE DRAWINGS

The patent or application file contains at least one drawing executed in color. Copies of this patent or patent application publication with color drawing(s) will be provided by the Office upon request and payment of the necessary fee.

For a better understanding of the various described implementations, reference should be made to the Description of Implementations below, in conjunction with the following drawings in which like reference numerals refer to corresponding parts throughout the figures.

FIG. 1 is an example operating environment in accordance with some implementations.

FIGS. 2A-2B are block diagrams illustrating an example electronic device that is applied as a voice interface to collect user voice commands in an operating environment in accordance with some implementations.

FIG. 3 is a block diagram illustrating an example server in the server system of an operating environment in accordance with some implementations. An example server is one of a voice assistance server.

FIGS. 4A-4B are diagrams illustrating an example device leadership negotiation, in accordance with some implementations.

FIG. 5 is a flow diagram illustrating a method of responding to a voice input of a user in accordance with some implementations.

5

FIG. 6 is a flow diagram illustrating a method of negotiating device leadership among multiple voice interface devices in accordance with some implementations.

FIG. 7 is a flow diagram illustrating a method of mitigating noise around a voice interface device in accordance with some implementations.

FIGS. 8A and 8B are a front view and a rear view of a voice-activated electronic device in accordance with some implementations.

FIG. 8C is a top view of a voice-activated electronic device in accordance with some implementations, and FIG. 8D shows six visual patterns displayed by an array of full color LEDs for indicating voice processing states in accordance with some implementations.

Like reference numerals refer to corresponding parts throughout the several views of the drawings.

DESCRIPTION OF IMPLEMENTATIONS

While the digital revolution has provided many benefits ranging from openly sharing information to a sense of global community, emerging new technology often induces confusion, skepticism and fear among consumers, preventing consumers from benefitting from the technology. Electronic devices are conveniently used as voice interfaces to receive voice inputs from users and initiate voice-activated functions, and thereby offer eyes-free and hands-free solutions to approach both existing and emerging technology. Specifically, the voice inputs received at the electronic device can carry instructions and information even if a user's line of sight is obscured and his hands are full. To enable a hands-free and eyes-free experience, the voice-activated electronic device listens to the ambient (i.e., constantly processes audio signals collected from the ambient) constantly or only when triggered. On the other hand, user identities are linked with a user's voice and a language used by the user. To protect the user identities, these voice-activated electronic devices are normally used in non-public places that are protected, controlled and intimate spaces (e.g., home and car).

In accordance with some implementations, a voice-activated electronic device identifies a speaker of a voice input as one of one or more associated or registered users. Responses to the speaker output by the electronic device are personalized to the identified speaker. The speaker is identified by comparing the speaker's voice input to a set of trained voice models or voice fingerprints. In some implementations, words and phrases used by a speaker to train the voice models are not necessarily predefined and do not need to match words later spoken by the speaker in order for the electronic device to identify the speaker.

Further, in accordance with some implementations, multiple voice-activated electronic devices negotiate a leader amongst themselves to respond to voice inputs from a user. The negotiation may be based on which device detected the voice input the best, or which device is the closest to the user. Additionally, if the voice input is particularly relevant to one of the devices, that device may be selected to respond even if it would otherwise not be selected for response; the relevance of the command included in the voice input to a particular device (e.g., "stop music" should refer to a device playing music) and the state of the device receiving the voice input (e.g., "screen on" vs. "screen off") are considered in determining which device will be the leader.

Further, in accordance with some implementations, a voice-activated device determines whether the surrounding noise is sufficiently interfering with detection of spoken

6

hotwords for waking up the device (in some implementations, a "hotword" is a user defined or predefined word or phrase used to "wake-up" or trigger a voice-activated device to attend/respond to a spoken command that is issued subsequent to the hotword) or other spoken voice inputs. If the noise is sufficiently interfering, the device indicates such and gives the user a hint to use an alternative way of waking up the device or other noise mitigation measures.

Voice Assistant Operating Environment

FIG. 1 is an example operating environment in accordance with some implementations. Operating environment 100 includes one or more voice-activated electronic devices 190 (e.g., electronic devices 190-1 thru 190-N). The one or more voice-activated electronic devices 190 may be located in one or more locations (e.g., all in a room or space of a structure, spread out throughout multiple spaces within a structure or throughout multiple structures (e.g., one in a house and one in the user's car)). The environment 100 optionally includes one or more devices (e.g., media devices, smart home devices) (not shown) with which the voice-activated electronic devices 190 can communicate.

The electronic devices 190 are communicatively coupled, through communication networks 110, to a voice assistance server 112 of a voice assistant service. One or more of the electronic devices 190 are communicatively coupled to a local network 104, which is communicatively coupled to the communication networks 110 (e.g., electronic devices 190-1 and 190-2). In some implementations, the local network 104 is a local area network implemented at a network interface (e.g., a router). The electronic devices 190 that are communicatively coupled to the local network 104 may also communicate with each other through the local network 104.

Optionally, one or more of the electronic devices 190 are communicatively coupled to the communication networks 110 and are not on the local network 104 (e.g., electronic device 190-N). For example, these electronic devices 190 are not on the Wi-Fi network corresponding to the local network 104 but are connected to the communication networks 110 through a cellular connection. In some implementations, communication between electronic devices 190 that are on the local network 104 and electronic devices 190 that are not on the local network 104 are done through the voice assistance server 112. The electronic devices 190 are registered in a device registry 118 of the voice assistant service and thus known to the voice assistance server 112.

The environment 100 also includes one or more content hosts 114. A content host 114 may be a remote content source from which content is streamed or otherwise obtained in accordance with a user voice request. A content host 114 may be an information source from which the voice assistance server 112 retrieves information in accordance with a user voice request.

In some implementations, an electronic device 190 may be associated with multiple users having respective user accounts in the user domain. Any of these users, as well as users not associated with the device, may make voice inputs to the electronic device 190. The electronic device 190 receives these voice inputs from these users 102-1 thru 102-M (including associated and non-associated users), and the electronic device 190 and/or the voice assistance server 112 proceeds to identify, for a voice input, the user making the voice input. With the user identification, a response to that voice input may be personalized to the identified user.

In some implementations, the environment 100 includes multiple electronic devices 190 (e.g., devices 190-1 thru

190-N). The devices 190 are located throughout the environment 100 (e.g., all within a room or space in a structure, spread throughout the structure, some within the structure and some without). When a user 102 makes a voice input, each of the devices 190 either receives the input or does not receive the input (e.g., if the device was too far away from the user). The devices that receive the input receive the input at varying degrees of quality; the quality of the sample of the voice input at a device is based on multiple factors, including but not limited to distance of the user from the device and the noise around the device. The multiple devices 190 negotiate a leader amongst themselves to respond to the user and to receive further voice input from the user 102 based on the quality of the samples of the voice inputs.

In some implementations, an electronic device 190 determines a level of noise around the device and determines whether the determined noise level is sufficiently high to interfere with recognition of the hotword in voice inputs, and thus interfere with awakening of the device by voice, or with recognition of voice requests. If the noise level is determined to be sufficiently high to be interfering, the electronic device 190 indicates to the user that the noise level is interfering and gives the user a hint that the user should use another way to wake up the electronic device 190 (e.g., activate a button). The indication of the interfering noise level and the hint to use another way to wake up the device may be done in the same presentation (e.g., illuminating the wake-up button).

In some implementations, one or more media devices are disposed in the operating environment 100 to provide to one or more occupants media content, news and/or other information. In some implementations, the content provided by the media devices is stored at a local content source, streamed from a remote content source (e.g., content host(s) 114), or generated locally (e.g., through a local text to voice processor that reads a customized news briefing, emails, texts, a local weather report, etc. to one or more occupants of the operating environment 100). In some implementations, the media devices include media output devices that directly output the media content to an audience (e.g., one or more users), and cast devices that are networked to stream media content to the media output devices. Examples of the media output devices include, but are not limited to television (TV) display devices and music players. Examples of the cast devices include, but are not limited to, set-top boxes (STBs), DVD players, TV boxes, and media streaming devices, such as Google's Chromecast™ media streaming device.

In some implementations, the media devices include one or more voice-activated electronic devices 190 that receive, process and respond to voice commands of occupants. In some implementations, the voice-activated electronic devices 190 respond to voice commands by: generating and providing a spoken response to a voice command (e.g., speaking the current time in response to the question, "what time is it?"); streaming media content requested by a user (e.g., "play a Beach Boys song"); reading a news story or a daily news briefing prepared for the user; playing a media item stored on the personal assistant device or on the local network; changing a state or operating one or more other connected devices within the operating environment 100 (e.g., turning lights, appliances or media devices on/off, locking/unlocking a lock, opening windows, etc.); or issuing a corresponding request to a server via a network 110.

In some implementations, the one or more electronic devices 190 are disposed in the operating environment 100 to collect audio inputs for initiating various functions, including media play functions of the media devices. In

some implementations, these voice-activated electronic devices 190 (e.g., devices 190-1 thru 190-N) are disposed in proximity to a media device, for example, in the same room with the cast devices and the media output devices. Alternatively, in some implementations, a voice-activated electronic device 190 is disposed in a room having one or more smart home devices but not any media device. Alternatively, in some implementations, a voice-activated electronic device 190 is disposed in a location having no networked electronic device. Further, in some implementations, a room or space in the structure may have multiple electronic devices 190.

In some implementations, the electronic device 190 includes at least one or more microphones, a speaker, a processor and memory storing at least one program for execution by the processor. The speaker is configured to allow the electronic device 190 to deliver voice messages and other audio (e.g., audible tones) to a location where the electronic device 190 is located in the operating environment 100, thereby broadcasting music, reporting a state of audio input processing, having a conversation with or giving instructions to a user of the electronic device 190. As an alternative to the voice messages, visual signals could also be used to provide feedback to the user of the electronic device 190 concerning the state of audio input processing. When the electronic device 190 is a mobile device (e.g., a mobile phone or a tablet computer), its display screen is configured to display a notification concerning the state of audio input processing.

In some implementations, the electronic device 190 is a voice interface device that is network-connected to provide voice recognition functions with the aid of a voice assistance server 112 and/or optionally a cloud cast service server (not shown). For example, the electronic device 190 includes a smart speaker that provides music to a user and allows eyes-free and hands-free access to a voice assistant service (e.g., Google Assistant). Optionally, the electronic device 190 is one of a desktop or laptop computer, a tablet, a mobile phone that includes a microphone, a cast device that includes a microphone and optionally a speaker, an audio system (e.g., a stereo system, a speaker system, a portable speaker) that includes a microphone and a speaker, a television that includes a microphone and a speaker, and a user interface system in an automobile that includes a microphone and a speaker and optionally a display. Optionally, the electronic device 190 is a simple and low cost voice interface device. Generally, the electronic device 190 may be any device that is capable of network connection and that includes a microphone, a speaker, and programs, modules, and data for interacting with voice assistant service. Given simplicity and low cost of the electronic device 190, the electronic device 190 includes an array of light emitting diodes (LEDs) rather than a full display screen, and displays a visual pattern on the LEDs to indicate the state of audio input processing. In some implementations, the LEDs are full color LEDs, and the colors of the LEDs may be employed as a part of the visual pattern to be displayed on the LEDs. Multiple examples of using LEDs to display visual patterns in order to convey information or device status are described in U.S. Provisional Patent Application No. 62/336,566, entitled "LED Design Language for Visual Affordance of Voice User Interfaces," filed May 13, 2016, which is incorporated by reference in its entirety. In some implementations, visual patterns indicating the state of voice processing operations are displayed using characteristic images shown on conventional displays associated with electronic devices that are performing the voice processing operations.

In some implementations, LEDs or other visual displays are used to convey a collective voice processing state of multiple participating electronic devices. For example, in an operating environment where there are multiple voice processing or voice interface devices (e.g., multiple electronic devices 400 as shown in FIG. 4A of the '566 application, duplicated as FIG. 8A below; multiple electronic devices 190), groups of color LEDs (e.g., LEDs 404 as shown in FIG. 4A of the '566 application) associated with respective electronic devices can be used to convey which of the electronic devices is listening to a user, and which of the listening devices is the leader (where the "leader" device generally takes the lead in responding to a spoken request issued by the user).

More generally, the '566 application describes (e.g., see paras. [0087]-[0100]) a "LED Design Language" for indicating visually using a collection of LEDs a variety of voice processing states of an electronic device, such as a "Hot word detection state and listening state," a "Thinking mode or working mode," and a "Responding mode or speaking mode." In some implementations, unique states of voice processing operations described herein are represented using a group of LEDs in accordance with one or more aspects of the "LED Design Language" of the '566 application. These visual indicators can also be combined with one or more audible indicators generated by electronic devices that are performing voice processing operations. The resulting audio and/or visual indicators will enable users in a voice-interactive environment to understand the state of various voice processing electronic devices in the environment and to effectively interact with those devices in a natural, intuitive manner.

When voice inputs from the electronic device 190 are used to control the media output devices via the cast devices, the electronic device 190 effectively enables a new level of control of cast-enabled media devices. In a specific example, the electronic device 190 includes a casual enjoyment speaker with far-field voice access and functions as a voice interface device for the voice assistant service. The electronic device 190 could be disposed in any area in the operating environment 100. When multiple electronic devices 190 are distributed in multiple rooms, they become cast audio receivers that are synchronized to provide voice inputs from all these rooms.

Specifically, in some implementations, the electronic device 190 includes a Wi-Fi speaker with a microphone that is connected to a voice-activated voice assistant service (e.g., Google Assistant). A user can issue a media play request via the microphone of electronic device 190, and ask the voice assistant service to play media content on the electronic device 190 itself or on another connected media output device. For example, the user can issue a media play request by saying to the Wi-Fi speaker "OK Google, play cat videos on my Living room TV." The voice assistant service then fulfills the media play request by playing the requested media content on the requested device using a default or designated media application.

In some implementations, a user can issue a voice request, via the microphone of the electronic device 190, concerning media content that has already been played or is being played on a display device (e.g., the user can ask for information about the media content, buy the media content through an online store, or compose and issue a social post about the media content).

In some implementations, a user may want to take a current media session with them as they move through the house and can request such a service from one or more of the

electronic devices 190. This requires the voice assistant service to transfer the current media session from a first cast device to a second cast device that is not directly connected to the first cast device or has no knowledge of the existence of the first cast device. Subsequent to the media content transfer, a second output device coupled to the second cast device continues to play the media content previously a first output device coupled to the first cast device from the exact point within a music track or a video clip where play of the media content was forgone on the first output device. In some implementations, the electronic device 190 that receives the request to transfer the media session can satisfy the request. In some implementations, the electronic device 190 that receives the request to transfer the media session relays the request to another device or system (e.g., a hub device, voice assistance server 112) for handling.

Further, in some implementations, a user may issue, via the microphone of electronic device 190, a request for information or for performance of an action or operation. The information requested may be personal (e.g., the user's emails, the user's calendar events, the user's flight information, etc.), non-personal (e.g., sports scores, news stories, etc.) or somewhere in between (e.g., scores for teams or sports preferred by the user, news stories from the user's preferred sources, etc.). The requested information or action/operation may involve access to personal information (e.g., purchasing a digital media item with payment information provided by the user, purchasing a physical good). The electronic device 190 responds to the request with voice message responses to the user, where the response may include, for example, requests for additional information to fulfill the request, confirmation that the request has been fulfilled, notice that the request cannot be fulfilled, and so forth.

In some implementations, in addition to the voice-activated electronic devices 190 and the media devices (e.g., the output devices and the cast devices), the operating environment 100 may also include one or more smart home devices (not shown). The integrated smart home devices include intelligent, multi-sensing, network-connected devices that integrate seamlessly with each other in a smart home network and/or with a central server or a cloud-computing system to provide a variety of useful smart home functions. In some implementations, a smart home device is disposed at the same location of the operating environment 100 as a cast device and/or an output device, and therefore, is located in proximity to or with a known distance with respect to the cast device and the output device.

The smart home devices in the operating environment 100 may include, but are not limited to, one or more intelligent, multi-sensing, network-connected thermostats, one or more intelligent, network-connected, multi-sensing hazard detectors, one or more intelligent, multi-sensing, network-connected entryway interface devices and (hereinafter referred to as "smart doorbells" and "smart door locks"), one or more intelligent, multi-sensing, network-connected alarm systems, one or more intelligent, multi-sensing, network-connected camera systems, and one or more intelligent, multi-sensing, network-connected wall switches, and one or more intelligent, multi-sensing, network-connected power sockets. In some implementations, the smart home devices in the operating environment 100 of FIG. 1 includes a plurality of intelligent, multi-sensing, network-connected appliances (hereinafter referred to as "smart appliances"), such as refrigerators, stoves, ovens, televisions, washers, dryers, lights, stereos, intercom systems, garage-door openers, floor fans, ceiling fans, wall air conditioners, pool heaters, irri-

11

gation systems, security systems, space heaters, window AC units, motorized duct vents, and so forth. In some implementations, any one of these smart home device types can be outfitted with microphones and one or more voice processing capabilities as described herein so as to in whole or in part respond to voice requests from an occupant or user.

In some implementations, each of the cast devices and the voice-activated electronic devices **190** is capable of data communications and information sharing with other cast devices, voice-activated electronic devices **190**, smart home devices, a central server or cloud-computing system, and/or other devices (e.g., the client device) that are network-connected. Data communications may be carried out using any of a variety of custom or standard wireless protocols (e.g., IEEE 802.15.4, Wi-Fi, ZigBee, 6LoWPAN, Thread, Z-Wave, Bluetooth Smart, ISA100.11a, WirelessHART, MiWi, etc.) and/or any of a variety of custom or standard wired protocols (e.g., Ethernet, HomePlug, etc.), or any other suitable communication protocol, including communication protocols not yet developed as of the filing date of this document.

Through the communication networks (e.g., the Internet) **110**, the cast devices, the electronic devices **190** and the smart home devices may communicate with a server system (also called a central server system and/or a cloud-computing system herein). Optionally, the server system may be associated with a manufacturer, support entity, or service provider associated with the cast devices and the media content displayed to the user. Accordingly, the server system includes the voice assistance server **112** that processes audio inputs collected by voice-activated electronic devices **190**, one or more content hosts **114** that provide the displayed media content, a cloud cast service server creating a virtual user domain based on distributed device terminals, and the device registry **118** that keeps a record of the distributed device terminals in the virtual user environment. Examples of the distributed device terminals include, but are not limited to the cast devices, the media output devices, the electronic devices **190** and the smart home devices. In some implementations, these distributed device terminals are linked to a user account (e.g., a Google user account) in the virtual user domain. It should be appreciated that processing of audio inputs collected by electronic devices **190** can be performed locally at an electronic device **190**, at a voice assistance server **112**, at another smart home device (e.g., a hub device) or at some combination of all or subset of the above.

It will be appreciated that in some implementations the electronic device(s) **190** also function in an environment without smart home devices. For example, an electronic device **190** can, even in the absence of smart home devices, respond to user requests for information or performance of an action, and/or to initiate or control various media play functions. An electronic device **190** can also function in a wide range of environments, including, without limitation, a vehicle, a ship, a business, or a manufacturing environment.

In some implementations, an electronic device **190** is “awakened” (e.g., to activate an interface for the voice assistant service on the electronic device **190**, to put the electronic device **190** into a state where the electronic device **190** is ready to receive voice requests to the voice assistant service) by a voice input that includes a hotword (also called a “wake word”). In some implementations, the electronic device **190** requires awakening if the electronic device **190** has been idle with respect to receipt of voice inputs for at least a predefined amount of time (e.g., 5 minutes); the predefined amount of time corresponds to an amount of idle

12

time allowed before a voice interface session or conversation times out. The hotword may be a word or phrase, and may be a predefined default and/or may be customized by a user (e.g., a user may set a nickname for a particular electronic device **190** as the device’s hotword). In some implementations, there may be multiple hotwords that can awaken an electronic device **190**. A user may speak the hotword, wait for an acknowledgement response from the electronic device **190** (e.g., the electronic device **190** outputs a greeting), and then make a first voice request. Alternatively, the user may combine the hotword and the first voice request in one voice input (e.g., the voice input includes the hotword followed by the voice request).

In some implementations, the voice inputs spoken by the user to an electronic device **190** may generally be freeform or natural language speech. That is, the voice input need not be strictly limited to a predefined set of words and phrases within a predefined syntax, with possibly certain exceptions (e.g., user needs to speak the hotword first to awaken the device).

In some implementations, an electronic device **190** includes one or more additional ways or affordances to wake up the device besides speaking the hotword to the electronic device **190**. The additional ways or affordances may include, for example, activating a predefined button or touch sensor (e.g., a touch sense array) on the electronic device **190**.

In some implementations, a voice-activated electronic device **190** interacts with a cast device, a client device or a server system of an operating environment **100** in accordance with some implementations. The voice-activated electronic device **190** is configured to receive audio inputs from an environment in proximity to the voice-activated electronic device **190**. Optionally, the electronic device **190** stores the audio inputs and at least partially processes the audio inputs locally. Optionally, the electronic device **190** transmits the received audio inputs or the partially processed audio inputs to a voice assistance server **112** via the communication networks **110** for further processing. The cast device is configured to obtain media content or Internet content from one or more content hosts **114** for display on an output device coupled to the cast device. In some implementations, the cast device and the voice-activated electronic device **190** are linked to each other in a user domain, and more specifically, associated with each other via a user account in the user domain. Information of the cast device and information of the electronic device **190** are stored in the device registry **118** in association with the user account. In some implementations, there is a device registry for cast devices and a registry for voice-activated electronic devices **190**. In some implementations, a cloud cast service server manages the cast devices registry and the voice assistance server **112** manages the voice-activated electronic devices registry. The cast devices registry may reference devices in the voice-activated electronic devices registry that are associated in the user domain, and vice versa.

In some implementations, one or more of the electronic devices **190** (and one or more cast devices) are commissioned to the voice assistant service via a client device (not shown). In some implementations, the voice-activated electronic device **190** does not include any display screen, and relies on the client device to provide a user interface during a commissioning process, and similarly for a cast device as well. Specifically, the client device is installed with an application that enables a user interface to facilitate commissioning of a new voice-activated electronic device **190** disposed in proximity to the client device. A user may send a request on the user interface of the client device to initiate

13

a commissioning process for the new electronic device **190** that needs to be commissioned. After receiving the commissioning request, the client device establishes a short range communication link with the new electronic device **190** that needs to be commissioned. Optionally, the short range communication link is established based near field communication (NFC), Bluetooth, Bluetooth Low Energy (BLE) and the like. The client device then conveys wireless configuration data associated with a wireless local area network (WLAN) (e.g., local network **104**) to the new or electronic device **190**. The wireless configuration data includes at least a WLAN security code (i.e., service set identifier (SSID) password), and optionally includes a SSID, an Internet protocol (IP) address, proxy configuration and gateway configuration. After receiving the wireless configuration data via the short range communication link, the new electronic device **190** decodes and recovers the wireless configuration data, and joins the WLAN based on the wireless configuration data.

In some implementations, additional user domain information is entered on the user interface displayed on the client device, and used to link the new electronic device **190** to an account in a user domain. Optionally, the additional user domain information is conveyed to the new electronic device **190** in conjunction with the wireless communication data via the short range communication link. Optionally, the additional user domain information is conveyed to the new electronic device **190** via the WLAN after the new device has joined the WLAN.

Once the electronic device **190** has been commissioned into the user domain, other devices and their associated activities may be controlled via multiple control paths. In accordance with one control path, an application installed on the client device is used to control the other device and its associated activities (e.g., media play activities). Alternatively, in accordance with another control path, the electronic device **190** is used to enable eyes-free and hands-free control of the other device and its associated activities.

Voice Activated Media Play on a Media Output Device

In some implementations, after the cast device and the voice-activated electronic device **190** are both commissioned and linked to a common user domain, the voice-activated electronic device **190** can be used as a voice user interface to enable eyes-free and hands-free control of media content streaming to the cast device involving no remote control, client device or other second screen device. For example, the user may give voice commands such as "Play Lady Gaga on Living Room speakers." A Lady Gaga music track or video clip is streamed to a cast device associated with the "Living Room speakers." The client device is not involved, nor is any cast device application or media play application loaded on the client device.

In some implementations, a proxy service (e.g., voice assistant service, a cloud cast service) communicatively links the voice-activated electronic device **190** to the cast device and makes casting to the cast device possible without involving any applications on the client device. Specifically, a voice message is captured and recorded by an electronic device **190**, and the voice message is configured to request media play on a media output device. Optionally, the electronic device **190** partially processes the voice message locally. Optionally, the electronic device **190** transmits the voice message or the partially processed voice message to a voice assistance server **112** via the communication networks

14

110 for further processing. The voice assistance server **112** or a cloud cast service server determines that the voice message includes a first media play request, and that the first media play request includes a user voice command to play media content on a media output device and a user voice designation of the media output device. The user voice command further includes at least information of a first media play application (e.g., YouTube and Netflix) and the media content (e.g., Lady Gaga music) that needs to be played.

In accordance with the voice designation of the media output device, the voice assistance server **112** or the cloud cast service server identifies in a device registry **118** a cast device associated in the user domain with the electronic device **190** and coupled to the media output device. The cast device is configured to execute one or more media play applications for controlling the media output device to play media content received from one or more media content hosts **114**. Then, the cloud cast service server sends to the cast device a second media play request including the information of the first media play application and the media content that needs to be played. Upon receiving the information sent by the cloud cast service server, the cast device executes the first media play application and controls the media output device to play the requested media content.

In some implementations, the user voice designation of the media output device includes a description of the destination media output device. The voice assistance server **112** or the cloud cast service server identifies in the device registry **118** the destination media output device among a plurality of media output devices according to the description of the destination media output device. In some implementations, the description of the destination media output device includes at least a brand ("Samsung TV") or a location of the media output device ("my Living Room TV").

Voice Activated Media Transfer Among Media Output Devices

In some implementations, the operating environment **100** includes a first cast device (not shown) and a first output device (not shown) coupled to the first cast device. The operating environment **100** also includes a second cast device (not shown) and a second output device (not shown) coupled to the second cast device. The cast devices are optionally located in the same location (e.g., the living room) or two distinct locations (e.g., two rooms) in the operating environment **100**. Each of the cast devices is configured to obtain media content or Internet content from content hosts **114** for display on the output device coupled to the respective cast device. Both the first and second cast devices are communicatively coupled to the voice assistance server **112**, optionally a cloud cast service server, and the content hosts **114**.

The operating environment **100** further includes one or more voice-activated electronic devices **190** that are communicatively coupled to the voice assistance server **112** and optionally the cloud cast service server. In some implementations, the voice-activated electronic devices **190** are disposed independently of the cast devices and the output devices. For example, an electronic device **190** is disposed in a room where no cast device or output device is located. In some implementations, a first electronic device **190** is disposed in proximity to the first cast device and the first output device, e.g., the first electronic device **190**, the first cast device and the first output device are located in the same

15

room. Optionally, a second electronic device **190** is disposed independently of or in proximity to the second cast device and the second output device.

When media content is being played on the first output device, a user may send a voice command to any of the electronic devices **190** to request play of the media content to be transferred to the second output device. The voice command includes a media play transfer request. In one situation, the user can issue the voice command to the electronic device **190** disposed in proximity to the first cast device before the user moves to a destination location. Alternatively, in another situation, the user can issue the voice command to the electronic device **190** disposed in proximity to the second device after the user reaches the destination location.

The voice command is transmitted to the voice assistance server **112**. The voice assistance server **112** sends a media display information request to the first cast device to request instant media play information of the media content that is currently being played on the first output device coupled to the first cast device. The first cast device then returns to the voice assistance server **112** the requested instant play information including at least information of a first media play application (e.g., YouTube), the media content that is currently being played (e.g., “Lady Gaga—National Anthem—Super Bowl 2016”), and a temporal position related to playing of the media content. The second cast device then receives a media display request including the instant play information from the voice assistance server **112**, and in accordance with the instant play information, executes the first media play application that controls the second output device to play the media content from the temporal location.

For example, when a music playlist is played on the first output device, the user says “Play on my living room speakers.” The first output device stops playing the currently played song, and the stopped song resumes on the living room speakers. When the song is completed, the living room speakers continue to play the next song on the music playlist previously played on the first output device. As such, when the user is moving around in the operating environment **100**, the play of the media content would seamlessly follow the user while only involving limited user intervention (i.e., giving the voice command).

Individual Devices Involved in the Operating Environment

FIGS. 2A-2B are block diagrams illustrating an example electronic device **190** that is applied as a voice interface to collect user voice commands in an operating environment (e.g., operating environment **100**) in accordance with some implementations. The electronic device **190**, typically, includes one or more processing units (CPUs) **502**, one or more network interfaces **504**, memory **506**, and one or more communication buses **508** for interconnecting these components (sometimes called a chipset). The electronic device **190** includes one or more input devices **510** that facilitate user input, such as a button **512**, a touch sense array **514**, and one or more microphones **516**. The electronic device **190** also includes one or more output devices **518**, including one or more speakers **520**, optionally an array of LEDs **522**, and optionally a display **524**. In some implementations, the array of LEDs **522** is an array of full color LEDs. In some implementations, an electronic device **190**, depending on the type of device, has either the array of LEDs **522**, or the display **524**, or both. In some implementations, the electronic device **190** also includes a location detection device

16

526 (e.g., a GPS module) and one or more sensors **528** (e.g., accelerometer, gyroscope, light sensor, etc.).

Memory **506** includes high-speed random access memory, such as DRAM, SRAM, DDR RAM, or other random access solid state memory devices; and, optionally, includes non-volatile memory, such as one or more magnetic disk storage devices, one or more optical disk storage devices, one or more flash memory devices, or one or more other non-volatile solid state storage devices. Memory **506**, optionally, includes one or more storage devices remotely located from one or more processing units **502**. Memory **506**, or alternatively the non-volatile memory within memory **506**, includes a non-transitory computer readable storage medium. In some implementations, memory **506**, or the non-transitory computer readable storage medium of memory **506**, stores the following programs, modules, and data structures, or a subset or superset thereof:

Operating system **532** including procedures for handling various basic system services and for performing hardware dependent tasks;

Network communication module **534** for connecting the electronic device **190** to other devices (e.g., the server system **140**, one or more cast devices, one or more client devices, one or more smart home devices, and other electronic device(s) **190**) via one or more network interfaces **504** (wired or wireless) and one or more networks **110**, such as the Internet, other wide area networks, local area networks (e.g., local network **104**), metropolitan area networks, and so on;

Input/output control module **536** for receiving inputs via one or more input devices and enabling presentation of information at the electronic device **190** via one or more output devices **518**, including:

Voice processing module **538** for processing audio inputs or voice messages collected in an environment surrounding the electronic device **190**, or preparing the collected audio inputs or voice messages for processing at a voice assistance server **112**;

LED control module **540** for generating visual patterns on the LEDs **522** according to device states of the electronic device **190**; and

Touch sense module **542** for sensing touch events on a top surface (e.g., on touch sensor array **514**) of the electronic device **190**;

Voice activated device data **544** for storing at least data associated with the electronic device **190**, including:

Voice device settings **546** for storing information associated with the electronic device **190** itself, including common device settings (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.), information of one or more user accounts in a user domain, settings regarding restrictions when dealing with a non-registered user, and display specifications associated with one or more visual patterns displayed by the LEDs **522**;

Voice control data **548** for storing audio signals, voice messages, response messages and other data related to voice interface functions of the electronic device **190**;

Voice models data **550** for storing voice models or voice fingerprints corresponding to users in the user domain that are associated with the electronic device **190**;

Device leadership module **552** for determining a leader amongst multiple electronic devices **190** in response to user voice inputs;

17

Voice model training module **554** for training and generating voice models or voice fingerprints **550** that can be used to identify and disambiguate users in the user domain that are associated with the electronic device **190**;

Response module **556** for performing instructions included in voice request responses generated by the voice assistance server **112**, and in some implementations, generating responses to certain voice inputs; and Noise module **558** for determining noise levels around the electronic device **190** and provide indications of interfering noise levels and hints to use alternative ways to wake up the electronic device.

In some implementations, the voice processing module **538** includes the following modules:

User identification module **560** for identifying and disambiguating users who provide voice inputs to the electronic device **190**;

Hotword recognition module **562** for determining whether voice inputs include a hotword for waking up the electronic device **190** and recognizing such in the voice inputs;

Request recognition module **564** for determining a user request included in a voice input; and

Recognition quality module **566** for determining a measure of the quality (e.g., a score) of recognition of hotwords and/or requests in voice inputs.

In some implementations, the noise module **558** includes the following modules:

Recognition analysis module **568** for analyzing how different levels of noise affect the operations of the voice processing module **538** (e.g., in particular the hotword recognition module **562** and/or the request recognition module **564**) with respect to the particular electronic device **190**;

User volume threshold module **568** for determining, for a given distance from the electronic device **190**, highest comfortable volume levels at which users will utter voice inputs to the electronic device, **190**;

Noise determination module **572** for determining a noise level and a noise profile around the electronic device **190**;

Noise profiles module **574** for training and generating noise profiles of noises that interfere with recognition of hotwords and/or requests in voice inputs; and

Noise profiles store **576** for storing noise profiles that have been determined to be interfering.

Each of the above identified elements may be stored in one or more of the previously mentioned memory devices, and corresponds to a set of instructions for performing a function described above. The above identified modules or programs (i.e., sets of instructions) need not be implemented as separate software programs, procedures, modules or data structures, and thus various subsets of these modules may be combined or otherwise re-arranged in various implementations. In some implementations, memory **506**, optionally, stores a subset of the modules and data structures identified above. Furthermore, memory **506**, optionally, stores additional modules and data structures not described above. In some implementations, a subset of the programs, modules, and/or data stored in the memory **506** can be stored on and/or executed by the server system **140**/voice assistance server **112**.

In some implementations, one or more of the modules in memory **506** described above are part of a voice processing library of modules. The voice processing library may be implemented and embedded on a wide variety of devices. An

18

example of a voice processing library is described in U.S. Provisional Patent Application No. 62/334,434, entitled "Implementations for Voice Assistant on Devices," filed May 10, 2016, which is incorporated by reference herein in its entirety.

FIG. 3 is a block diagram illustrating an example server in the server system **140** of an operating environment (e.g., operating environment **100**) in accordance with some implementations. An example server is one of a voice assistance server **112**. The server **140**, typically, includes one or more processing units (CPUs) **702**, one or more network interfaces **704**, memory **706**, and one or more communication buses **708** for interconnecting these components (sometimes called a chipset). The server **140** could include one or more input devices **710** that facilitate user input, such as a keyboard, a mouse, a voice-command input unit or microphone, a touch screen display, a touch-sensitive input pad, a gesture capturing camera, or other input buttons or controls. Furthermore, the server **140** could use a microphone and voice recognition or a camera and gesture recognition to supplement or replace the keyboard. In some implementations, the server **140** includes one or more cameras, scanners, or photo sensor units for capturing images, for example, of graphic series codes printed on the electronic devices. The server **140** could also include one or more output devices **712** that enable presentation of user interfaces and display content, including one or more speakers and/or one or more visual displays.

Memory **706** includes high-speed random access memory, such as DRAM, SRAM, DDR RAM, or other random access solid state memory devices; and, optionally, includes non-volatile memory, such as one or more magnetic disk storage devices, one or more optical disk storage devices, one or more flash memory devices, or one or more other non-volatile solid state storage devices. Memory **706**, optionally, includes one or more storage devices remotely located from one or more processing units **702**. Memory **706**, or alternatively the non-volatile memory within memory **706**, includes a non-transitory computer readable storage medium. In some implementations, memory **706**, or the non-transitory computer readable storage medium of memory **706**, stores the following programs, modules, and data structures, or a subset or superset thereof:

Operating system **716** including procedures for handling various basic system services and for performing hardware dependent tasks;

Network communication module **718** for connecting the server system **140** to other devices (e.g., various servers in the server system **140**, client devices, cast devices, electronic devices **190**, and smart home devices) via one or more network interfaces **704** (wired or wireless) and one or more networks **110**, such as the Internet, other wide area networks, local area networks, metropolitan area networks, and so on;

User interface module **720** for enabling presentation of information (e.g., a graphical user interface for presenting application(s) **826-830**, widgets, websites and web pages thereof, and/or games, audio and/or video content, text, etc.) at a client device;

Command execution module **721** for execution on the server side (e.g., games, social network applications, smart home applications, and/or other web or non-web based applications for controlling a client device, a cast devices, an electronic device **190** and a smart home devices and reviewing data captured by such devices), including one or more of:

19

a cast device application **722** that is executed to provide server-side functionalities for device provisioning, device control, and user account management associated with cast device(s);

one or more media player applications **724** that is executed to provide server-side functionalities for media display and user account management associated with corresponding media sources;

one or more smart home device applications **726** that is executed to provide server-side functionalities for device provisioning, device control, data processing and data review of corresponding smart home devices; and

a voice assistance application **728** that is executed to arrange voice processing of a voice message received from the electronic device **190** or directly process the voice message to extract a user voice command and one or more parameters for the user voice command (e.g., a designation of a cast device or another electronic device **190**); and

Server system data **730** storing at least data associated with automatic control of media display (e.g., in an automatic media output mode and a follow-up mode), including one or more of:

Client device settings **732** for storing information associated with one or more client device, including common device settings (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.), and information for automatic media display control;

Cast device settings **734** for storing information associated with user accounts of the cast device application **722**, including one or more of account access information, information for device settings (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.), and information for automatic media display control;

Media player application settings **736** for storing information associated with user accounts of one or more media player applications **724**, including one or more of account access information, user preferences of media content types, review history data, and information for automatic media display control;

Smart home device settings **738** for storing information associated with user accounts of the smart home applications **726**, including one or more of account access information, information for one or more smart home devices (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.);

Voice assistance data **740** for storing information associated with user accounts of the voice assistance application **728**, including one or more of account access information, information for one or more electronic device **190** (e.g., service tier, device model, storage capacity, processing capabilities, communication capabilities, etc.);

User data **742** for storing information associated with users in the use domain, including users' subscriptions (e.g., music streaming service subscriptions, video streaming service subscriptions, newsletter subscriptions), user devices (e.g., devices registered in the device registry **118** associated with respective users), user accounts (e.g., users' email accounts, calendar accounts, financial accounts), and other user data; and

20

User voice profiles **744** for storing voice profiles of the users in the user domain, including, for example, voice models or voice fingerprints of the users and comfortable volume level thresholds of the users.

Device registration module **750** for managing the device registry **118** coupled to the voice assistance server **112**;

Device leadership moderator module **780** for moderating device leadership determinations between the electronic devices **190** in the operating environment;

Noise module **790** for determining noise levels around the electronic device **190**; and

Voice processing module **7100** for processing audio inputs or voice messages collected in an environment surrounding the electronic device **190**.

In some implementations, the voice assistance server **112** is primarily responsible for processing of voice inputs and for noise mitigation, and thus one or more of the programs, modules, and data structures in memory **506** described above with reference to FIGS. 2A-2B are included in respective modules in memory **706** (e.g., the programs, modules, and data structures included with voice processing module **538** are included in voice processing module **7100**, and the programs, modules, and data structures included with noise module **558** are included in noise module **790**).

The electronic device **190** either transmits captured voice inputs to the voice assistance server **112** for processing, or first pre-processes the voice inputs and transmits the pre-processed voice inputs to the voice assistance server **112** for processing. In some implementations, the voice assistance server **112** and the electronic device **190** has some shared and some divided responsibilities regarding processing of voice inputs and noise mitigation, and the programs, modules, and data structures shown in FIGS. 2A-2B may be included in both or divided amongst the voice assistance server **112** and the electronic device **190**. Other programs, modules, and data structures shown in FIGS. 2A-2B (e.g., voice models data **550**, voice model training module **554**), or analogues thereof, may also be included in the voice assistance server **112**.

Each of the above identified elements may be stored in one or more of the previously mentioned memory devices, and corresponds to a set of instructions for performing a function described above. The above identified modules or programs (i.e., sets of instructions) need not be implemented as separate software programs, procedures, modules or data structures, and thus various subsets of these modules may be combined or otherwise re-arranged in various implementations. In some implementations, memory **706**, optionally, stores a subset of the modules and data structures identified above. Furthermore, memory **706**, optionally, stores additional modules and data structures not described above.

Example Device Leadership Negotiation

FIGS. 4A-4B illustrate an example of a device leadership negotiation between multiple electronic devices (e.g., electronic devices **190**) in accordance with some implementations. In an operating environment with multiple electronic devices **190** (e.g., operating environment **100**), when a user in the environment makes a voice input, one or more of the multiple electronic devices **190** detect and receive the voice input. The electronic devices **190** determine a leader amongst themselves to respond to the user's voice input and to detect further voice inputs from the user.

FIG. 4A shows three voice-activated electronic devices **1002-1** thru **1002-3** (e.g., electronic devices **190**) in an operating environment (e.g., operating environment **100**).

21

They may all be in the same space as a user (not shown), e.g., in the same room. The user makes a voice input (e.g., an utterance) (not shown), and the three electronic devices **1002-1** thru **1002-3** detect and capture a sample of the voice input at varying degrees or levels of quality. The quality of a captured sample of the voice input at an electronic device may be affected by one or more factors, including but not limited to surrounding noise, capabilities of the device, distance and position of the device relative to the user, position of the device within the space, and so on.

In some implementations, each of the electronic devices determines a level of quality of the respective sample of the voice input captured by the electronic device. The quality level may be expressed as a score or some other value. For example, in FIG. 4A, each of the electronic devices **1002** has captured a respective sample of a user's voice input and has determined as score for its respective sample; the sample at electronic device **1002-1** is scored as 0.98 (on a scale of 0 to 1, with 1 corresponding to the highest quality), the sample at electronic device **1002-2** is scored as 0.37, and the sample at electronic device **1002-3** is scored as 0.61. In some implementations, the recognition quality module **566** (FIG. 2B) at the electronic device **1002** determines the quality level of the respective captured sample. In some implementations, criteria for evaluating the quality of a voice sample include, for example, loudness or amplitude of the sample, presence or absence of noise (e.g., cross-talk, hissing) and corresponding level, presence or absence of distortion (e.g., echo) and corresponding level, frequency profile of the sample, and so forth.

The electronic devices **1002** communicate their quality scores with each other. Thus, for example, electronic device **1002-1** receives the quality scores from electronic devices **1002-2** and **1002-3**. Electronic device **1002-2** receives the quality scores from electronic devices **1002-1** and **1002-3**. Electronic device **1002-3** receives the quality scores from electronic devices **1002-1** and **1002-2**. If all of the electronic devices **1002** are on the same local network (e.g., local network **104**, FIG. 1), the electronic devices **1002** exchange scores through the local network. For example, an electronic device **1002** may send its quality score to the other electronic devices **1002** by sending multicast messages within the local network. If at least one of the electronic devices **1002** is not on the local network, scores may be sent to server system **140**, and the device leadership moderator module **780** of the voice assistance server **112** distributes the scores to the electronic devices **1002**. Alternatively, the device leadership moderator module **780** determines which of the devices is to be the leader based on the scores received, and sends the result of the determination to the devices. In some implementations, the electronic devices **1002** discover each other and their locations through the voice assistance server **112** and device registry **118** (e.g., a device **1002** periodically checks in with the voice assistance server **112** and is informed of other devices associated with the same user account based on information in the device registry **118**).

In some implementations, if just one of the multiple electronic devices **1002** captured a sample of the voice input (e.g., the other electronic devices **1002** were too far away from the user to detect the voice input), the device that captured the voice input sample is determined to be the leader. In some implementations, if an electronic device **1002** failed to capture a sample of the voice input (e.g., the device was too far away from the user to detect the voice input), the quality score for that device for the voice input is 0.00 or a null score; a device with a "0.00" or null score is eliminated from contention to be the leader.

22

When an electronic device **1002** receives the scores from the other electronic devices **1002**, the electronic device compares its quality score with the scores from the other devices. If its score is not the best (e.g., highest) amongst the scores, the electronic device forgoes outputting responses (e.g., responses generated by the server system **140**) to the voice input (e.g., by powering down or "turning off" the speaker **520**). If its score is the best amongst the scores, the electronic device outputs responses to the voice input and subsequent voice inputs; this device is determined to be the "leader." In some implementations, the leader also proceeds to detect and process or pre-process subsequent voice inputs from the user (whether it be the leader processing the voice inputs and generating the responses, the leader pre-processing the voice inputs for transmission to the voice assistance server **112** which generates the responses, or the leader simply transmitting the voice inputs to the voice assistance server **112** which generates the response), and the non-leaders forgo receiving, detecting, and processing subsequent voice inputs from the user.

In some implementations, the leader determination is performed in response to a user's voice input to awaken the device (e.g., a voice input that includes the hotword). In some implementations, the leader determination may be performed in response to each voice input from the user or at intervals of a predefined number of voice inputs from the user (e.g., at every 5th voice input from the user). In this way, the leadership determination may be updated to account for changing conditions (e.g., the user moving around, the noise level around the devices change, etc.).

In some implementations, the leader determination based on quality scores may be overridden based on additional criteria. That is, a device may be determined to be the leader even if it would otherwise not be determined to be the leader based merely on the quality scores. The additional criteria may include, for example, a type of request or command in the voice input and a state of the device. For example, if the voice input includes a request to play or stop playing a video, then the request is of particular relevance to an electronic device **1002** that is capable of playing videos (e.g., a smart TV, a cast device) and is not particularly relevant to an electronic device **1002** that is an audio-only device (e.g., sound system, portable speaker). When the request is of particular relevance to an electronic device **1002**, then that electronic device **1002** may be determined to be the leader even if the quality score for that device is not the highest. In some implementations, if the request is of particular relevance to multiple electronic devices in the space, then the one with the best quality score amongst the devices to which the request has particular relevance is determined to be the leader. As another example, if the command in the voice input is a command to change device state (e.g., "screen off"), then a device in a state to which the command is applicable (e.g., a smart TV whose screen is currently on) may be determined to be the leader despite a lower quality score, and a device in a state to which the command is not applicable (e.g., a smart TV whose screen is already off) is passed over for leadership. Further, if the request/command is not of particular relevance to any of the devices **1002** (e.g., the devices **1002** are audio-only voice interface devices that can communicate with a non-voice-interface cast device through the voice interface server **112**, and the command is a request to play video via the cast device), the leadership determination is determined based on quality scores as described above. In some implementations, multiple leadership determination criteria (e.g., quality scores, relevance of command, and state of the device, as described above)

23

may be implemented as a multi-step leadership determination (e.g., determine relevance of the command and the device state to narrow leadership candidates, then determine leader based on quality scores; determine device with highest score, then check if command relevance or device state criteria apply to the other devices) or as a weighted determination (e.g., each criterion is accounted for and weighted in a weighted score).

FIG. 4B illustrates a result of the device leadership negotiation amongst the electronic devices **1002** in FIG. 4A. Electronic device **1002-1** determines that its quality score is the best amongst the three electronic devices **1002**, and it assumes leadership for outputting responses to the user. Electronic devices **1002-2** and **1002-3** determine that their quality scores are not the best amongst the three electronic devices **1002**, and power down their speakers **520** and forgo outputting responses to the user.

In some implementations, the electronic devices **1002** that are not the leader also power down their microphones **516**, and thus do not detect further voice inputs from the user; the leader device also is the leader with respect to detection and processing of further voice inputs from the user. In some implementations, the electronic devices **1002** that are not the leader maintain power to their microphones **516** and provide assistance re detection of further voice inputs from the user. For example, the leader device and the non-leader devices receive and record respective samples of a voice input and transmit them to the voice assistance server **112**, which may use the multiple samples together to better process the voice input (e.g., aggregate the samples, choose the best sample for processing, compare the samples to get the best parts of each).

In some implementations, the electronic devices **1002** repeat the leadership determination process as described above after each voice input from the user. In some implementations, the electronic devices **1002** repeat the leadership determination process as described above after intervals of a predefined number of voice inputs (e.g., determine a leader again at every 5th voice inputs or at every 10th voice input).

In some implementations, one or more of the electronic devices **1002** output a visual response after a leader is determined (e.g., along with the audio response to the voice input, the determined leader displays a notification or a predefined LED light pattern indicating that it is the leader).

In some implementations, each of the electronic devices **1002** outputs a visual indication (e.g., a predefined LED light pattern) that it is listening whenever it is listening to a user (e.g., receiving and detecting the user's voice input), and then just the leader device outputs the response to the voice input per the leadership determination. For example, when a user is speaking a voice input, the electronic devices **1002-1** thru **1002-3** display the visual indication to indicate that they are listening while receiving the voice input, and afterwards just the leader device **1002-1** outputs the response per the leadership determination.

In some implementations, an electronic device **1002** indicates that it is listening by displaying using a group of LEDs a characteristic visual pattern, such as one of the patterns shown in FIG. 4G of the '566 application (duplicated as FIG. 8D below). The visual patterns can be static (unchanging) or dynamic (giving the impression of movement through changes in the color, brightness, duty cycle, etc. of individual LEDs). In some implementations, an electronic device **1002** indicates that it is the leader among multiple listening devices by displaying another characteristic visual pattern using its LEDs. Similarly, multiple listening devices engaged in a negotiation to determine which device should

24

be the leader can indicate that the negotiation is ongoing by displaying another characteristic visual pattern using their respective LEDs.

Example Process for Multi-User Personalization

FIG. 5 is a flow diagram illustrating a method **1100** of responding to a voice input of a user in accordance with some implementations. The method **1100** is implemented at an electronic device **190** with one or more microphones, a speaker, one or more processors and memory storing one or more programs for execution by the one or more processors.

The electronic device **190** receives (1102) via the one or more microphones **516** a first voice input from a user. The electronic device **190** captures and records a sample of the first voice input via the microphone **516**. In some implementations, the first voice input is a freeform voice input, in the sense that the voice input need not be strictly limited to a predefined set of words and phrases within a predefined syntax; the voice input may be a natural language voice input, for example.

The electronic device **190** compares (1104) the first voice input to a set of voice models associated with a plurality of users associated with the electronic device **190**, where the voice models are trained to the electronic device. The user identification module **560** compares the recorded sample of the first voice input to voice models in voice models data **550**. The voice models data **550** include voice models or voice fingerprints of one or more users in the user domain that are associated with the electronic device **190** (e.g., users registered to the electronic device **190**). In some implementations, the voice models are trained by the voice model training module **554** to the electronic device **190**.

Based on the comparing, the electronic device **190** determines (1106) that the first voice input corresponds to a first user of the plurality of users. For example, the user identification module **560** identifies a voice model in voice models data **550** that best matches the first voice input, and in accordance with the identification of the match determines that the user speaking the first voice input is the user to which the matching voice model corresponds. In some implementations, the user identification module **560** also determines a confidence level or some other similar measure of the quality or closeness of the match between a voice model and the voice input, and identifies a match only if the match is the best and the confidence level is above a predefined threshold.

The electronic device **190** presents (1108) a response in accordance with the determination. In some implementations, the electronic device **190** generates the response in accordance with the determination and with the first voice input. The response includes an indication of the identity of the identified first user, to let the first user know that he has been identified. The response also includes a response to any voice request in the first voice input. For example, in response to a hotword voice input to awaken the electronic device **190**, the electronic device **190** generates and presents (e.g., outputs) a voice greeting that includes the first user's name (e.g., "Hello David," "Good morning Mary"). The response may include a voice output and/or instructions to another device to perform an operation. For instructions to perform an operation, the indication of the identified user's identity is included, so that the operation is performed with respect to the identified user. For example, if the operation is playing music from a music subscription, the device plays music from the identified user's music subscription.

25

In some implementations, the response is an audible voice output, and presenting the response includes outputting the audible voice output. The electronic device **190** outputs the audible voice output through the speaker **520**.

In some implementations, the electronic device determines a user request based on the first voice input, and generates the response in accordance with the determined user request. In some implementations, generating the response in accordance with the determination and with the first voice input includes these determining and generating operations. In addition to identifying the speaker of the voice input, the electronic device **190** determines the user request in the voice input (e.g., a media playback request, a request for news headlines, a request for emails to be read, etc.), and generates a response that is responsive to the request (e.g., an audible acknowledgement of the request, an audible inquiry to the user for more information needed to fulfill the request).

In some implementations, the determined user request includes a request to the electronic device to awaken, and the response includes an audible greeting, where the greeting includes an indication of an identity of the determined first user. If the voice input includes a hotword (i.e., the voice input is a request to the electronic device **190** to awaken), the generated response includes an audible greeting that includes an identity of the first user (e.g., “Good morning Dave”). The greeting indicates that the electronic device **190** has awakened and indicates the identity of the user as identified by the electronic device **190**.

In some implementations, the determined user request includes a request for information, and generating the response in accordance with the determined request includes obtaining the requested information with respect to the first user. If the request is a request for information (e.g., a request to read the user’s emails, request to say out loud the user’s upcoming calendar events), the electronic device **190** accesses the identified user’s information (e.g., the user’s emails, the user’s calendar) to generate the response. In some implementations, obtaining the requested information with respect to the determined first user comprises accessing an account associated with the first user; in order to obtain the identified user’s information for the response, the electronic device **190** accesses the user’s accounts (e.g., the user’s email account, the user’s calendar account). At the electronic device **190**, the user account information may be stored in the voice device settings **546**.

In some implementations, the electronic device **190** receives a second voice input, compares the second voice input to the set of voice models, based on the comparing, determines that the second voice input corresponds to a second user of the plurality of users, and presents a response in accordance with the determination that the second voice input corresponds to the second user. In some implementations, the electronic device generates the response in accordance with the determination and with the second voice input. If a different user associated with the electronic device **190** speaks a voice input to the electronic device **190**, the electronic device **190** identifies the different user in a similar manner as the identification of the first user—comparing the different user’s voice input to the voice models and finding a best match. With the identification of the different user, the electronic device **190** is able to generate a response based on this new identification. Thus, the electronic device **190** is able to switch between associated users by comparing the voice inputs to the voice models and identifying the speaking user based on the comparing, negating a need for the

26

users to explicitly speak a command to switch the focus of the electronic device **190** from one user to another user.

In some implementations, the electronic device **190** receives a third voice input, compares the third voice input to the set of voice models, based on the comparing, determines that the third voice input corresponds to none of the plurality of users, and in accordance with the determination that the third voice input corresponds to none of the plurality of users, restricts functionality of the electronic device. If a user who is not associated with the electronic device **190** speaks a voice input to the electronic device **190**, the electronic device **190** receives the voice input and attempts to identify the user. As this user is not associated with the electronic device **190** and thus does not have an associated voice model in the voice models data **550**, the electronic device **190** determines that this user is an unidentified user. For this unidentified user, the electronic device **190** may restrict its functionality in order to prevent personal or private information from being accessed. For example, the electronic device **190** may act on only requests from the unidentified user that involve non-private or non-personal information (e.g., request for sports scores, news, stocks, weather). In some implementations, the electronic device **190** may deny the unidentified user access to any functionality of the electronic device **190** and the voice assistant service (e.g., by ignoring the unidentified user’s voice inputs or responding with a response indicating that the user is not authorized). In some implementations, the restriction of functionality to unidentified users is based on settings for the device (e.g., voice device settings **546**). The device owner can set, for example, whether to restrict functionality, for unidentified users, entirely or just to functionality that does not involve personal or private information.

In some implementations, the electronic device includes a display, and displays on the display an indication of an identity of the determined first user. If the electronic device **190** includes a display **524**, the electronic device **190** may display the identified user’s identify (e.g., display the identified user’s name, display a greeting with the identified user’s name, displaying the identified user’s picture, avatar, or other image associated with the identified user) on the display **524** to indicate that the electronic device **190** has identified a speaker of the voice input and to give the user an opportunity to verify the identification.

In some implementations, the electronic device includes a display, and displays on the display a visual user interface associated with the determined first user. If the electronic device **190** includes a display **524**, the electronic device **190** may display a visual user interface identified with the identified user (e.g., a user interface with a theme associated with the identified user applied). In some implementations, the electronic device **190** displays on the LEDs **522** a pattern associated with the identified user.

In some implementations, the electronic device **190** receives the voice input (e.g., capture a sample of the voice input), as in step **1102**, and sends the voice input to the voice assistance server **112**. The voice assistance server **112** compares the voice input to a set of voice models (e.g., voice models in user voice profiles **744**) associated with a plurality of users associated with the electronic device **190**, and based on the comparing, determines the identity of the user who spoke the voice input, similar to steps **1104** and **1106**. The voice assistance server **112** generates the response to the voice input, and transmits the generated response to the electronic device **190**, which outputs the response. In some implementations, the voice assistance server **112** identifies the user, and transmits the user identification result to the

27

electronic device 190, which generates and outputs a response in accordance with the identification result received from the voice assistance server 112.

In some implementations, the electronic device 190 receives the voice input (e.g., capture a sample of the voice input), as in step 1102, performs pre-processing on the voice input, and sends the pre-processed voice input to the voice assistance server 112. The pre-processing may include identification of the speaking user, as in steps 1104 and 1106, and the identification result accompanies the voice input to the voice assistance server 112. Alternatively, the user identification is performed by the voice assistance server 112. The voice assistance server 112 receives the pre-processed voice input, identifies the speaking user if not already performed by the electronic device 190 (similar to steps 1104 and 1106), generates the response to the voice input, and transmits the generated response to the electronic device 190, which outputs the response.

In some implementations, the electronic device 190 receives the voice input (e.g., capture a sample of the voice input) and either sends the voice input to the voice assistance server 112 or performs pre-processing and sends the pre-processed voice input to the voice assistance server 112. The voice assistance server 112 compares the voice input to a set of voice models (e.g., voice models in user voice profiles 744) associated with a plurality of users associated with the electronic device 190, and based on the comparing, determines the identity of the user who spoke the voice input, as in steps 1104 and 1105. The voice assistance server 112 generates the response to the voice input and transmits the generated response to the electronic device 190, which outputs the response. In some implementations, the voice assistance server 112 identifies the user, and transmits the user identification result to the electronic device 190, which generates and outputs a response in accordance with the identification result received from the voice assistance server 112.

As described above, the speaking user is identified by matching a voice model or voice fingerprint to the voice input. In some implementations, the electronic device 190 is configured to perform a process for capturing and training voice models or voice fingerprints. For example, when a user is to be associated with the electronic device 190 (e.g., registered to the device), the electronic device 190 (e.g., voice model training module 554) may initiate and guide the user to be associated with the electronic device 190 through a step-by-step process to capture and train a voice model of the user. During the process, the electronic device 190 may prompt the user to speak one or more words or phrases (e.g., the hotword, a phrase or sentence that includes multiple distinct phonemes) in order to generate and train the voice model. The voice model training module 554 may direct the LED control module 540 to illuminate various visual patterns on the LEDs 522 to indicate to the user a status of the voice model capture and training process. In some implementations, the electronic device 190 stores the trained voice models in voice models data 550, and optionally transmits the trained voice models to the voice assistance server 112 for storage (e.g., in user voice profiles 744).

In some implementations, a given user has voice models for each electronic device 190 with which the user is associated. The voice models at the respective devices 190 are trained to the particular device, thus accounting for the particular devices and the environments surrounding the respective devices.

In some implementations, the voice model training process is performed by the voice assistance server 112 in

28

conjunction with the electronic device 190. For example, when a user is to be associated with an electronic device 190, a voice model training module (not shown) at the voice assistance server 112 transmits instructions and audio outputs to the electronic device 190. The electronic device 190 executes the instructions and output the audio outputs to guide the user through the training process similar to the training process performed by the voice model training module 554 described above. The electronic device 190 captures the user's spoken voice inputs to the training process and transmits the captured voice inputs to the voice assistance server 112, which generates the voice models and stores them in user voice profiles 744. In some implementations, the voice assistance server 112 transmits to the electronic device 190 voice models of users associated with the electronic device 190 for storage (e.g., in voice models data 550) and local use.

In some implementations, when a voice model is generated and trained for a user, an authentication token is also generated for the user. The authentication token authenticates the user to the voice assistance server 112. The speaker's voice input is matched to voice models associated with different users. When a matching voice model is found for the speaker's voice input, the authentication token for the user account associated with the matching voice model is used. In some implementations, authentication tokens are pre-generated but initially not associated with any voice model. The training process associates a voice model trained as a result of the training process to a pre-generated authentication token. In some implementations, after the electronic device 190 identifies the speaker of a voice input, the voice input (or the data containing the user request and associated parameters determined from the voice input) is transmitted to the voice assistance server 112 along with the authentication token associated with the speaker; the authentication token authenticates the speaker to the voice assistance server 112. In some implementations, a speaker may be associated with an authentication token of a user who has granted permission to any speaker. For example, a user associated with an electronic device 190 may configure his account to allow anyone to use his music service login credentials, so that other users can access any connected output device (e.g., speakers) near the user's device 190.

In some implementations, an electronic device 190 identifies the speaking user by detecting a nearby key device associated with the speaking user (e.g., using Bluetooth low energy or ultrasonic RF).

In some implementations, a content host 114 generates the audible response to a voice input. For example, the voice input may include a request for the balance in a bank account associated with the user (e.g., "what is the balance in my Chase account?"). For such a voice input, the electronic device 190 or the voice assistance server 112 identifies the speaking user, as described above. After identifying the user and obtaining the bank account information (from, e.g., user data 742), the electronic device 190 or the voice assistance server 112 transmits a query for the account balance (e.g., through an application programming interface (API)) to a content host 114 associated with the bank where the bank account is located (e.g., the bank's database system). The bank content host obtains the balance from its databases and generates an audible response responsive to the request. The bank content host transmits the audible response to the electronic device 190 or to the voice assistance server 112 (which then transmits the audible response to the electronic device 190), and the electronic device 190 outputs the audible response. In this way, transmission and storage of

account data in numerical form over networks and on the device is reduced, which promotes security and privacy of user data.

Example Process for Device Leadership Negotiation

FIG. 6 is a flow diagram illustrating a method **1200** of negotiating device leadership among multiple voice interface devices, in accordance with some implementations. The method **1200** is implemented at an electronic device (e.g., device **1002-1**, **1002-2**, or **1002-3**) of two or more electronic devices **190** (electronic devices **1002**), where each of the two or more electronic devices includes one or more microphones, a speaker, one or more processors and memory storing one or more programs for execution by the one or more processors.

The electronic device **190** detects (**1202**) a voice input. The electronic device **190** captures and records a sample of the voice input from a user via the microphone **516**.

The electronic device **190** determines (**1204**) a quality score for the detected voice input. The electronic device **190** determines a score, rating, or some other measure of the quality of the captured voice input sample. The quality of the sample may be affected by multiple factors, including the user's distance from the device, the volume at which the user spoke the voice input, the surrounding environment and noise, the capabilities of the device, and so on. In some implementations, criteria for evaluating the quality of a voice sample include, for example, loudness or amplitude of the sample, presence or absence of noise (e.g., cross-talk, hissing) and corresponding level, presence or absence of distortion (e.g., echo) and corresponding level, frequency profile of the sample, and so forth.

The electronic device **190** communicates (**1206**) the quality score to the other devices of the plurality of electronic devices, and receives (**1208**) quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices. The electronic device **190** transmits the score for its voice input sample to the other devices of the plurality of electronic devices, and receives scores from the other devices for their respective samples of the voice input.

In accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, the electronic device outputs (**1210**) an audible response and/or a visual response (e.g., a notification or a LED light pattern) to the detected voice input, and the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input. In some implementations, the device with the highest score (or more generally, the best score) amongst the devices with respect to the voice input optionally pre-processes its respective voice input sample, and transmits the voice input sample or the pre-processed voice input sample to the voice assistance server **112**, which generates a response, which includes audible output (e.g., voice confirmation of the user request, reading out of the user-requested information), to the voice input and transmits the response to the devices. Alternatively, the device with the highest score generates the response to the voice input. In either situation, the device with the highest score (e.g., device **1002-1**) outputs the response, and the other devices (e.g., devices **1002-2** and **1002-3**), which have the lower scores, do not output the response. In some implementations, the electronic device that is the leader (e.g., the device

with the highest score in the example here) also is the leader with respect to processing or pre-processing subsequent voice inputs from the user.

In accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, the electronic device **190** forgoes outputting a response to the detected voice input. If a device determines that its score is not the highest amongst the devices with respect to the voice input (e.g., devices **1002-2** and **1002-3**), the device does not output a response to the voice input. For example, the devices with the lower scores power down their speakers **520**. In some implementations, the devices with the lower scores also power down their microphones **516** so that just the device with the highest score is detecting further voice inputs. In some implementations, in accordance with the determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, the electronic device **190** forgoes outputting an audible response to the detected voice input, and may output a visual response (e.g., a visual indication that the device was not determined to be the leader).

In some implementations, the plurality of electronic devices is communicatively coupled through a local network, and the communicating and receiving are performed through the local network. The electronic devices **190** are communicatively coupled through a local network (e.g., local network **104**), and may transmit their scores to each other through the local network. For example, a device may multi-cast or broadcast its score through the local network.

If at least one of the devices is not on the local network, then the devices may transmit their scores to the voice assistance server **112**. In some implementations, the device leadership moderator module **780** transmits the received scores to the devices, and the devices determine whether their respective scores are the highest. In some implementations, the device leadership moderator module determines which score is the highest and transmits a result of the determination to the devices. In some implementations, a hub device may include a module similar to the device leadership moderator module **780** and serve in a similar role.

In some implementations, the quality score includes a confidence level of detection of the voice input; the quality score is a confidence level value for the voice input sample. In some implementations, the quality score includes a signal-to-noise rating of detection of the voice input; the quality score is a signal-to-noise rating or value (e.g., signal-to-noise ratio) for the voice input sample.

In some implementations, the electronic device **190** recognizes a command in the voice input, and in accordance with a determination that a type of the command is related to the first electronic device, outputs an audible and/or a visual response to the detected voice input. If the command or request in the voice input is recognized as one with particular relevance to a device of the multiple devices that detected the voice input, then that device outputs a response to the voice input and the other devices forgo outputting the response, even if the device to which the command has particular relevance does not have the highest score among the devices with respect to the voice input. Examples of commands or requests that have particular relevance include video playback requests (e.g., for cast or TV devices) and music playback requests (e.g., for audio systems or speaker systems). Additionally, a command or request may have particular relevance to a device if the command or request

31

identifies the particular device that the user wishes to fulfill the request (e.g., the command “Play video on my bedroom TV” has particular relevance to the TV device in the bedroom because the command explicitly identifies that device), or the command/request includes a device state parameter that is applicable to the current state of the device (e.g., a “screen off” command is relevant to a device whose screen is on and not relevant to a device without a screen or whose screen is off).

In this way, the multiple devices capture respective samples of the voice input, and a determination as to which sample is best is made. Based on this determination, one of the devices respond to the voice input and further voice inputs from the user. In some implementations, the multiple devices repeat the process described above for each voice input or at intervals of a predefined number of voice inputs (e.g., determine the leader again at every 5th voice input or at every 10th voice input).

In some implementations, the multiple devices detect the voice input and send their captured voice input samples to the voice assistance server 112. The voice assistance server 112 aggregates the samples or selects the best quality sample for processing or compares the samples to get the best parts of each. In some implementations, when the multiple devices are receiving and detecting the voice input, the devices each display a visual indication (e.g., a predefined LED pattern) that it is listening to the user.

Example Process for Noise Mitigation

FIG. 7 is a flow diagram illustrating a method 1300 of mitigating noise around a voice interface device in accordance with some implementations. The method 1300 is implemented at an electronic device 190 with one or more microphones, a speaker, one or more processors and memory storing one or more programs for execution by the one or more processors. The electronic device 190 is configured to be awaked by any of a plurality of affordances, including a voice-based affordance (e.g., a voice input that includes a hotword).

The electronic device determines (1302) a current noise profile of an environment around the electronic device. The electronic device 190, while it is idle, may run the noise module 558 (e.g., noise determination module 572) to determine a noise profile for its surroundings. In some implementations, determining the current noise profile includes determining a noise level (e.g., volume) of the noise, and capturing and processing the ambient noise. The processing may include, for example, determining frequencies and other parameters of the ambient noise.

The electronic device determines (1304) whether the noise profile interferes with the voice-based affordance. The electronic device 190 compares the determined noise profile with noise profiles in the noise profile store 576 and evaluates the parameters of the noise profile to determine if the noise profile (i.e., the current noise surrounding the electronic device 190) interferes with the voice-based affordance.

In accordance with a determination that the noise profile interferes with the voice-based affordance, the electronic device 190 presents (1306) a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device. The hint directs the user to use another affordance other than the voice-based affordance to awaken the electronic device 190. In some implementations, the hint is also or otherwise includes indication that the surrounding noise interferes with

32

hotword recognition. For example, the electronic device 190 may display a predefined pattern on the LEDs 522 that indicates surrounding noise interference with the voice-based affordance and hints to the user that the user should use another affordance to wake up the device (e.g., pressing a button 512, touching a touch sensor array 514, use another device communicatively coupled to the electronic device 190, such as a smartphone). The indication/hint may be displayed on the display 524 or output as sounds or tones through speaker 520.

In some implementations, wherein the voice-based affordance includes a spoken hotword. The voice-based affordance includes the user speaking the hotword to awaken the electronic device 190, as described above.

In some implementations, determining whether the noise profile interferes with the voice-based affordance includes determining whether the noise profile negatively affects recognition of the hotword in voice inputs detected by the electronic device. The electronic device 190 determines if the surrounding noise (represented by the noise profile) would negatively affect detection/recognition of the hotword by the electronic device 190 (e.g., is the noise sufficiently loud to drown out the user’s voice input with the hotword, does the noise have a frequency profile that would interfere with the user’s voice input with the hotword, would the noise distort the user’s voice input with the hotword).

In some implementations, determining whether the noise profile negatively affects recognition of the hotword in voice inputs includes comparing the noise profile to a noise level threshold of a hotword recognition process associated with the electronic device. The recognition analysis module 568 of the electronic device 190 (or the voice assistance server 112) analyzes the hotword recognition module 562 of the electronic device 190 (e.g., when the electronic device 190 is idle) to determine noise thresholds at which noise would cause problems for the algorithm or process used by the module. These thresholds are saved and may be compared to current noise profiles to determine if the current noise profile would pose a problem for the hotword recognition module 562. The analysis of the hotword recognition module is with respect to the electronic device 190, and thus the threshold may be different between devices.

In some implementations, determining whether the noise profile negatively affects recognition of the hotword in voice inputs includes comparing the noise profile to a voice volume level threshold, where the voice volume level threshold is defined based on a predefined distance from the electronic device. The user volume threshold module 570 of the electronic device 190 (or the voice assistance server 112) performs analyses of the user’s voice inputs to determine a maximum comfortable volume level at which the user will speak voice inputs, with the determination based on a predefined distance (e.g., a predefined distance that is based on a “typical user distance” beyond which voice inputs attenuate).

In some implementations, determining whether the noise profile negatively affects recognition of the hotword in voice inputs includes comparing the noise profile to a predetermined set of noise profiles associated with the electronic device, wherein the noise profiles of the predetermined set of noise profiles are known to negatively affect recognition of the hotword by the electronic device in voice inputs. The electronic device 190 compares the noise profiles to previous noise profiles (e.g., in noise profiles store 576) that interfered with hotword detection.

In some implementations, the electronic device includes a button, and the affordance of the plurality of affordances

other than the voice-based affordance includes activating the button. The electronic device **190** includes a physical push button **512** and the user can awaken the device by activating the button **512** in lieu of speaking a voice input with the hotword.

In some implementations, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device includes illuminating the button on the electronic device corresponding to a button-based affordance of the plurality of affordances. The button **512** may include lighting (e.g., an embedded LED light), and the electronic device **190** may hint the user to use the button **512** instead of speaking the hotword by lighting up the button **512**.

In some implementations, presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device includes outputting a predefined audible sound. The electronic device **190** may hint to the user to use an alternate affordance by outputting a predefined sound or tone (or set of sounds or tones) through the speaker **520**.

In some implementations, the electronic device includes a display, and presenting a hint to a user to use an affordance of the plurality of affordances other than the voice-based affordance to awaken the electronic device includes displaying the hint on a display. The electronic device **190** may hint to the user to use an alternate affordance by displaying a hint message on the display **524**.

In some implementations, the electronic device includes one or more LED lights, and presenting a hint to a user to use an affordance of the plurality of affordances different from the voice-based affordance to awaken the electronic device includes displaying the hint using the one or more LED lights. The electronic device **190** may hint to the user to use an alternate affordance by displaying a predefined light pattern on the LEDs **522**.

FIGS. **8A** and **8B** are a front view **400** and a rear view **420** of a voice-activated electronic device **190** in accordance with some implementations. The electronic device **190** is designed as warm and inviting, and fits naturally in many areas of a home. The electronic device **190** includes one or more microphones **402** and an array of full color LEDs **404**. The full color LEDs **404** could be hidden under a top surface of the electronic device **190** and invisible to the user when they are not lit. In some implementations, the array of full color LEDs **404** is physically arranged in a ring. Further, the rear side of the electronic device **190** optionally includes a power supply connector **408** configured to couple to a power supply.

In some implementations, the electronic device **190** presents a clean look having no visible button, and the interaction with the electronic device **190** is based on voice and touch gestures. Alternatively, in some implementations, the electronic device **190** includes a limited number of physical buttons (e.g., a button **406** on its rear side), and the interaction with the electronic device **190** is further based on press on the button in addition to the voice and touch gestures.

In some implementations, given simplicity and low cost of the electronic device **190**, the electronic device **190** includes an array of full color light emitting diodes (LEDs) rather than a full display screen. A LED design language is adopted to configure illumination of the array of full color LEDs and enable different visual patterns indicating different voice processing state of the electronic device **190**. The LED Design Language consists of a grammar of colors, patterns, and specific motion applied to a fixed set of full

color LEDs. The elements in the language are combined to visually indicate specific device states during the use of the electronic device **190**. In some implementations, illumination of the full color LEDs aims to clearly delineate the passive listening and active listening states of the electronic device **190** among other important states. Placement of the full color LEDs complies with physical constraints of the electronic device **190**, and the array of full color LEDs could be used in a speaker that is made by a third party original equipment manufacturer (OEM) based on specific technology (e.g., Google Assistant).

When the array of full color LEDs is used in a speaker that is made by a third party OEM based on specific technology, the full color LEDs and the LED design language are configured to fit a corresponding physical user interface of the OEM speaker. In this situation, device states of the OEM speaker remain the same, while specific visual patterns representing the device states could be varied (for example, the colors of the full color LEDs could be different but are displayed with similar animation effects).

In a voice-activated electronic device **190**, passive listening occurs when the electronic device **190** processes audio inputs collected from its surrounding environment but does not store the audio inputs or transmit the audio inputs to any remote server. In contrast, active listening occurs when the electronic device **190** stores the audio inputs collected from its surrounding environment and/or shares the audio inputs with a remote server. In accordance with some implementations of this application, the electronic device **190** only passively listens to the audio inputs in its surrounding environment without breaching privacy of users of the electronic device **190**.

FIG. **8C** is a top view of a voice-activated electronic device **190** in accordance with some implementations, and FIG. **8D** shows six visual patterns displayed by an array of full color LEDs for indicating voice processing states in accordance with some implementations. In some implementations, the electronic device **190** does not include any display screen, and the full color LEDs provide a simple and low cost visual user interface compared with the a full display screen. The full color LEDs could be hidden under a top surface of the electronic device and invisible to the user when they are not lit. Referring to FIGS. **8C** and **8D**, in some implementations, the array of full color LEDs are physically arranged in a ring.

Reference has been made in detail to implementations, examples of which are illustrated in the accompanying drawings. In the detailed description above, numerous specific details have been set forth in order to provide a thorough understanding of the various described implementations. However, it will be apparent to one of ordinary skill in the art that the various described implementations may be practiced without these specific details. In other instances, well-known methods, procedures, components, circuits, and networks have not been described in detail so as not to unnecessarily obscure aspects of the implementations.

It will also be understood that, although the terms first, second, etc. are, in some instances, used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first device could be termed a second device, and, similarly, a second device could be termed a first device, without departing from the scope of the various described implementations. The first device and the second device are both types of devices, but they are not the same device.

35

The terminology used in the description of the various described implementations herein is for the purpose of describing particular implementations only and is not intended to be limiting. As used in the description of the various described implementations and the appended claims, the singular forms “a”, “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term “and/or” as used herein refers to and encompasses any and all possible combinations of one or more of the associated listed items. It will be further understood that the terms “includes,” “including,” “comprises,” and/or “comprising,” when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

As used herein, the term “if” is, optionally, construed to mean “when” or “upon” or “in response to determining” or “in response to detecting” or “in accordance with a determination that,” depending on the context. Similarly, the phrase “if it is determined” or “if [a stated condition or event] is detected” is, optionally, construed to mean “upon determining” or “in response to determining” or “upon detecting [the stated condition or event]” or “in response to detecting [the stated condition or event]” or “in accordance with a determination that [a stated condition or event] is detected,” depending on the context.

For situations in which the systems discussed above collect information about users, the users may be provided with an opportunity to opt in/out of programs or features that may collect personal information (e.g., information about a user’s preferences or usage of a smart device). In addition, in some implementations, certain data may be anonymized in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user’s identity may be anonymized so that the personally identifiable information cannot be determined for or associated with the user, and so that user preferences or user interactions are generalized (for example, generalized based on user demographics) rather than associated with a particular user.

Although some of various drawings illustrate a number of logical stages in a particular order, stages that are not order dependent may be reordered and other stages may be combined or broken out. While some reordering or other groupings are specifically mentioned, others will be obvious to those of ordinary skill in the art, so the ordering and groupings presented herein are not an exhaustive list of alternatives. Moreover, it should be recognized that the stages could be implemented in hardware, firmware, software or any combination thereof.

The foregoing description, for purpose of explanation, has been described with reference to specific implementations. However, the illustrative discussions above are not intended to be exhaustive or to limit the scope of the claims to the precise forms disclosed. Many modifications and variations are possible in view of the above teachings. The implementations were chosen in order to best explain the principles underlying the claims and their practical applications, to thereby enable others skilled in the art to best use the implementations with various modifications as are suited to the particular uses contemplated.

What is claimed is:

1. A method performed at a first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more microphones,

36

one or more processors, and memory storing one or more programs for execution by the one or more processors, the method comprising:

detecting a voice input from a user;
determining a first value associated with the voice input;
receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices; and
in accordance with a determination that the first value is higher than the second values, responding to the detected input.

2. The method of claim 1, further comprising:

in accordance with a determination that the first value is not higher than the second scores, forgoing responding to the detected input.

3. The method of claim 2, wherein forgoing responding to the detected input further comprises changing a state of operation of the first electronic device.

4. The method of claim 1, further comprising:

in accordance with a determination that the first value is not higher than the second values, identifying a criterion associated with the voice input; and
in accordance with a determination that the criterion is the most relevant to the first electronic device, responding to the detected input.

5. The method of claim 4, further comprising:

determining a state of the first electronic device;
wherein the determination that the criterion is the most relevant to the first electronic device is based on the state of the electronic device.

6. The method of claim 4, wherein the criterion comprises one or more of: a type of request, a command in the voice input, and a state of each electronic device of the plurality of electronic devices.

7. The method of claim 1, further comprising:

at the first electronic device:
detecting subsequent voice inputs from the user;
pre-processing the subsequent voice inputs; and
transmitting the pre-processed voice inputs to a server system;
wherein the other electronic devices forgo the detecting, pre-processing, and transmitting.

8. The method of claim 1, wherein responding to the detected input comprises outputting an audible and/or a visual response to the detected voice input.

9. The method of claim 1, wherein the first electronic device further comprises a speaker, and responding to the detected input further comprises outputting an audible response.

10. A first electronic device of a plurality of electronic devices, the first electronic device comprising:

one or more microphones;
one or more processors; and
memory storing one or more programs to be executed by the one or more processors, the one or more programs comprising instructions for:
detecting a voice input from a user;
determining a first value associated with the voice input;

receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices; and
in accordance with a determination that the first value is higher than the second values, responding to the detected input.

11. The first electronic device of claim 10, the one or more programs further comprising instructions for:

37

recognizing a command in the voice input; and
in accordance with a determination that a type of the
command is related to the first electronic device,
responding to the detected voice input.

12. The first electronic device of claim 10, the one or more
programs further comprising instructions for:
communicating to the other devices of the plurality of
electronic devices to forgo responding to the detected
output.

13. The first electronic device of claim 10, wherein the
instruction for detecting the voice input is performed at
intervals of a predefined number of voice inputs.

14. The first electronic device of claim 10, wherein:
the plurality of electronic devices is communicatively
coupled through a local network; and
the receiving is performed through the local network.

15. The first electronic device of claim 10, wherein the
first electronic device is configured to be awakened by a
plurality of affordances including a voice-based affordance;
and

wherein detecting the voice input is in response to the
awakening of the first electronic device.

16. A system comprising a plurality of electronic devices,
each of the electronic devices including one or more micro-

38

phones, one or more processors, and memory storing one or
more programs for execution by the one or more processors,
each of the electronic devices programmed to:

detect a voice input;

determine a first value associated with the detected voice
input;

receive respective second values from other electronic
devices of the plurality of electronic devices;

compare the first value and the respective second values;
and

in accordance with a determination that the first value is
higher than the respective second values, respond to the
voice input.

17. The system of claim 16, wherein the first value
comprises a first quality score of the voice input.

18. The system of claim 17, wherein the first quality score
comprises a confidence level of detection of the voice input.

19. The system of claim 17, wherein the first quality score
comprises a signal-to-noise rating of detection of the voice
input.

20. The system of claim 16, wherein each of the respec-
tive second values is indicative of a quality of the voice input
detected by a respective one of the other electronic devices.

* * * * *

Exhibit 4

(12) **United States Patent**
Mathews et al.

(10) **Patent No.:** **US 11,050,615 B2**
(45) **Date of Patent:** ***Jun. 29, 2021**

(54) **APPARATUS AND METHOD FOR SEAMLESS COMMISSIONING OF WIRELESS DEVICES**

(56) **References Cited**

U.S. PATENT DOCUMENTS

(71) Applicant: **GOOGLE LLC**, Mountain View, CA (US)

4,049,973 A 9/1977 Lambert
4,393,342 A 7/1983 Matsuoka et al.
(Continued)

(72) Inventors: **Jeffrey P. Mathews**, Longmont, CO (US); **Lee R. Taylor**, Boulder, CO (US)

OTHER PUBLICATIONS

(73) Assignee: **Google LLC**, Mountain View, CA (US)

Kim et al., "A New Commissioning and Deployment Method for Wireless Sensor Networks", Oct. 11-16, 2009, IEEE, 2009 Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, pp. 232-237, DOI: 10.1109/UBICOMM.2009.63 (Year: 2009).*

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(Continued)

(21) Appl. No.: **16/708,189**

Primary Examiner — Paul H Masur

(22) Filed: **Dec. 9, 2019**

(74) *Attorney, Agent, or Firm* — Butzel.Long

(65) **Prior Publication Data**

US 2020/0244522 A1 Jul. 30, 2020

Related U.S. Application Data

(63) Continuation of application No. 15/960,337, filed on Apr. 23, 2018, now Pat. No. 10,505,797, which is a (Continued)

(51) **Int. Cl.**
H04L 12/24 (2006.01)
H04W 76/10 (2018.01)
(Continued)

(52) **U.S. Cl.**
CPC **H04L 41/0806** (2013.01); **H04B 10/116** (2013.01); **H04W 4/70** (2018.02);
(Continued)

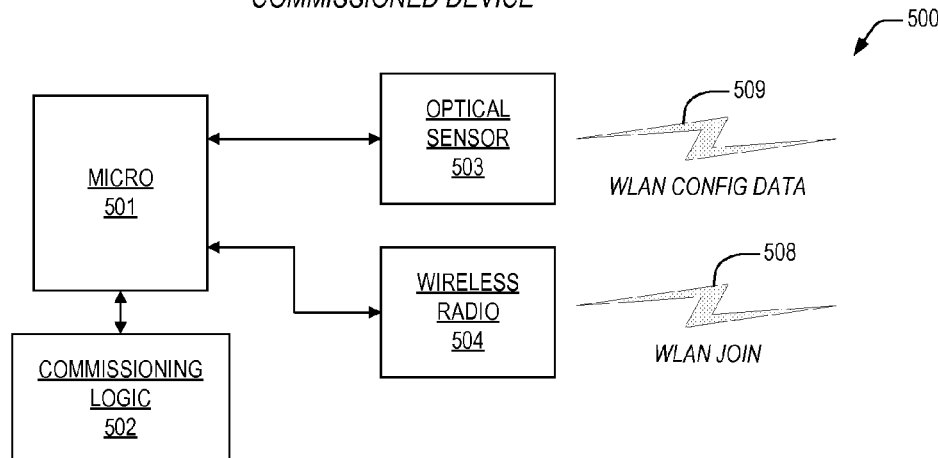
(58) **Field of Classification Search**
CPC H04W 84/18; H04W 4/005; H04W 4/008; H04W 84/12; H04W 4/70; H04W 4/80;
(Continued)

(57) **ABSTRACT**

An electronic device includes a first processor; a first commissioning element to communicate with external devices using wireless communication and a first memory storing programs for commissioning a target device onto a communication network. The communication network has configuration data including a network identifier and a network key. The programs provide a user interface for commissioning the target device. The target device includes a second processor, a second wireless radio, and a second commissioning element to receive the network configuration data. The target device decodes the network configuration data to commission the target device onto the communication network. The electronic device sends the network configuration data to the target device and receives confirmation that the target device has successfully connected to the communication network. The electronic device displays the success in a user interface.

20 Claims, 6 Drawing Sheets

COMMISSIONED DEVICE



Related U.S. Application Data

- continuation of application No. 15/452,654, filed on Mar. 7, 2017, now Pat. No. 9,998,325, which is a continuation of application No. 15/159,735, filed on May 19, 2016, now Pat. No. 9,591,690, which is a continuation of application No. 14/927,406, filed on Oct. 29, 2015, now Pat. No. 9,485,790, which is a continuation of application No. 13/839,828, filed on Mar. 15, 2013, now Pat. No. 9,198,204.
- (60) Provisional application No. 61/622,620, filed on Apr. 11, 2012.
- (51) **Int. Cl.**
H04W 4/80 (2018.01)
H04W 4/70 (2018.01)
H04W 84/18 (2009.01)
H04B 10/116 (2013.01)
H04W 84/12 (2009.01)
- (52) **U.S. Cl.**
CPC **H04W 4/80** (2018.02); **H04W 76/10** (2018.02); **H04W 84/18** (2013.01); **H04W 84/12** (2013.01)
- (58) **Field of Classification Search**
CPC H04W 76/10; H04L 41/0803; H04L 41/0806; H04B 10/116
See application file for complete search history.
- (56) **References Cited**
U.S. PATENT DOCUMENTS
- | | | | |
|--------------|----|---------|------------------------|
| 5,838,226 | A | 11/1998 | Hougy |
| 6,597,396 | B1 | 7/2003 | Quendt |
| 7,260,597 | B1 | 8/2007 | Hofrichter et al. |
| 7,352,930 | B2 | 4/2008 | Lowles |
| 7,570,485 | B2 | 8/2009 | Krah |
| 7,752,329 | B1 | 7/2010 | Meenan |
| 7,830,258 | B2 | 11/2010 | McAllister |
| 7,953,327 | B2 | 5/2011 | Pereira et al. |
| 7,961,674 | B2 | 6/2011 | Jing et al. |
| 7,965,983 | B1 | 6/2011 | Swan et al. |
| 8,049,434 | B2 | 11/2011 | Crouse et al. |
| 8,096,695 | B2 | 1/2012 | Ong |
| 8,156,500 | B2 | 4/2012 | Helander |
| 8,161,420 | B2 | 4/2012 | Ding |
| 8,204,979 | B2 | 6/2012 | Vutharkar et al. |
| 8,228,198 | B2 | 7/2012 | McAllister |
| 8,239,928 | B2 | 8/2012 | Huang et al. |
| 8,265,674 | B2 | 9/2012 | Choong et al. |
| 8,279,158 | B2 | 10/2012 | Lowles et al. |
| 8,295,990 | B2 | 10/2012 | Venkatakrishnan et al. |
| 8,370,370 | B2 | 2/2013 | Huang et al. |
| 8,406,819 | B2 | 3/2013 | Steer et al. |
| 8,407,347 | B2 | 3/2013 | Zhang et al. |
| 8,409,001 | B2 | 4/2013 | Chang |
| 8,471,500 | B2 | 6/2013 | Fletcher et al. |
| 8,478,450 | B2 | 7/2013 | Lu et al. |
| 8,508,465 | B2 | 8/2013 | Broga et al. |
| 8,519,844 | B2 | 8/2013 | Richey et al. |
| 8,543,688 | B1 | 9/2013 | Ramamurthy |
| 8,576,276 | B2 | 11/2013 | Bar-zeev et al. |
| 8,577,378 | B2 | 11/2013 | Nagaraja et al. |
| 8,606,645 | B1 | 12/2013 | Applefeld |
| 8,613,070 | B1 | 12/2013 | Borzycki et al. |
| 8,653,760 | B1 | 2/2014 | Pearce et al. |
| 8,688,392 | B2 | 4/2014 | Tam et al. |
| 8,823,795 | B1 | 9/2014 | Scalisi et al. |
| 8,843,995 | B2 | 9/2014 | Buckley et al. |
| 8,942,694 | B2 | 1/2015 | Woo |
| 9,009,805 | B1 | 4/2015 | Kirkby et al. |
| 9,207,659 | B1 | 12/2015 | Sam I |
| 9,325,516 | B2 | 4/2016 | Pera et al. |
| 9,326,126 | B2 | 4/2016 | Yang |
| 9,401,901 | B2 | 7/2016 | Huang et al. |
| 9,412,266 | B2 | 8/2016 | Chen et al. |
| 9,419,871 | B2 | 8/2016 | Foley et al. |
| 9,462,624 | B2 | 10/2016 | Logue |
| 9,479,504 | B2 | 10/2016 | Bae et al. |
| 9,488,994 | B2 | 11/2016 | Zywicki et al. |
| 9,528,861 | B1 | 12/2016 | Haupt et al. |
| 9,547,980 | B2 | 1/2017 | Chen et al. |
| 9,554,061 | B1 | 1/2017 | Proctor, Jr. |
| 9,800,429 | B2 | 10/2017 | Crayford et al. |
| 9,948,685 | B2 | 4/2018 | Na et al. |
| 10,262,210 | B2 | 4/2019 | Kirkby et al. |
| 2001/0016492 | A1 | 8/2001 | Igarashi |
| 2002/0016639 | A1 | 2/2002 | Smith et al. |
| 2002/0178385 | A1 | 11/2002 | Dent et al. |
| 2003/0061284 | A1 | 3/2003 | Mandarino et al. |
| 2003/0067394 | A1 | 4/2003 | Tsui |
| 2003/0169728 | A1 | 9/2003 | Choi |
| 2004/0083393 | A1 | 4/2004 | Jordan et al. |
| 2004/0243257 | A1 | 12/2004 | Theimer |
| 2005/0041686 | A1 | 2/2005 | Roy et al. |
| 2005/0281277 | A1 | 12/2005 | Killian |
| 2006/0109988 | A1 | 5/2006 | Metcalf |
| 2006/0174102 | A1 | 8/2006 | Smith et al. |
| 2006/0259183 | A1 | 11/2006 | Hayes et al. |
| 2007/0014303 | A1 | 1/2007 | Schulz et al. |
| 2007/0105542 | A1 | 5/2007 | Friedman |
| 2007/0250592 | A1 | 10/2007 | Reckamp et al. |
| 2007/0294335 | A1 | 12/2007 | Gershon |
| 2008/0037444 | A1 | 2/2008 | Chhabra |
| 2008/0066093 | A1 | 3/2008 | Igoe et al. |
| 2008/0089300 | A1 | 4/2008 | Yee |
| 2008/0122606 | A1 | 5/2008 | Bradley |
| 2008/0168523 | A1 | 7/2008 | Ansari et al. |
| 2008/0219672 | A1 | 9/2008 | Tam et al. |
| 2008/0277486 | A1 | 11/2008 | Seem et al. |
| 2008/0278100 | A1 | 11/2008 | Hwang |
| 2009/0033485 | A1 | 2/2009 | Naeve et al. |
| 2009/0070681 | A1 | 3/2009 | Dawes et al. |
| 2009/0080896 | A1 | 3/2009 | Pereira et al. |
| 2009/0244097 | A1 | 10/2009 | Estevez |
| 2010/0068997 | A1 | 3/2010 | Dunko |
| 2010/0083356 | A1 | 4/2010 | Steckley et al. |
| 2010/0130166 | A1 | 5/2010 | Tsuria et al. |
| 2010/0138007 | A1 | 6/2010 | Clark et al. |
| 2010/0141153 | A1 | 6/2010 | Recker et al. |
| 2010/0150348 | A1 | 6/2010 | Fairbanks et al. |
| 2010/0189096 | A1 | 7/2010 | Flynn |
| 2010/0192212 | A1 | 7/2010 | Raleigh |
| 2010/0246825 | A1 | 9/2010 | Baras et al. |
| 2010/0248707 | A1 | 9/2010 | Hoffner et al. |
| 2010/0283579 | A1 | 11/2010 | Kraus et al. |
| 2010/0283584 | A1 | 11/2010 | McAllister |
| 2011/0046798 | A1 | 2/2011 | Imes et al. |
| 2011/0107364 | A1 | 5/2011 | Lajoie et al. |
| 2011/0121654 | A1 | 5/2011 | Recker et al. |
| 2011/0172844 | A1 | 7/2011 | Choong et al. |
| 2011/0199004 | A1 | 8/2011 | Henig et al. |
| 2011/0202151 | A1 | 8/2011 | Covaro et al. |
| 2011/0225373 | A1 | 9/2011 | Ito et al. |
| 2011/0299412 | A1 | 12/2011 | Diab |
| 2011/0302634 | A1 | 12/2011 | Karaoguz |
| 2012/0011567 | A1 | 1/2012 | Cronk et al. |
| 2012/0045060 | A1 | 2/2012 | Maestrini et al. |
| 2012/0049765 | A1 | 3/2012 | Lu et al. |
| 2012/0082062 | A1 | 4/2012 | McCormack |
| 2012/0144469 | A1 | 6/2012 | Ainslie et al. |
| 2012/0167063 | A1 | 6/2012 | Detwiler et al. |
| 2012/0204243 | A1 | 8/2012 | Wynn et al. |
| 2012/0216296 | A1 | 8/2012 | Kidron |
| 2012/0239936 | A1 | 9/2012 | Holtmanns et al. |
| 2013/0026947 | A1 | 1/2013 | Economy et al. |
| 2013/0041516 | A1 | 2/2013 | Rockenfeller et al. |
| 2013/0064132 | A1 | 3/2013 | Low et al. |
| 2013/0073705 | A1 | 3/2013 | Hester |
| 2013/0076491 | A1 | 3/2013 | Brandtsma et al. |
| 2013/0086665 | A1 | 4/2013 | Filippi et al. |
| 2013/0124855 | A1 | 5/2013 | Varadarajan et al. |

(56)

References Cited**U.S. PATENT DOCUMENTS**

2013/0132008	A1	5/2013	Borean
2013/0191755	A1	7/2013	Balog et al.
2013/0198786	A1	8/2013	Cook et al.
2013/0223279	A1 *	8/2013	Tinnakornrisuphap
H04W 12/04			
370/254			
2013/0227656	A1	8/2013	Holtmanns et al.
2013/0236183	A1	9/2013	Chao et al.
2013/0268357	A1	10/2013	Heath
2013/0276140	A1	10/2013	Coffing et al.
2013/0340050	A1	12/2013	Harrison
2014/0007222	A1	1/2014	Qureshi et al.
2014/0068705	A1	3/2014	Chambers et al.
2014/0068789	A1	3/2014	Watts et al.
2014/0099933	A1	4/2014	Yerrabommanahalli et al.
2014/0129006	A1	5/2014	Chen et al.
2014/0137188	A1	5/2014	Bartholomay et al.
2014/0157370	A1	6/2014	Plattner et al.
2014/0164758	A1	6/2014	Ramamurthy
2014/0164776	A1	6/2014	Hook et al.
2014/0173692	A1	6/2014	Srinivasan et al.
2014/0189359	A1	7/2014	Marien et al.
2014/0189808	A1	7/2014	Mahaffey et al.
2014/0245411	A1	8/2014	Meng et al.
2014/0245461	A1	8/2014	O'Neill et al.
2014/0248852	A1	9/2014	Raleigh et al.
2014/0266600	A1	9/2014	Alberth, Jr. et al.
2014/0270801	A1	9/2014	Sleator
2014/0273963	A1	9/2014	Su et al.
2014/0281497	A1	9/2014	Medvinsky et al.
2014/0282570	A1	9/2014	Prasanna
2014/0282877	A1	9/2014	Mahaffey et al.
2014/0310509	A1	10/2014	Potlapally et al.
2014/0310510	A1	10/2014	Potlapally et al.
2015/0015369	A1	1/2015	Lamb
2015/0043377	A1	2/2015	Cholas et al.
2015/0070585	A1	3/2015	Sharif-Ahmadi
2015/0071052	A1	3/2015	Hershberg et al.
2015/0126153	A1	5/2015	Spitz et al.
2015/0160634	A1	6/2015	Smith et al.
2015/0177292	A1	6/2015	Silveira Filho et al.
2015/0195100	A1	7/2015	Imes et al.
2015/0215297	A1	7/2015	Rathod et al.
2015/0220993	A1	8/2015	Bente
2015/0249855	A1	9/2015	Dewa et al.
2015/0282216	A1	10/2015	Reshef et al.
2016/0044032	A1	2/2016	Kim et al.
2016/0089457	A1	3/2016	Liao et al.
2016/0091217	A1	3/2016	Verberkt
2016/0132031	A1	5/2016	Kozura et al.
2016/0142263	A1	5/2016	Erdmann et al.
2016/0191264	A1	6/2016	Kim et al.
2016/0248629	A1	8/2016	Erdmann
2016/0370208	A1	12/2016	Patel et al.
2016/0380945	A1	12/2016	Wood et al.

2016/0381500	A1	12/2016	Larson
2017/0223808	A1	8/2017	Barna
2017/0285893	A1	10/2017	Shim et al.
2018/0077778	A1	3/2018	Vangeel

OTHER PUBLICATIONS

Detailed Technical Specification of Security for Heterogeneous Access, May 31, 2002, 161 pgs, www.isrc.rhul.ac.uk/shaman/docs/d09v1.pdf.

Google Inc., International Search Report and Written Opinion, PCT/US2015/053291, dated Feb. 5, 2016, 18 pgs.

Google Inc., International Search Report and Written Opinion, PCT/US2015/060405, dated Feb. 25, 2016, 9 pgs.

Goadrich, Mark H., and Michael P. Rogers. "Smart smartphone development: iOS versus Android." In Proceedings of the 42nd ACM technical symposium on Computer science education, pp. 607-612. ACM, 2011.

Manashty, Ali Reza, Amir Rajabzadeh, and Zahra Forootan Jahromi. "A Scenario-Based Mobile Application for Robot-Assisted Smart Digital Homes." arXiv preprint arXiv:1009.5398 (2010).

Armac, Ibrahim, and Daniel Retkowitz. "Simulation of smart environments." In IEEE International Conference on Pervasive Services, pp. 257-266. IEEE, 2007.

Ramlee, Ridza Azri, Man Hong Leong, Ranjit Singh A. Sarban Singh, Mohd Muzafar Ismail, Mohd Azlishah Othman, Hamzah Asyran Sulaiman, Mohamad Harris Misran, Meor Said, and Maizatul Alice. "Bluetooth remote home automation system using android application." (2013): 1-5.

Van Nguyen, Tarn, Jin Gook Kim, and Deokjai Choi. "ISS: the interactive smart home simulator." In Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on, vol. 3, pp. 1828-1833. IEEE, 2009.

Rajabzadeh, Amir, Ali Reza Manashty, and Zahra Forootan Jahromi. "A Mobile Application for Smart House Remote Control System." arXiv preprint arXiv:1009.5557 (2010).

Gavalas, Damianos, and Daphne Economou. "Development platforms for mobile applications: Status and trends." IEEE software 28, No. 1 (2011): 77-86.

Y. Zatout, "Using wireless technologies for healthcare monitoring at home: A survey," 2012 IEEE 14th International Conference on e-Health Networking, Applications and Services (Healthcom), Beijing, 2012, pp. 383-386. URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6379443&isnumber=6379371>.

"INSTEON Compared" www.insteon.com/pdf/insteoncompared.pdf version.2 2013.

"Thread Group Information Event" <https://www.threadgroup.org/Portals/O/documents/events/ThreadIntro.pdf>; Sep 30, 2014.

Google Inc., International Preliminary Report on Patentability, PCT/US2015/060405, dated May 16, 2017, 7 pgs.

Google Inc., International Preliminary Report on Patentability, PCT/US2015/053291, dated Apr. 4, 2017, 14 pgs.

* cited by examiner

FIG. 1 (Prior Art)

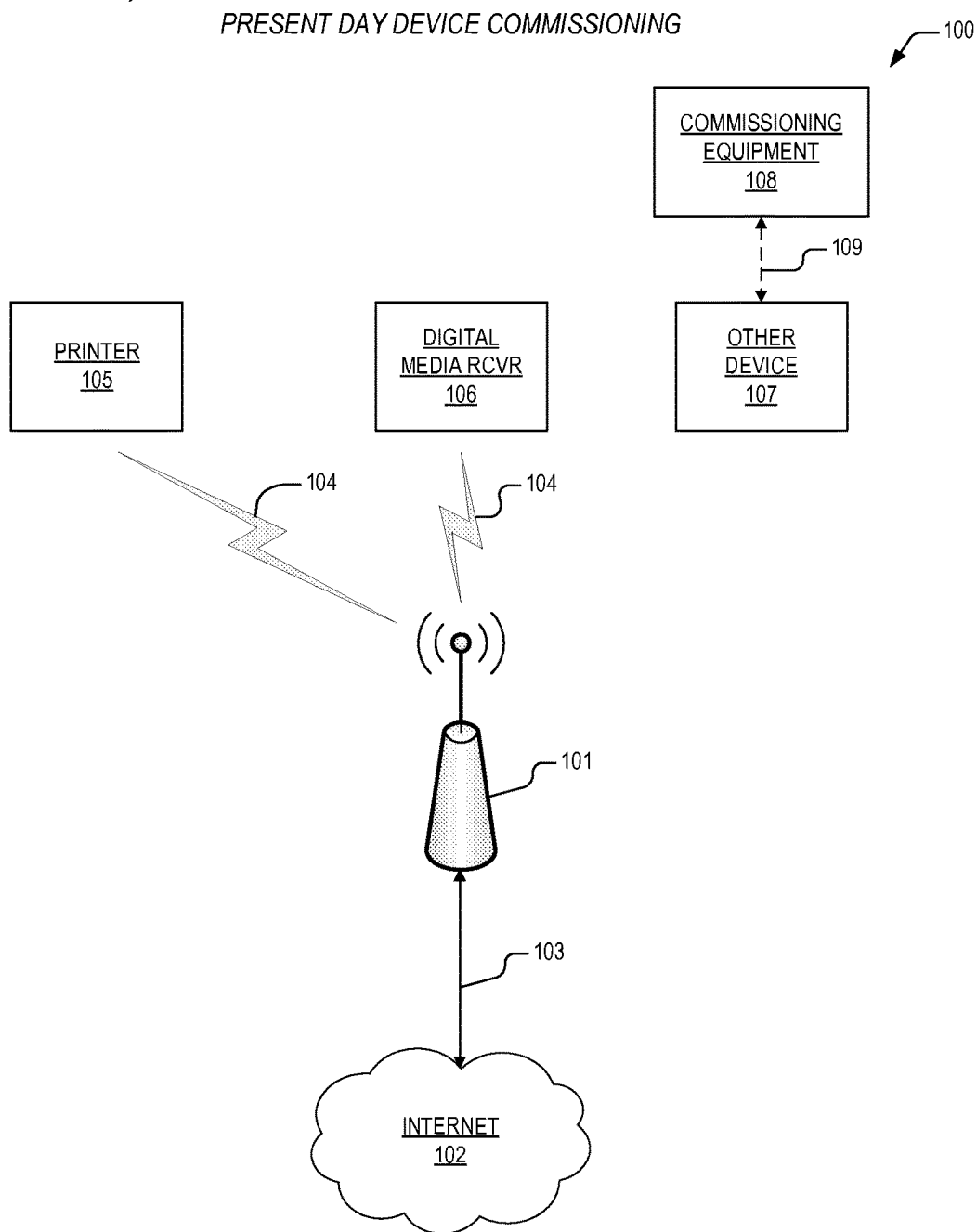


FIG. 2 (Prior Art)

PRESENT DAY DEVICE COMMISSIONING MECHANISMS

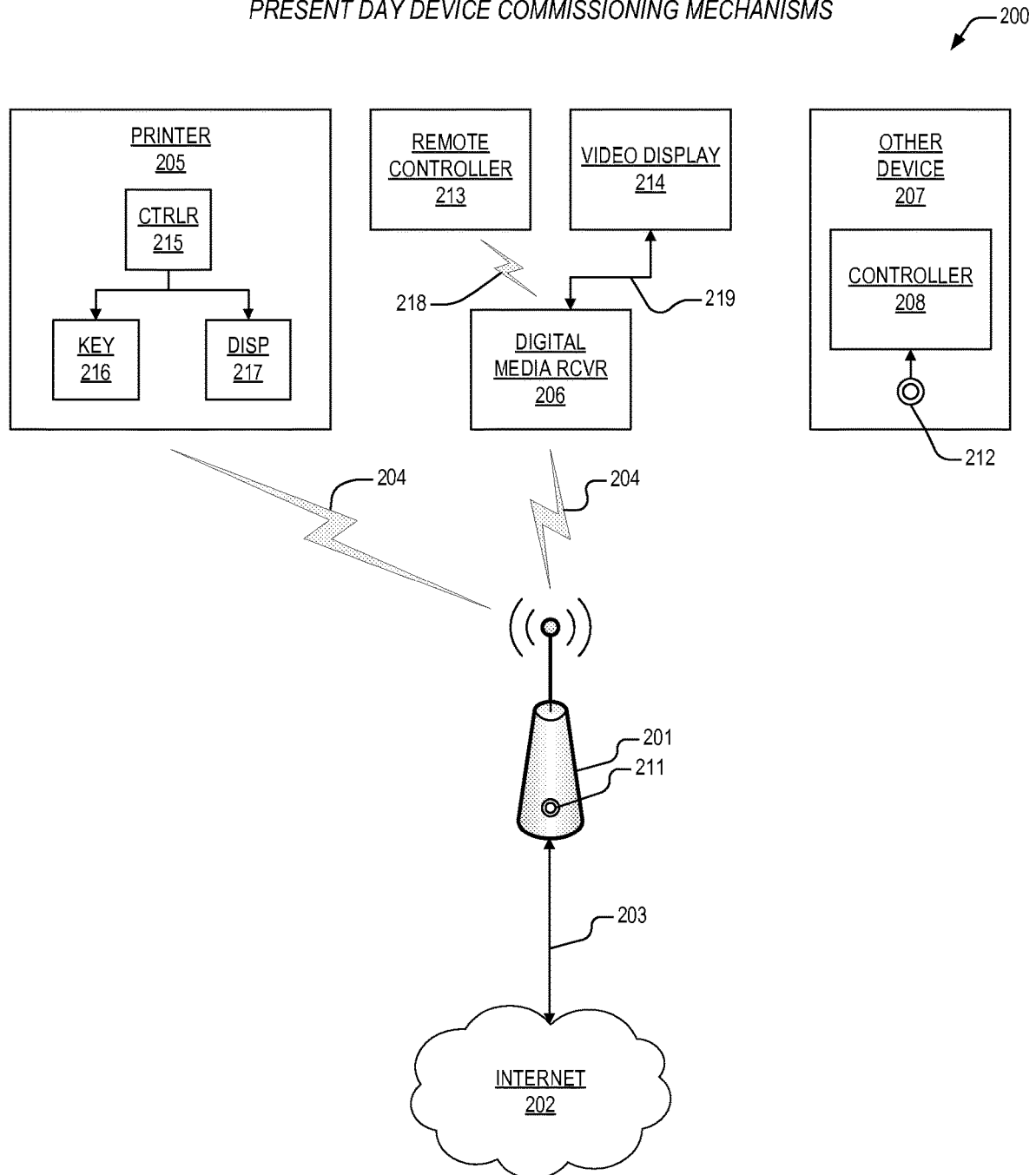


FIG. 3

EXEMPLARY SEAMLESS DEVICE COMMISSIONING TECHNIQUE

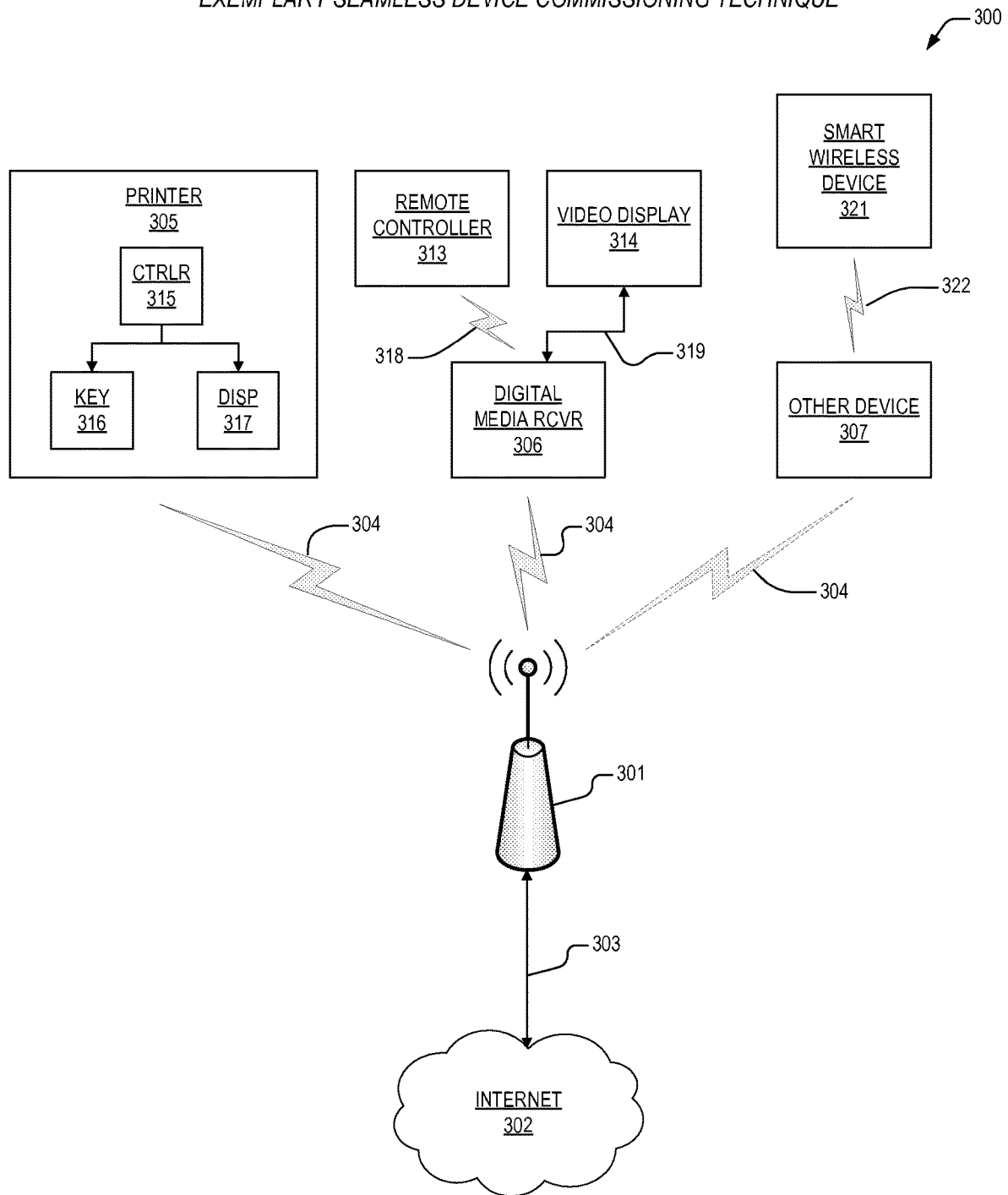


FIG. 4

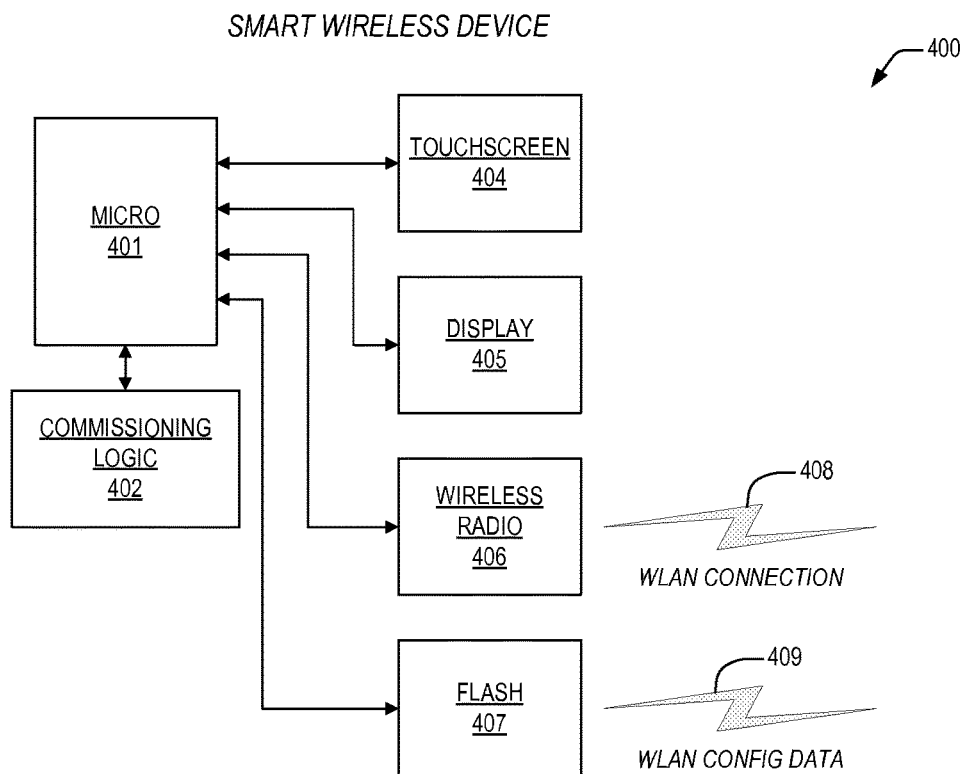


FIG. 5

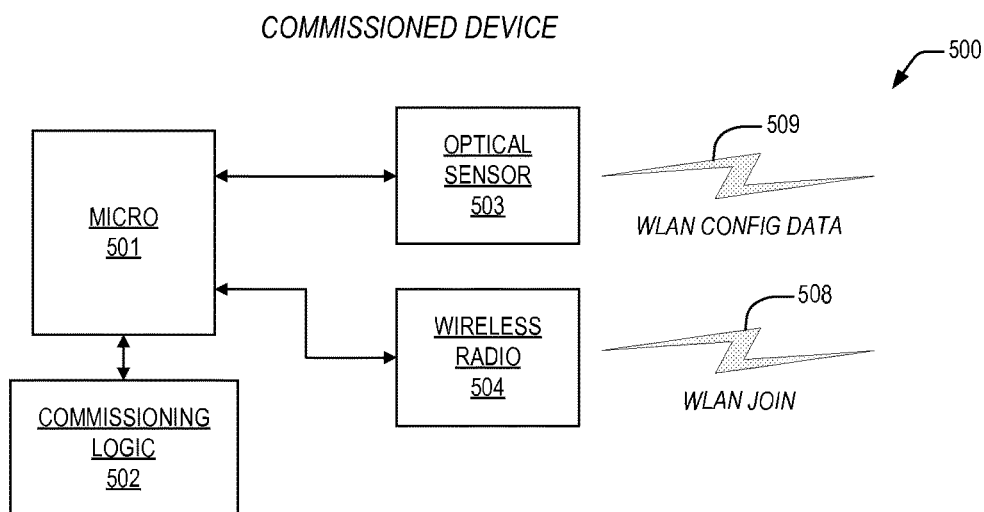


FIG. 6

EXEMPLARY COMMISSIONING DEVICE

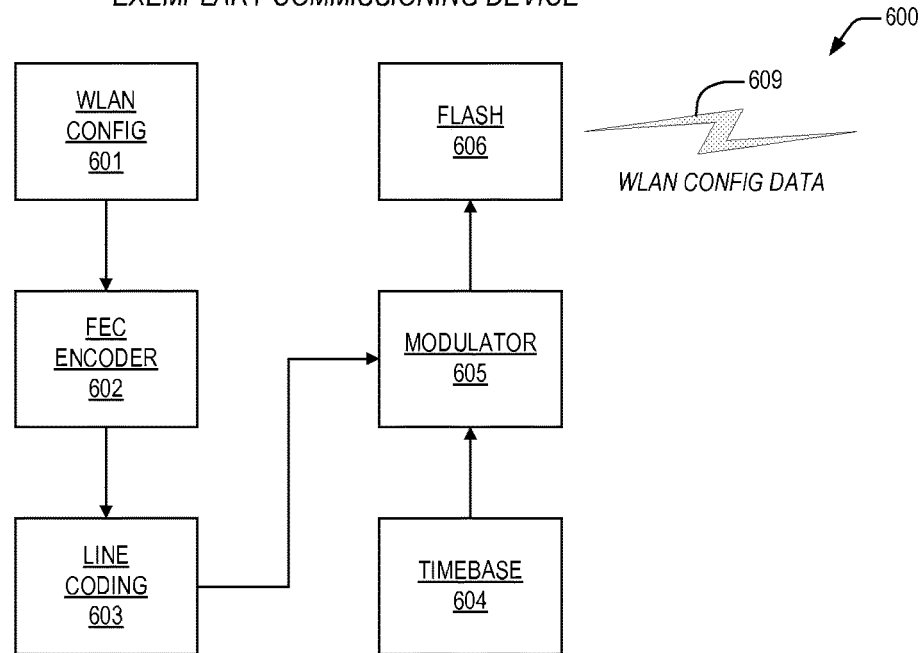


FIG. 7

EXEMPLARY COMMISSIONED DEVICE

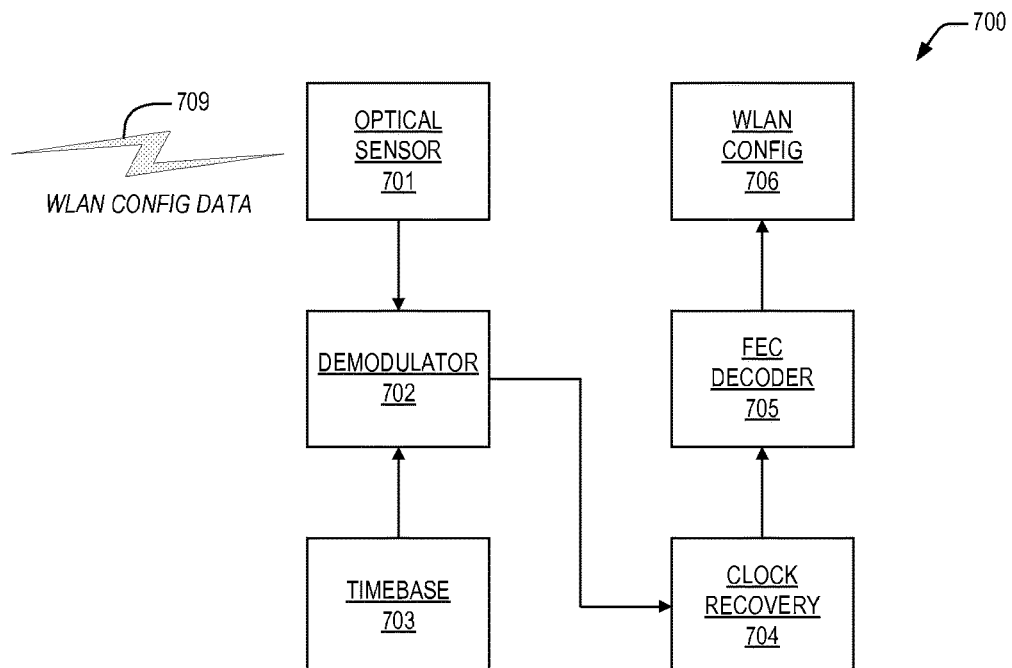
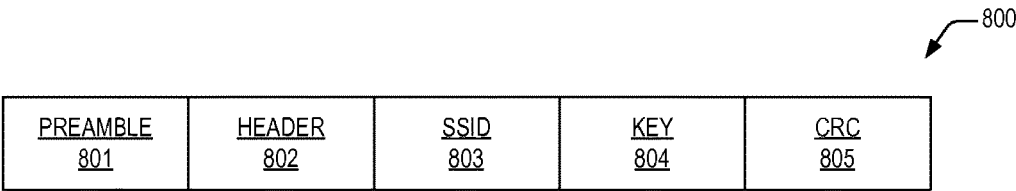


FIG. 8

EXEMPLARY COMMISSIONED DEVICE



1

APPARATUS AND METHOD FOR SEAMLESS COMMISSIONING OF WIRELESS DEVICES

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation of and claims priority to U.S. Utility patent application Ser. No. 15/960,337, filed Apr. 23, 2018, titled "Apparatus and Method for Seamless Commissioning of Wireless Devices," which is a continuation of and claims priority to U.S. Utility patent application Ser. No. 15/452,654, filed Mar. 7, 2017, issued as U.S. Pat. No. 9,998,325 on Jun. 12, 2018, titled "Apparatus and Method for Seamless Commissioning of Wireless Devices," which is a continuation of and claims priority to U.S. Utility patent application Ser. No. 15/159,735, filed May 19, 2016, issued as U.S. Pat. No. 9,591,690 on Mar. 7, 2017, titled "Apparatus and Method for Seamless Commissioning of Wireless Devices," which is a continuation of and claims priority to U.S. Utility patent application Ser. No. 14/927,406, filed Oct. 29, 2015, issued as U.S. Pat. No. 9,485,790 on Nov. 1, 2016, titled "Apparatus and Method for Seamless Commissioning of Wireless Devices," which is a continuation of and claims priority to U.S. Utility patent application Ser. No. 13/839,828, filed Mar. 15, 2013, issued as U.S. Pat. No. 9,198,204 on Nov. 24, 2015, titled "Apparatus and Method for Seamless Commissioning of Wireless Devices," which claims priority to and the benefit of U.S. Provisional Application No. 61/622,620, filed on Apr. 11, 2012, titled "Apparatus and Method for Seamless Commissioning of Wireless Devices." The content of each of the above applications is herein incorporated by reference in its entirety.

TECHNICAL FIELD

This invention relates in general to the field of electronic device commissioning, and more particularly to an apparatus and method for seamless commissioning of electronic devices onto wireless local area networks (WLANs).

BACKGROUND

It is almost incredible to realize the number of electronic devices that exist today, and even more so when one considers the exponentially increasing number of those devices that are configured for wireless network integration. On the business front, laptop computers, smart phones, and tablet computers abound. On the personal front, home automation is crossing thresholds which have heretofore been untouched. Today's average consumer has access to the Internet via a high speed connection and most probably has a wireless network in their home. Attached to that wireless network are home computers, digital readers, wireless printers, and some form of wireless entertainment equipment, such as streaming media players and the like.

On top of this, everyday appliances such as door locks, thermostats, garage door openers, outlets, and light switches are now being configured for wireless operation. Both businesses and homes frequently are equipped with alarm systems that function wirelessly. The list continues to grow.

But as one skilled in the art will appreciate, wireless electronic devices do not merely connect themselves to the nearest wireless network. Rather, they must be "commissioned" onto a desired network. That is, the particular devices must be provided with configuration data to enable them to join the desired wireless network so that they can

2

function according to specification by utilizing the communication properties of the network.

Yet, at this point in time, virtually every electronic device is different in the manner in which they are commissioned onto a wireless network. Some devices require a sequence of switch actuations that are coordinated with similar actuations on a controller device. Other devices have rudimentary keypads whereby a user enters the configuration information, often in cryptic and unintelligible form. Still other devices have keypads and displays that allow for data entry that is somewhat more user friendly. Yet more devices utilize other coupled devices (e.g., televisions and special remote controls) to enter network names and passwords. And as a result, the average user of these wireless devices must either be very smart, or they require assistance from someone who is knowledgeable.

Accordingly, what is needed is a technique for commissioning a wireless device onto a wireless network that the average consumer can employ without undue distress.

In addition, what is needed is a mechanism for commissioning wireless devices that utilizes common, user friendly steps and equipment.

SUMMARY

The present invention, among other applications, is directed to solving the above-noted problems and addresses other problems, disadvantages, and limitations of the prior art. The present invention provides a superior technique for commissioning one or more electronic devices onto an existing wireless local area network. In one embodiment an apparatus is provided for commissioning a target device onto a wireless local area network (WLAN). The apparatus includes a smart wireless device. The smart wireless device has commissioning logic and a transducer. The commissioning logic is configured to format and direct transmission of one or more WLAN configuration packets that convey WLAN configuration data. The transducer is coupled to the commissioning logic, and is configured to transmit the one or more WLAN configuration packets over the transmission path to the target device, wherein the transducer is an existing component of the smart wireless device.

One aspect of the present invention contemplates an apparatus for wireless local area network (WLAN) device commissioning. The apparatus has a smart wireless device and a target device. The smart wireless device includes first commissioning logic and a first transducer. The first commissioning logic is configured to format and direct transmission of one or more WLAN configuration packets. The first transducer is coupled to the first commissioning logic, and is configured to transmit the one or more WLAN configuration packets over the transmission path, where the first transducer is an existing component of the smart wireless device. The target device includes a second transducer, second commissioning logic, and a wireless radio. The second transducer is configured to receive the one or more WLAN configuration packets. The second commissioning logic is coupled to the second transducer, and is configured to decode and process the one or more WLAN configuration packets to recover WLAN configuration data. The wireless radio is operatively coupled to the commissioning logic, and is configured to employ the WLAN configuration data to enable the target device to join a WLAN.

Another aspect of the present invention comprehends a method for wireless local area network (WLAN) device commissioning. The method includes: executing a first

3

application on a smart wireless device to format and direct transmission of one or more WLAN packets; via a first transducer of the smart wireless device, transmitting the one or more WLAN configuration packets over a transmission path, where the first transducer is an existing component of the smart wireless device; via a second transducer disposed within a target device, receiving the one or more WLAN configuration packets; executing a second application on the target device to decode and process the one or more WLAN configuration packets to recover WLAN configuration data; and via a wireless radio disposed within the target device, employing the WLAN configuration data to enable the target device to join a WLAN.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects, features, and advantages of the present invention will become better understood with regard to the following description, and accompanying drawings where:

FIG. 1 is a block diagram illustrating a present day device commissioning technique;

FIG. 2 is a block diagram depicting present day commissioning mechanisms such as might be employed in the technique of FIG. 1;

FIG. 3 is a block diagram featuring an exemplary seamless device commissioning technique according to the present invention;

FIG. 4 is a block diagram showing a smart wireless commissioning device according to the present invention;

FIG. 5 is a block diagram illustrating a commissioned device according to the present invention;

FIG. 6 is a block diagram detailing features of an exemplary commissioning device according to the present invention;

FIG. 7 is a block diagram showing details of an exemplary commissioned device according to the present invention; and

FIG. 8 is a block diagram illustrating an exemplary commissioning packet according to the present invention.

DETAILED DESCRIPTION

Exemplary and illustrative embodiments of the invention are described below. In the interest of clarity, not all features of an actual implementation are described in this specification, for those skilled in the art will appreciate that in the development of any such actual embodiment, numerous implementation specific decisions are made to achieve specific goals, such as compliance with system-related and business related constraints, which vary from one implementation to another. Furthermore, it will be appreciated that such a development effort might be complex and time-consuming, but would nevertheless be a routine undertaking for those of ordinary skill in the art having the benefit of this disclosure. Various modifications to the preferred embodiment will be apparent to those skilled in the art, and the general principles defined herein may be applied to other embodiments. Therefore, the present invention is not intended to be limited to the particular embodiments shown and described herein, but is to be accorded the widest scope consistent with the principles and novel features herein disclosed.

The present invention will now be described with reference to the attached figures. Various structures, systems, and devices are schematically depicted in the drawings for purposes of explanation only and so as to not obscure the

4

present invention with details that are well known to those skilled in the art. Nevertheless, the attached drawings are included to describe and explain illustrative examples of the present invention. The words and phrases used herein should be understood and interpreted to have a meaning consistent with the understanding of those words and phrases by those skilled in the relevant art. No special definition of a term or phrase (i.e., a definition that is different from the ordinary and customary meaning as understood by those skilled in the art) is intended to be implied by consistent usage of the term or phrase herein. To the extent that a term or phrase is intended to have a special meaning (i.e., a meaning other than that understood by skilled artisans) such a special definition will be expressly set forth in the specification in a definitional manner that directly and unequivocally provides the special definition for the term or phrase.

In view of the above background discussion on device commissioning and associated techniques employed within present day systems for commissioning a wireless device into a wireless network, a discussion of the limitations of these present day mechanisms will now be discussed with reference to FIGS. 1-2. Following this, a discussion of the present invention will be presented with reference to FIGS. 3-8. The present invention overcomes limitations of present day device commissioning techniques by providing apparatus and methods that enable devices to be commissioned into a wireless network without requiring special purpose commissioning hardware, administrative access to network equipment, the creation of ad hoc networks, or compromises to operational security.

Turning to FIG. 1, a block diagram 100 is presented illustrating a present day device commissioning technique. The diagram 100 depicts a wireless local area network (WLAN) access point 101 that controls access to and communications over a WLAN 104. As one skilled in the art will appreciate, the access point 101 may also be referred to as a Wi-Fi access point 101, a Wi-Fi router 101, a Wi-Fi gateway 101, a Wi-Fi hub 101, and the like which refer to a controller 101 that allows electronic devices 105-107 to exchange data wirelessly over a wireless network 104. Often, the network 104 utilizes protocols based on the Institute of Electrical and Electronics Engineers' (IEEE) 802.11 standards, however, such references are not intended to limit protocols of the network 104 to 802.11, but rather reference is made to the well known 802.11 standards in order to teach aspects of the present invention.

The WLAN access point 101 is coupled via bus 103 to the Internet in any of number of well known ways to include digital subscriber line (DSL), cable, and wireless radio link. As such, the access point 101 enables the devices 105-107 to communicate with each other, and also to other devices (not shown) over the Internet 102.

The diagram 100 shows a printer 105 and a digital media receiver 106 (e.g., APPLETV®, ROKU®, DIRECTV®) that are coupled to the access point 101 via the WLAN 104. The diagram additionally shows an other device 107 that is not yet coupled to the access point 101 via the WLAN 104, and which requires commissioning into the WLAN 104. As one skilled in the art will appreciate, the other device 107 may comprise, but is not limited to a personal computer, a video game console, a smartphone, a tablet computer, an alarm console, a streaming media player, network attached storage, and any number of other Wi-Fi enabled appliances (e.g., outlets, lights, locks, sensors, etc.). Although reference is made to "Wi-Fi enabled appliances," the present inventors noted that applications of the present invention extend beyond commonly construed "Wi-Fi" protocols (i.e., IEEE

5

802.11 protocols) to comprehend any wireless protocol, as will be discussed in further detail below. For purposes of the present disclosure, “commissioning” is construed to mean configuration of the other device **107** and the system **100** as a whole such that the other device **107** is enabled to function properly over the WLAN **104** including, if required, access to the Internet **102**.

As one skilled in the art will appreciate, there are a number of varying techniques that may be employed to commission present day devices onto a WLAN **104**. Some of these techniques require operator intervention, some require the use of special purpose commissioning equipment **108**, and some require the creation of temporary (or ad hoc) wireless networks **109**. And the present inventors have noted that as wireless device usage proliferates in the office and the home, this cornucopia of commissioning techniques poses challenges to those in the art, because each individual technique is limited with regard to one system attribute or another. A more detailed discussion of the limitations associated with present day commissioning techniques will now be presented with reference to FIG. **2**.

FIG. **2** is a block diagram **200** depicting present day commissioning mechanisms such as might be employed in the technique of FIG. **1**. The diagram **200** depicts a WLAN access point **201** that is coupled to the Internet **202** via a bus **203**, and which controls access to a wireless LAN **204**. Operationally, the access point **201** functions substantially like the access point **101** of FIG. **1**. A printer **205** is coupled to the access point **201** via the WLAN **204**. The printer **205** has a controller **215** that is coupled to a keyboard **216** and display **217**. The diagram **200** also shows a digital media receiver **206** that is coupled to the access point **201** via the WLAN **204**. The receiver **206** is also coupled to a remote controller **213** via a radio wave bus **218**, and to a video display **214** (e.g., a television) via bus **219**. The diagram **200** additionally depicts an other device **207** that requires commissioning onto the WLAN **204**. The other device **207** includes a controller **208** that is coupled to a pushbutton switch **212**.

The particular elements shown in FIG. **2** are well known to those in the art and are selected to illustrate limitations associated with some of the most prevalently employed present day commissioning techniques. For example, one well known commissioning technique utilizes coordinated activation of pushbutton switches to initiate the exchange of configuration data so that, say, the other device **207** can join the WLAN **204**. Accordingly, a commissioning pushbutton switch **211** is shown on the access point **201**, which functions in this scenario as a system controller for the other device **207**. Thus, a technician is required to perform a time synchronized actuation of the pushbutton switches **211**, **212** on both the access point and the other device **207** and by doing so, the access point **201** begins transmission of packets that include configuration data, thus enabling the other device **207** to subsequently join the WLAN **204**. Transmission of this configuration data is typically accomplished utilizing the same radio transceivers (not shown) as are required for interoperability over the WLAN **204** itself. But the present inventors have observed that this “pushbutton” commissioning technique is disadvantageous because physical and/or administrative access to the access point **201** is required in order to accomplish commissioning of the other device **207**, and in office and other scenarios where security concerns are paramount, such access may not be easily obtained.

Another well known present day commissioning technique provides for the direct entry of device configuration

6

data utilizing integral or directly coupled controls and displays. For instance, consider a scenario under which an operator may commission the printer **205** onto the WLAN **204** by performing a series of data entry steps via the keypad **216** and display **217**. The operator may select from a menu of WLAN types and then the printer **205** will monitor for available networks of that type, thus allowing the operator to select from a menu of available networks. The operator may then be required to enter secure access information (e.g., passwords, etc.) to enable the printer to join the WLAN.

Or consider a more intelligent commissioning technique where, say, the digital media receiver **206** possesses the capabilities to determine the appropriate WLAN **204** to join, and all that is needed is for the operator to enter a personal identification number (PIN) via the remote controller **213** and verify configuration setup via the video display **214**.

The above commissioning techniques may be categorized in general as Wi-Fi protected setup techniques, and are well known to those in the art. Other techniques also include the use of near-field communications and the use of a direct connection such as Universal Serial Bus (USB). In more extreme cases, a device to be commissioned (e.g., the other device **207**), may create a temporary network and require the operator to configure a computing device (not shown) to join the temporary network. Then, the computing device is subsequently used to issue commands and configuration data to the devices, thus allowing it to join the intended WLAN **204**.

Alternatively, the device to be commissioned may be required to monitor for specially formatted probes or beacon request messages in order to decode network configuration data therefrom. Such a scenario could be employed utilizing the pushbutton method discussed above.

The present inventors note that, as one skilled in the art will appreciate, virtually all of the above noted commissioning techniques require physical and/or administrative access to either the access point **201** or to special commissioning equipment which, as is alluded to above, may not be easily obtained. It is also noted that commissioning techniques that involve the use of PINS and/or the direct entry of configuration data present the opportunity for tampering and or breach of security through the compromise of trusted data. It is furthermore noted that the creation of near-field communications and ad hoc networks requires dedicated equipment for such purposes. And moreover, all of the techniques discussed above present challenges to those other than a skilled technician. A typical operator may not be skilled in the entry of network configuration data or configuration of special commissioning equipment. In addition, for those techniques that temporarily open a network in order to commission as device, such an action is both observable and sniffable, thus providing a pathway for exploitation of the compromised parameter that allow unauthorized users and devices to join the network at a later time. Consequently, the wide range of complex commissioning techniques are challenging those in the art to develop more user-friendly, secure, and effective mechanisms for device commissioning than have heretofore been provided. More specifically, with the exponential increase in the deployment of wireless devices in the marketplace, those in the art are motivated to provide commissioning techniques that a typical user can perform without problems.

The present invention overcomes the above noted limitations, and others, by providing an apparatus and method for seamless device commissioning which, in one embodiment, leverages the ubiquitous nature and use of so-called

“smart” wireless devices in the home and office. The present invention will now be discussed with reference to FIGS. 3-8.

Turning to FIG. 3, a block diagram 300 is presented featuring an exemplary seamless device commissioning technique according to the present invention. The diagram. The diagram 300 depicts a wireless local area network (WLAN) access point 301 that controls access to and communications over a WLAN 304. Like the access point 101 of FIG. 1, the access point 301 according to the present invention allows electronic devices 305-307 to exchange data wirelessly over the wireless network 304 once the devices 305-307 are commissioned onto the WLAN 304. In one embodiment, the network 304 utilizes protocols based on the Institute of Electrical and Electronics Engineers’ (IEEE) 802.11 standards, however, such references are not intended to limit protocols of the network 304 to 802.11, but rather reference is made to the well known 802.11 standards in order to teach aspects of the present invention.

The WLAN access point 301 is coupled via bus 303 to the Internet in any of a number of well-known ways to include digital subscriber line (DSL), cable, and wireless radio link. As such, the access point 301 enables the devices 305-307 to communicate with each other, and also to other devices (not shown) over the Internet 302.

The diagram 300 shows a printer 305 and a digital media receiver 306 (e.g., APPLETV®, ROKU®, DIRECTV®) that are coupled to the access point 301 via the WLAN 304. The diagram additionally shows another device 307 that is not yet coupled to the access point 301 via the WLAN 304, and which requires commissioning into the WLAN 104. As one skilled in the art will appreciate, the other device 307 may comprise, but is not limited to a personal computer, a video game console, a smartphone, a tablet computer, an alarm console, a streaming media player, network attached storage, and any number of other Wi-Fi enabled appliances (e.g., outlets, lights, locks, sensors, etc.).

Like the printer 205 of FIG. 2, the printer 305 of FIG. 3 includes a controller 315 that is coupled to a keyboard 316 and a display 317, which are employed according to one of the present day methods described above to commission the printer 305 onto the WLAN 304. And like the receiver 206 of FIG. 2, the receiver 306 of FIG. 3 is coupled to a remote controller 313 via bus 318 and a video display 314 via bus 319, which are employed according to one of the present day methods described above to commission the receiver 306 onto the WLAN 304.

However, in contrast to present day commissioning techniques, the technique according to the present invention utilizes a smart wireless device 321 to commission the other device 307 according to the present invention onto the WLAN 304 by utilizing existing mechanisms (i.e., transmitting elements (not shown)) on the smart wireless device 321 to transmit configuration information over a transmission path 322 to the other device 307, thus enabling the other device 307 to join the WLAN. The other device 307 is configured to include receiving elements (not shown) that correspond to the transmitting elements and transmission path 322, thus enabling the other device 307 to receive and process the configuration information. It is a feature of the present invention to employ a smart wireless device 321 and to utilize existing elements within the smart wireless device 321 to transmit the configuration information to the other device such as an internal microcomputer, memory, a camera flash, backlit display, touchscreen or keyboard, and the like. Advantageously then, commissioning can be affected without a requirement for special commissioning equipment or special user skills.

In one embodiment, the smart wireless device 321 comprises a smart cellular telephone such as, but not limited to, an IPHONE® or ANDROID® phone. In another embodiment, the smart wireless device 321 comprises a tablet computer such as, but not limited to, an IPAD® or an ANDROID® tablet. Other embodiments of the smart wireless device 321 are contemplated which are capable of providing the functions and mechanisms discussed below.

In one embodiment, the smart wireless device 321 utilizes a side band, proximity based, communication channel 322 that is established between the smart wireless device 321 and the other device 307 to be commissioned onto the WLAN 304. The transmitting and receiving elements perform the functions and operations as discussed above. The transmitting and receiving elements comprise logic, circuits, devices, or program instructions executed by a microprocessor or other processing unit, or a combination of logic, circuits, devices, or program instructions executing on a microprocessor or other processing unit, or equivalent elements that are employed to execute the functions and operations according to the present invention as disclosed herein. The elements employed to accomplish these operations and functions within the transmitting and receiving elements may be shared with other circuits, program instructions, etc., that are employed to perform other functions and/or operations within the transmitting and receiving elements. In one embodiment, the smart device 321 constructs a commissioning message that includes WLAN configuration data. The message is transmitted by the smart device 321 to the other device 307 via the transmitting elements and the message is received by the other device 307 via the receiving elements. In one embodiment, the WLAN configuration data comprises the name of the WLAN, security mode, and any required security passphrase. Other embodiments contemplate additional or different configuration data as is appropriate for commissioning the other device 307.

In operation, the other device 307 extracts the WLAN configuration data from the received configuration message and employs the configuration data in order to join the WLAN 304.

In one embodiment, the transmitting elements comprise an existing camera flash and the receiving elements comprise an optical sensor. Another embodiment contemplates utilization of a display backlight that transmits optical signals to an optical sensor. A further embodiment comprehends the use of infrared transmission and receiving elements. Still another embodiment utilizes an acoustic speaker and microphone to transfer the configuration data via sound. Yet another embodiment utilizes BLUETOOTH® or substantially similar radio transmissions to transfer the configuration data. Further embodiments may use radio communications (NFC) or mechanical vibration/accelerometer transmission.

Alternative embodiments of the transmitting elements may use disparate radio frequencies such as, but not limited to, 433 Megahertz (MHz), 868 MHz, 900 MHz, 2.4 Gigahertz (GHz), or 5.2 GHz. These embodiments may use radio modulation techniques such as, but not limited to, frequency shift keying (FSK), phase shift keying (PSK), on-off keying (OOK), and quadrature amplitude modulation (QAM). These embodiments may use radio frequency (RF) communication standards such as, but not limited to, near field communication (NFC), RF identification (ID), BLUETOOTH LOW ENERGY®, Zigbee®, radio frequency for consumer electronics (RF4CE®), Z-Wave®, or other proprietary standards.

Referring now to FIG. 4, a block diagram is presented showing a smart wireless commissioning device **400** according to the present invention. The smart device **400** is configured to format and transmit WLAN configuration data to a corresponding device to be commissioned according to the present invention by utilizing an optical transmission path. The smart device **400** comprises a microcomputer **401** (e.g., microprocessor, central processing unit, processor, microcontroller, and etc.) that is coupled to commissioning logic **402**. The microcomputer is also coupled to an existing touchscreen **404**, a display **405**, a wireless radio **406**, and a camera flash **407**. In one embodiment, the camera flash **407** is employed as a transmitter to convey configuration data to the device to be commissioned.

Operationally, the microcontroller **401** includes processing circuits that are employed to format and configure WLAN configuration packets which are optically transmitted by the camera flash **407** by modulating the flash **407** on and off to transmit the packets according to any well known communication protocol or via a proprietary protocol. The commissioning logic **402** includes program instructions that direct the microcontroller **401** to format and transmit the WLAN configuration packets to the device to be commissioned. In one embodiment, the commissioning logic **402** comprises an existing memory element (e.g., random access memory (RAM), read-only memory (ROM), electrically programmable ROM (EPROM), flash memory) within the device **400**. In this embodiment, the commissioning logic **402** may be external to the microcontroller **401** or internal to the microcontroller **401**. An alternative embodiment comprehends stand-alone commissioning logic **402** (e.g., circuits and/or program instructions) that operates to transmit the commissioning packets independent from operation of other device functions provided for by the microcontroller **401**.

The touchscreen **404** and display **405** are employed to initiate execution of the program instructions by the microcontroller via, in one embodiment, launching a commissioning application on the smart wireless device **400**. Additionally, the display **405** may be employed to provide status and/or commissioned device information. The WLAN configuration data is optically transmitted by the flash **407** over an optical transmission patch **409** to the device to be commissioned. The wireless radio **406** may be employed via a WLAN **408** to communicate with a WLAN access point (not shown) or to the commissioned device following commissioning.

In lieu of using the camera flash **407**, an alternative embodiment of the smartphone commissioning device **400** employs a backlight function of the display **405** as the transmitter portion of the transmitting element, where the backlight is modulated on and off as described above. Other embodiments for optically transmitting the commissioning message are contemplated such as intensity modulation (e.g., brightness) and/or modulation of the color of the backlight.

Operationally, the remaining elements of the smart wireless device **400** work according to control provided by the commissioning logic **402** (or the commissioning logic **401** in conjunction with the microcontroller **401**) to cause the camera flash **407** to transmit the commissioning packets to the device to be commissioned, thereby allowing it to join the WLAN **408**. In one embodiment, after the commissioned device has joined the WLAN **408**, the wireless radio **406** is employed to optionally receive commissioning confirmation data from and/or to transmit supplemental configuration data to the commissioned device via the WLAN connection **408**.

As is appropriate for the chosen transmitting element and receiving element, a variety of well-known communication protocols, error correcting codes, line coding, and/or modulation techniques may be used to send the messages from the transmitter to the receiver in a reliable manner according to the present invention.

Now turning to FIG. 5, a block diagram is presented illustrating a commissioned device **500** according to the present invention. The commissioned device **500** may be commissioned onto a WLAN by the smart wireless device **400** of FIG. 4. The commissioned device **500** comprises a microcontroller **501** that is coupled to commissioning logic **502**. The microcontroller **501** is also coupled to an optical sensor **503** and to a wireless radio **504**. In the embodiment shown, the optical sensor **503** is employed as a transducer to receive WLAN configuration data from the wireless smart device **400** of FIG. 4 over an optical transmission link **509**.

Operationally, the microcontroller **501** includes processing circuits that are employed to decode and process WLAN configuration packets which are optically received by the optical sensor **503** over the transmission link **509**. The commissioning logic **502** includes program instructions that direct the microcontroller **501** to decode and process the WLAN configuration packets. In one embodiment, the commissioning logic **502** comprises an existing memory element within the device **500**. In this embodiment, the commissioning logic **502** may be external to the microcontroller **501** or internal to the microcontroller **501**. An alternative embodiment comprehends stand-alone commissioning logic **502** (e.g., circuits and/or program instructions) that operates to receive the commissioning packets independent from operation of other device functions provided for by the microcontroller **501**.

In one embodiment, after the commissioned device has joined the WLAN **508**, the wireless radio **506** is employed to optionally transmit commissioning confirmation data to and/or to receive supplemental configuration data from the smart wireless device and to communicate with the access point and other devices on the WLAN connection **508** as required.

Referring to FIG. 6, a block diagram is presented detailing features of an exemplary commissioning device **600** according to the present invention, such as the smart wireless device **400** of FIG. 4. The device **600** includes WLAN configuration data **601** that is provided to a forward error correction (FEC) encoder **602**. Encoded configuration data is generated by the encoder **602** and is provided to a line coding element **603**. One embodiment contemplates a Manchester encoder **603**. The line coding element **603** removes any direct current (DC) component from the configuration packets and allows transmission thereof to be self-clocking. Line coded configuration data is thus provided to a modulator **605**. The modulator **605** is also coupled to a timebase **604**. The modulator **605** combines the line coded data with timebase data to enables one or more corresponding optical messages to be serially transmitted. Modulated configuration data is thus provided by the modulator **605** to a flash **606**, which transmits the WLAN configuration packets over an optical transmission patch **609**. Elements **601-605** may be implemented by a combination of elements within the microcontroller **401** and commissioning logic **402** of FIG. 4.

Now referring to FIG. 7, a block diagram is presented showing details of an exemplary commissioned device **700** according to the present invention, such as the commissioned device **500** of FIG. 5. The commissioned device **700** includes an optical sensor **701** that is employed as a transducer to receive WLAN configuration data packets which

11

are transmitted over an optical channel **709** by a commissioning device according to the present invention. The received bit stream is provided by the sensor **701** to a demodulator **702** that is also coupled to a timebase **703**. The demodulator **702** may employ accurate timebase data to sample the transmission at a higher rate than the original timebase **604** used by the modulator **605** of FIG. 6. The output of the demodulator **702** is then sent to clock recovery **704**. The clock recovery utilizes the line coding added by the transmitter to discover the original timebase used by the modulator **605**. The clock recovery **704** then discards any extra samples and removes the line coding. The output of clock recovery **704** is then sent to an FEC decoder **705** which corrects any errors found in the transmission and recovers one or more commissioning messages. The commissioning messages provide the WLAN configuration data **706**, which is used by the product to join the WLAN. Elements **702-706** may be implemented by a combination of elements within the microcontroller **501** and commissioning logic **502** of FIG. 5.

Referring now to FIG. 8, a block diagram is presented illustrating an exemplary commissioning packet **800** according to the present invention. The packet **800** includes a preamble field **801**, a header field **802**, a service set identifier (SSID) field **803**, a key field **804**, and a cyclic redundancy check (CRC) field **805**. Operationally, the preamble **801** may be a 16-bit stream of repeating zeros and ones that allow a receiver according to the present invention to asynchronously synchronize with a transmitter according to the present invention. The header **802** may comprise two bytes, the first of which may be employed as a start frame delimiter and the second of which identifies a message type. The SSID **803** identifies the wireless network service set, and the key field **804** may comprise the network key. The CRC **805** is the result of a 16-bit CRC over the SSID **803** and key **804** fields using the polynomial $X^{16}+X^2+X+1$.

Although the present invention and its objects, features, and advantages have been described in detail, other embodiments are encompassed by the invention as well. For example, some embodiments may require that the device to be commissioned be placed into a commissioning mode to receive the commissioning messages. Other embodiments may accept the commissioning messages concurrent with other operations of the commissioned device.

Some embodiments may transmit the WLAN configuration data in a single commissioning message, while other embodiments employ multiple commissioning messages to communicate the configuration data.

Some embodiments of the present invention may transmit other information in addition to WLAN configuration data. In one such embodiment, the commissioning device sends its own Internet protocol (IP) address, proxy, and gateway configuration to the device to be commissioned. Once the device to be commissioned has successfully joined the WLAN, then the device proceeds to contact the commissioning device via the WLAN to indicate success and to optionally communicate additional configuration data as is described above.

In some embodiments, the commissioned device provides a visual or audio indication to the user about the status of the commissioning process. In one embodiment, the commissioned device may illuminate an indicator (e.g., light emitting diode (LED)) to indicate progress, success, or failure of the commissioning. In another embodiment, the commissioned device may use a speaker to provide an audio indication of the commissioning status.

12

One embodiment of the present invention includes a configuration mode where the commissioned device communicates back to the smart wireless device via optical or audio transmission. The smart wireless device may observe this communication with existing hardware, such as a built-in camera for reception of the optical transmission, or a microphone for reception of the audio transmission. This type of communication may be used to alert the smart wireless device (and operator) of success or failure conditions.

In other embodiments, the commissioning device may be informed of the WLAN configuration data in a variety of ways. In one embodiment, it may derive the WLAN configuration data from its own active connection to the WLAN. In another embodiment, WLAN configuration data may be entered by the user directly into the commissioning device via the touchscreen and/or keyboard. Another embodiment contemplates automatic derivation of the WLAN configuration data by contact with a server or via some other record of the configuration data. Combinations of the above noted embodiments are also comprehended for determination of the configuration data by the commissioning device.

Further embodiments contemplate usage of desktop and laptop computers as commissioning devices, where their display screens, backlights, speakers, microphones, and the like are employed for visual communication of commissioning messages.

Advantageously, a commissioned device according to the present does not require its own user interface, but rather the commissioning device's existing display and/or touchscreen may be utilized to input commissioning data and to output commissioning results.

In addition, from an operator perspective, configuration data is sent from the commissioning device to the product device in a single step rather than a complex series of single data entry actions.

Furthermore, since a smart wireless device is employed as a commissioning device, it already possesses the ability to automatically discover WLAN configuration, thus simplifying user experience.

Portions of the present invention and corresponding detailed description are presented in terms of software, or algorithms and symbolic representations of operations on data bits within a computer memory. These descriptions and representations are the ones by which those of ordinary skill in the art effectively convey the substance of their work to others of ordinary skill in the art. An algorithm, as the term is used here, and as it is used generally, is conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of optical, electrical, or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise, or as is apparent from the discussion, terms such as "processing" or "computing" or "calculating" or "determining" or "displaying" or the like, refer to the action and processes of a computer system, a microprocessor, a central processing unit, or similar electronic computing device, that manipulates and transforms data represented as physical, electronic

13

quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

Note also that the software implemented aspects of the invention are typically encoded on some form of program storage medium or implemented over some type of transmission medium. The program storage medium may be electronic (e.g., read only memory, flash read only memory, electrically programmable read only memory), random access memory magnetic (e.g., a floppy disk or a hard drive) or optical (e.g., a compact disk read only memory, or "CD ROM"), and may be read only or random access. Similarly, the transmission medium may be metal traces, twisted wire pairs, coaxial cable, optical fiber, or some other suitable transmission medium known to the art. The invention is not limited by these aspects of any given implementation.

The particular embodiments disclosed above are illustrative only, and those skilled in the art will appreciate that they can readily use the disclosed conception and specific embodiments as a basis for designing or modifying other structures for carrying out the same purposes of the present invention, and that various changes, substitutions and alterations can be made herein without departing from the scope of the invention as set forth by the appended claims.

What is claimed is:

1. A method for commissioning a target device onto a communication network, comprising:
 - at the target device including a microprocessor, a wireless radio, and commissioning logic:
 - receiving a request to initiate a commissioning process with the target device;
 - establishing a short-range signal transmission path with a wireless device independently of the communication network;
 - in response to receiving the request, receiving one or more network configuration packets from the wireless device via the short-range signal transmission path, wherein the one or more network configuration packets convey network configuration data associated with the communication network according to a predetermined data format, the network configuration data including at least a security passphrase for accessing the communication network;
 - executing an application on the target device to decode and process the one or more network configuration packets and recover the network configuration data; and
 - after receiving the network configuration packets, employing via the wireless radio the network configuration data conveyed in the one or more network configuration packets to enable the target device to join the communication network.
2. The method of claim 1, wherein the wireless device is selected from a group consisting of a smartphone and a tablet computer.
3. The method of claim 1, wherein information received via the short-range signal transmission path is received at a first wireless radio of the target device, and the first wireless radio employs at least one of a plurality of radio modulation techniques including frequency shift keying (FSK), phase shift keying (PSK), on-off keying (OOK), and quadrature amplitude modulation (QAM).
4. The method of claim 1, wherein the short-range signal transmission path is associated with a communication channel based on a proprietary RF communication standard.

14

5. The method of claim 1, wherein the short-range signal transmission path is associated with a communication channel based on one of Bluetooth communication, radio frequency identification (RFID), Zigbee, radio frequency for consumer electronics (RF4CE), Z-Wave, and near field communication.

6. The method of claim 1, further comprising: contacting the wireless device via the communication network to indicate that the target device has successfully entered the communication network.

7. The method of claim 6, further comprising: after contacting the wireless device, continuing the commissioning process, including receiving, via the communication network, device configuration data from the wireless device.

8. The method of claim 7, wherein the device configuration data including at least one of an Internet Protocol address, proxy, and gateway configuration.

9. The method of claim 1, further comprising: generating one of a visual indication and an audio indication to a user about a status of the commissioning process.

10. The method of claim 1, further comprising an application that includes a sequence of program instructions directed towards commissioning the target device onto the communication network.

11. A target device, comprising:

- a microprocessor;
- a wireless radio;
- a commissioning logic; and
- a memory storing programs for execution by the target device, the programs further including instructions for:
 - receiving a request to initiate a commissioning process with the target device;
 - establishing a short-range signal transmission path with a wireless device independently of a communication network;
 - in response to receiving the request, receiving one or more network configuration packets from the wireless device via the short-range signal transmission path, wherein the one or more network configuration packets convey network configuration data associated with the communication network according to a predetermined data format, the network configuration data including at least a security passphrase for accessing the communication network;
 - executing an application on the target device to decode and process the one or more network configuration packets and recover the network configuration data; and
 - after receiving the network configuration packets, employing via the wireless radio the network configuration data conveyed in the one or more network configuration packets to enable the target device to join the communication network.

12. The target device of claim 11, further comprising an application that includes a sequence of program instructions directed towards commissioning the target device onto the communication network.

13. The target device of claim 11, wherein the short-range signal transmission path includes an audio signal transmission path, and audio signals are modulated to transmit the one or more network configuration packets over the audio signal transmission path to an audio sensor of the target device.

14. The target device of claim 11, wherein the short-range signal transmission path includes a visual signal transmiss-

15

sion path, and visual signals are modulated to transmit the one or more network configuration packets over the visual signal transmission path to an optical sensor of the target device.

15. The target device of claim **11**, wherein the communication network includes a wireless local area network (WLAN).

16. A non-transitory computer readable storage medium storing one or more programs configured for execution by a target device for commissioning the target device onto a communication network, the target device including a microprocessor, a wireless radio, and commissioning logic, the one or more programs comprising instructions for:

receiving a request to initiate a commissioning process with the target device;

establishing a short-range signal transmission path with a wireless device independently of the communication network;

in response to receiving the request, receiving one or more network configuration packets from the wireless device via the short-range signal transmission path, wherein the one or more network configuration packets convey network configuration data associated with the communication network according to a predetermined data format, the network configuration data including at least a security passphrase for accessing the communication network;

executing an application on the target device to decode and process the one or more network configuration packets and recover the network configuration data; and

16

after receiving the network configuration packets, employing via the wireless radio the network configuration data conveyed in the one or more network configuration packets to enable the target device to join the communication network.

17. The non-transitory computer readable storage medium of claim **16**, wherein the one or more programs further comprise instructions for:

generating one of a visual indication and an audio indication to a user about a status of the commissioning process.

18. The non-transitory computer readable storage medium of claim **16**, further comprising an application that includes a sequence of program instructions directed towards commissioning the target device onto the communication network.

19. The non-transitory computer readable storage medium of claim **16**, wherein the one or more programs further comprise instructions for:

contacting the wireless device via the communication network to indicate that the target device has successfully entered the communication network.

20. The non-transitory computer readable storage medium of claim **19**, wherein the one or more programs further comprise instructions for:

after contacting the wireless device, continuing the commission process, including receiving, via the communication network, device configuration data from the wireless device.

* * * * *

Exhibit 5

THE **smart
audio** REPORT

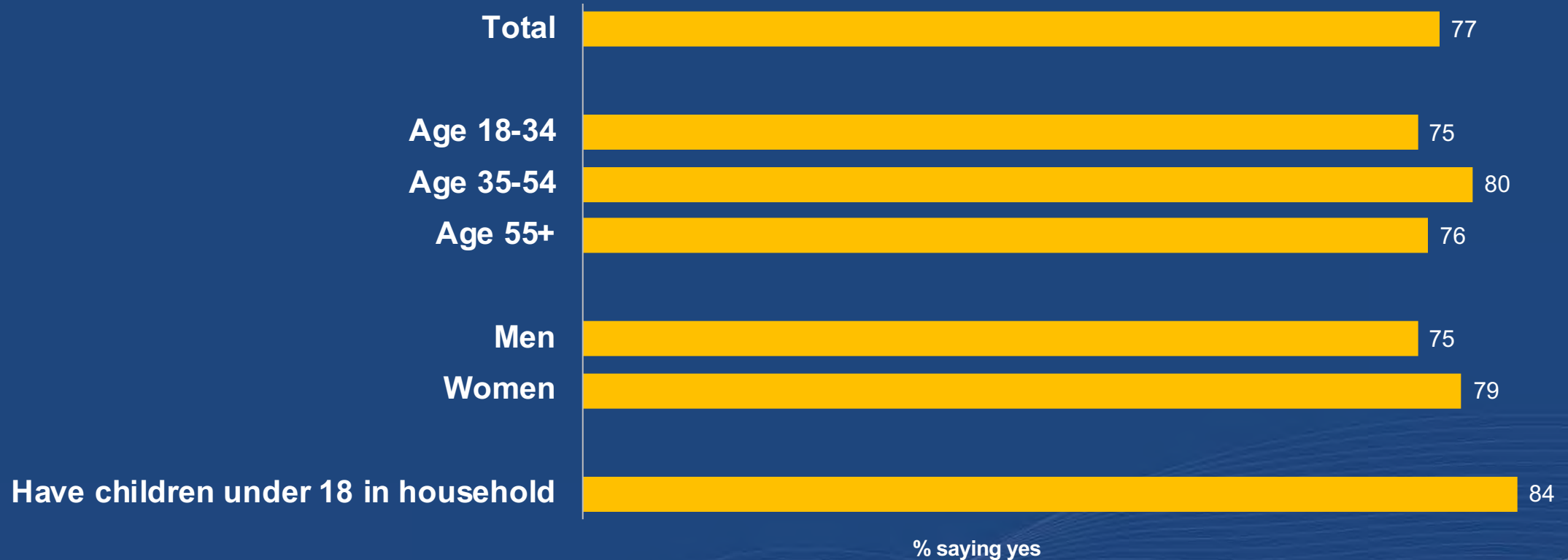
#SmartAudioReport
npr.org/smartaudio

Methodology

- National telephone survey of 1015 adults age 18 and older
 - Conducted 12/31/2019 – 1/5/2020
 - Tracking from previous telephone study in December 2018
- National online survey of adults age 18 and older
 - Conducted 3/31/2010 – 4/1/2020, starting 20 days after COVID-19 was declared a pandemic by the WHO on 3/11/2020
 - 815 smart speaker owners
 - 428 do not own smart speaker; use voice-operated assistants on at least one other device (Voice-users/no smart speaker)
 - 417 do not use voice-operated assistants at all (Non-voice users)
 - Tracking from previous online studies from Spring 2017 and Spring 2018
- Ethnographic interviews in Denver, Kansas City, Orlando
 - Conducted in Feb 2020

Total U.S. online population 18+

Has your typical routine changed due to the outbreak of COVID-19?

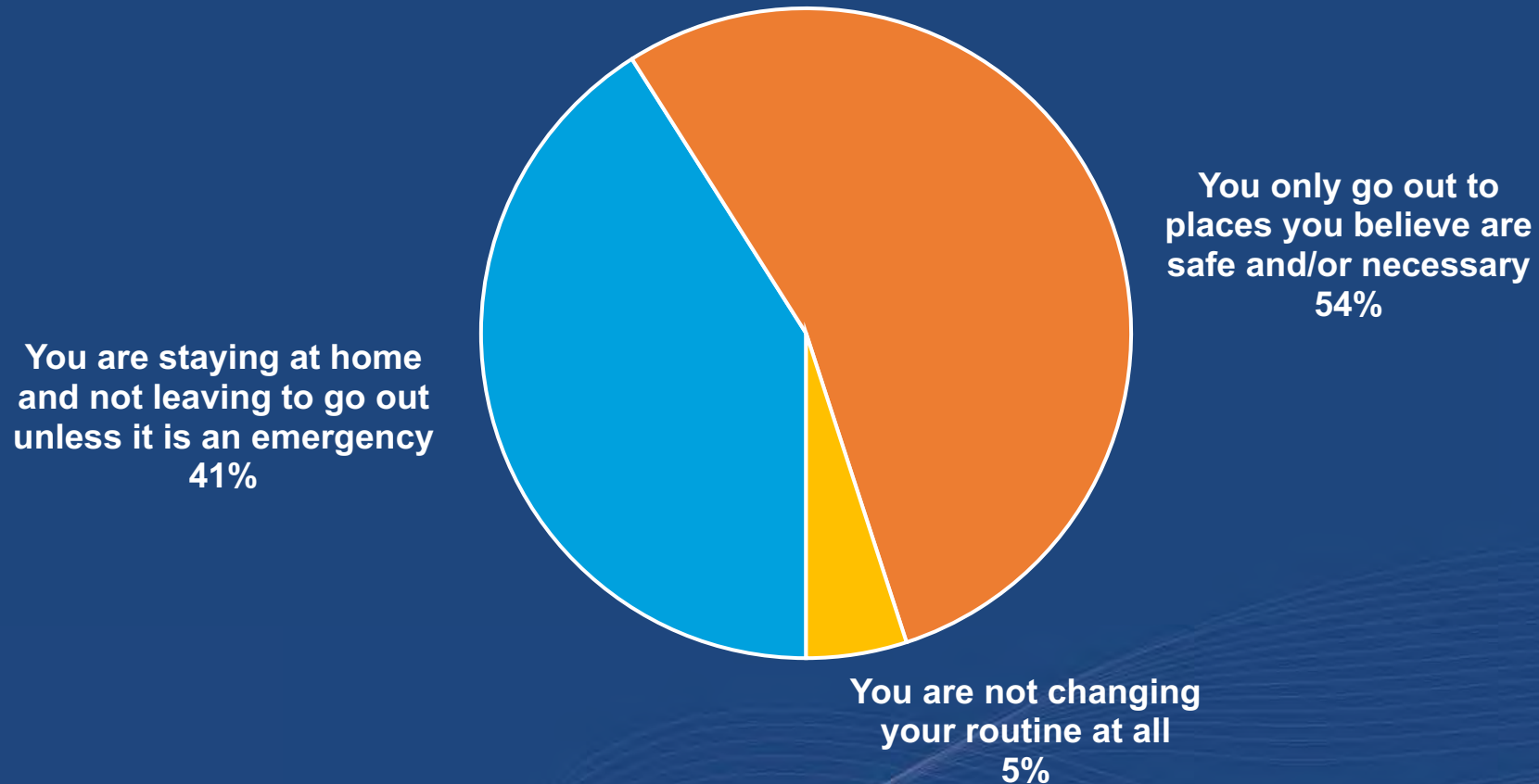


Spring 2020

Base: Total U.S. Online population 18+

Total U.S. online population 18+

Since the outbreak of COVID-19, which best describes your daily routine?

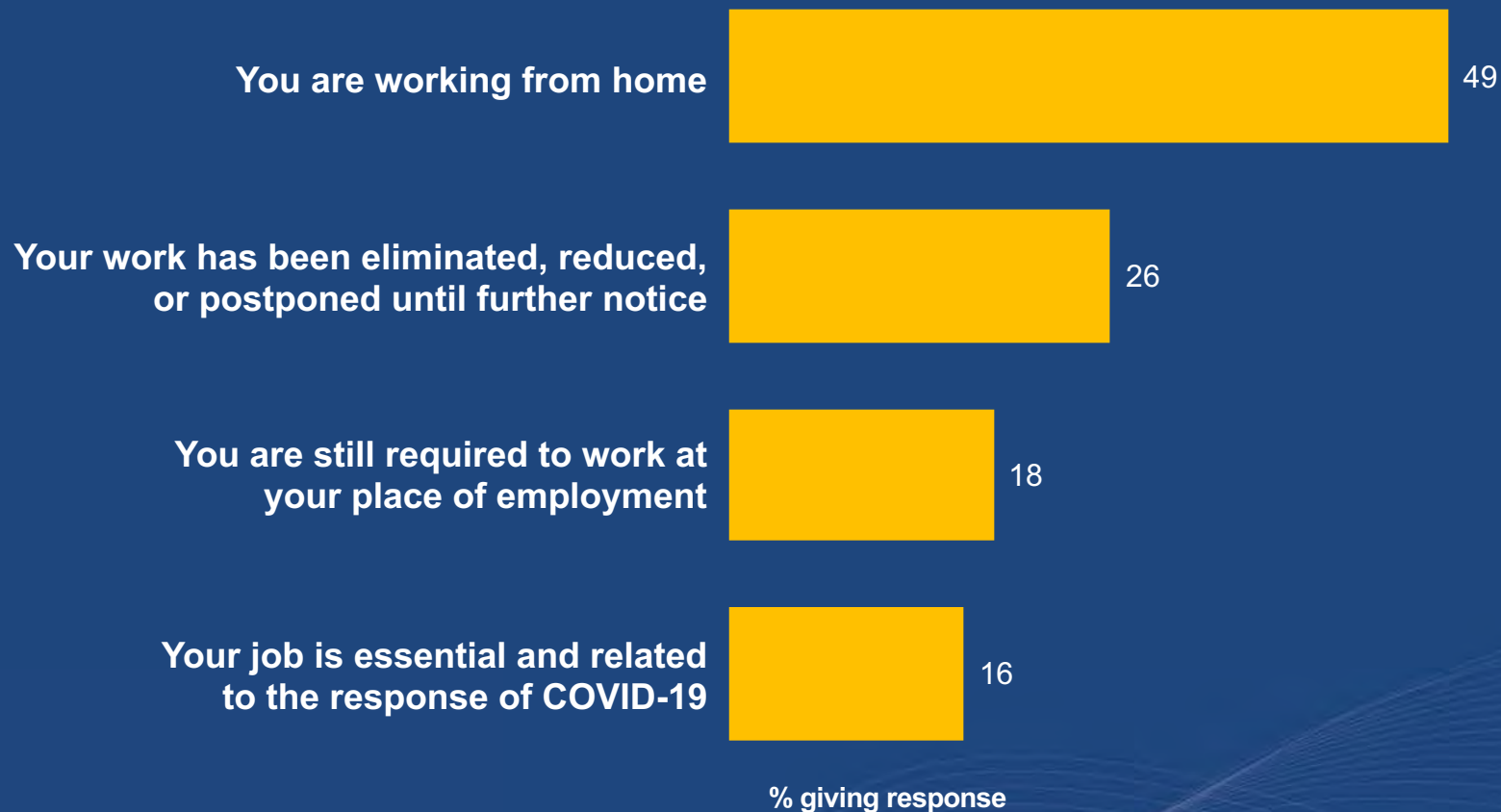


Spring 2020

Base: Total U.S. Online population 18+

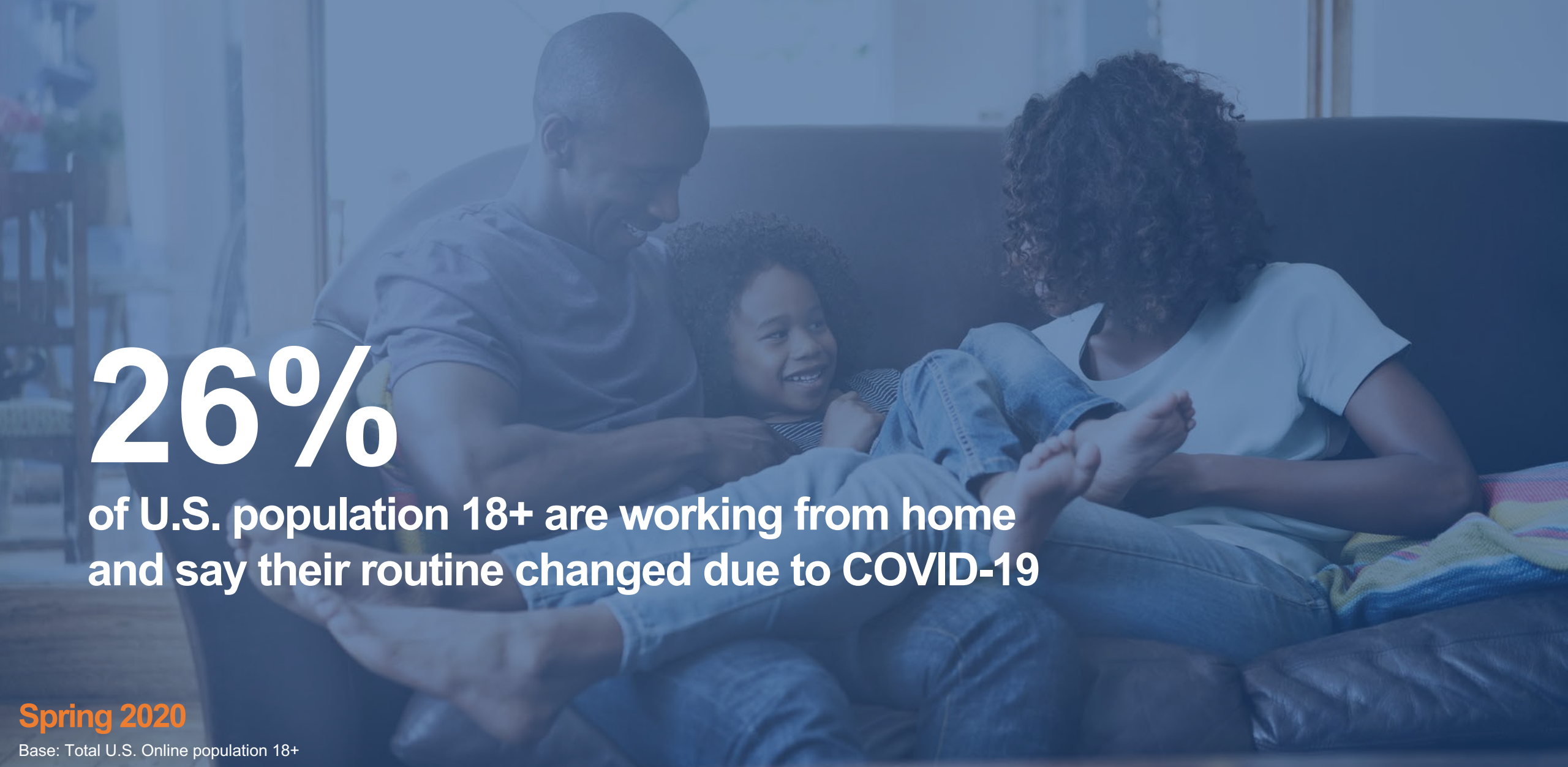
Employed/Temporarily Unemployed

Since the outbreak of COVID-19, which best describes your work routine?



Spring 2020

Base: Age 18+, Full-time employed, part-time employed, or temporarily unemployed (61%)

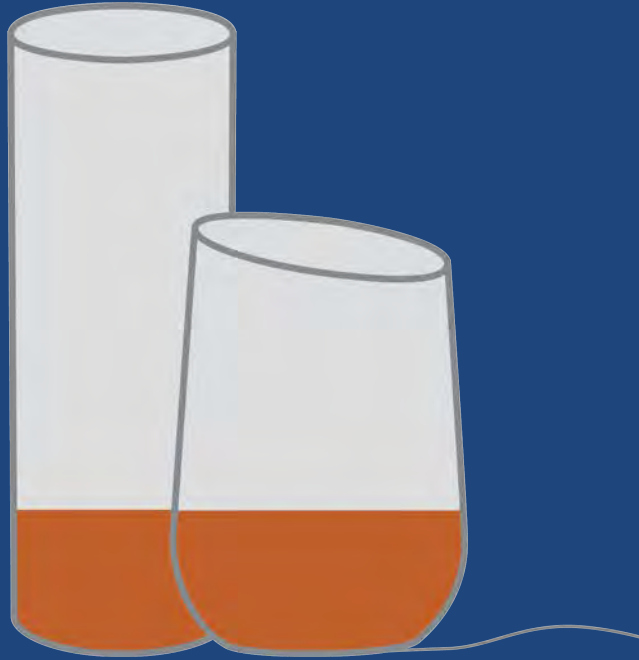


26%

of U.S. population 18+ are working from home
and say their routine changed due to COVID-19

Spring 2020

Base: Total U.S. Online population 18+



24% of Americans
18+ own a Smart Speaker,
or around 60 million people

In 2019, 21% of U.S. adults 18+ owned a
smart speaker, or around 53 million people

Google Home Google Home is a trademark of Google Inc.

Smart speaker owners % owning type of smart speaker

Any Amazon speaker

78

Any Google speaker

41

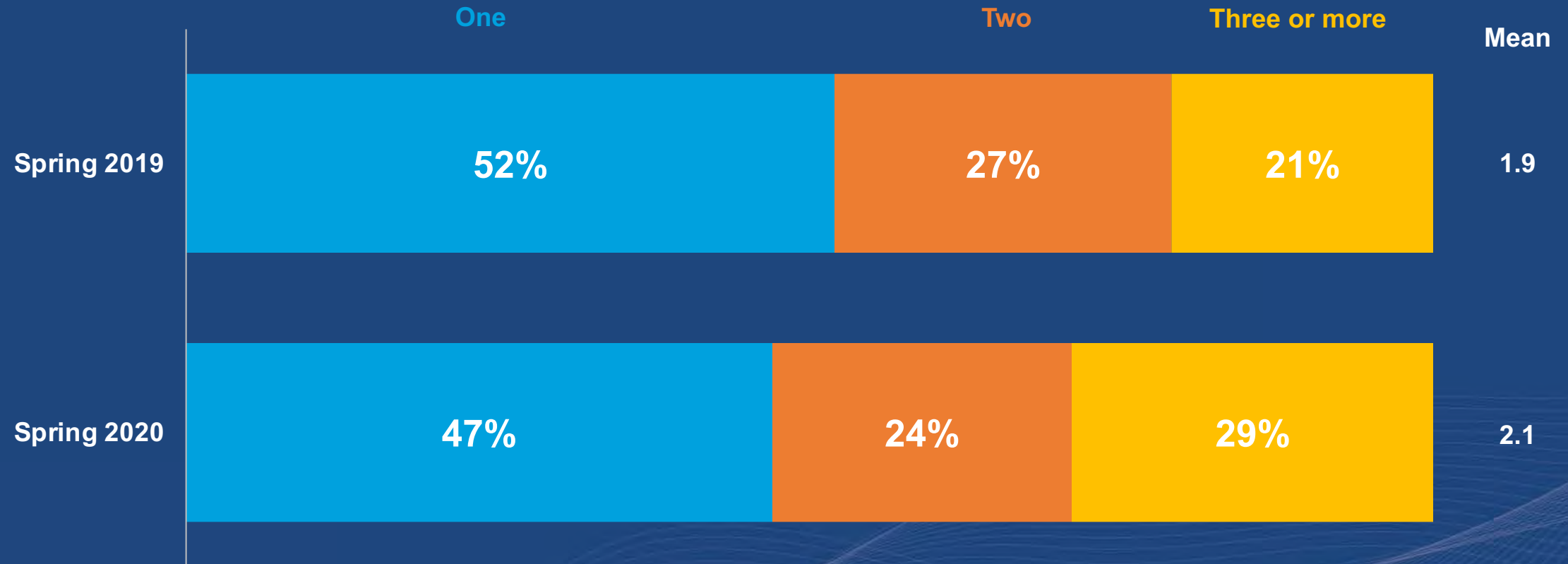


Spring 2020

Base: Own a smart speaker

Smart speaker owners

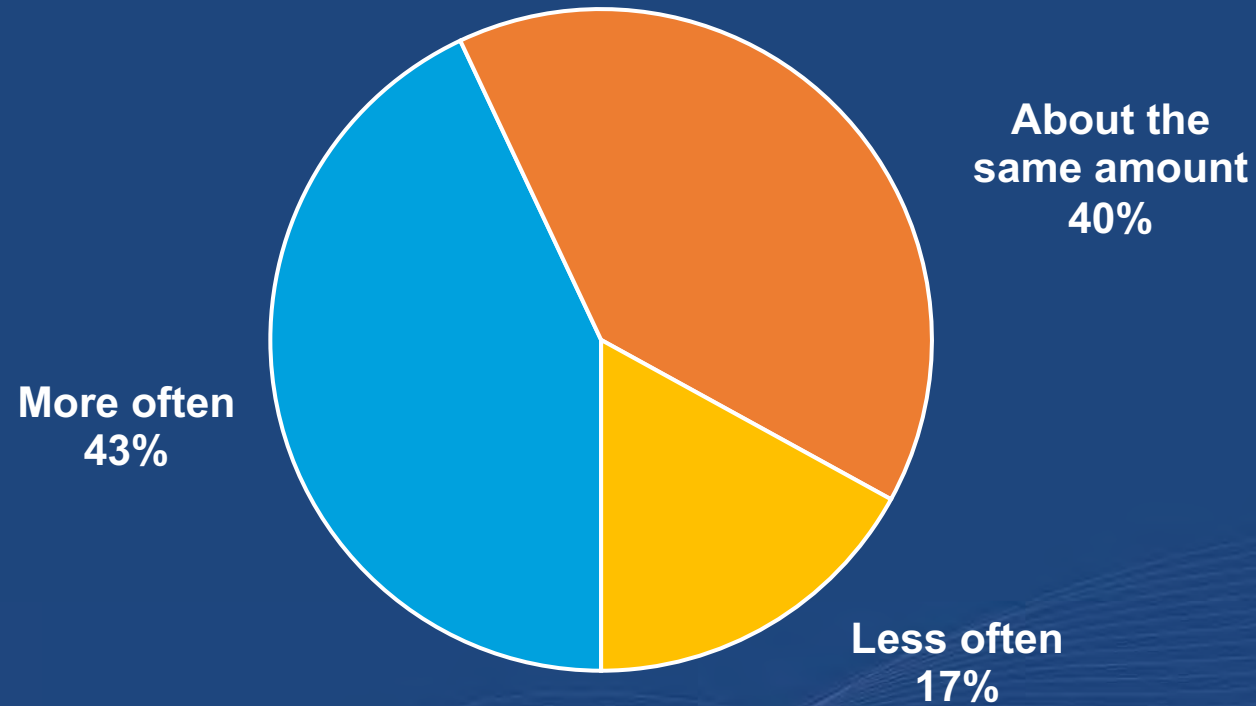
How many smart speakers do you have in your home?



Base: Own a smart speaker

Smart speaker owners

Compared to the first month you owned your smart speaker,
are you using your smart speaker...?

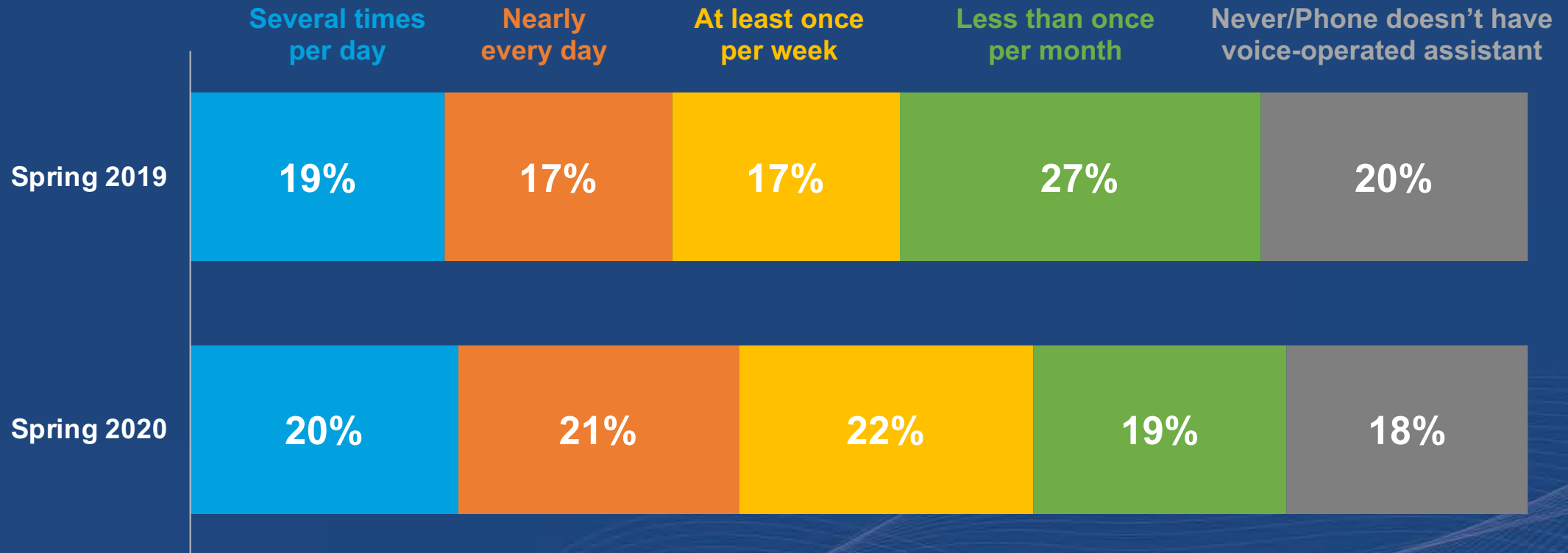


Spring 2020

Base: Own a smart speaker

Smart speaker owners and smartphone voice users

How often do you use the voice-operated personal assistant on your smartphone?

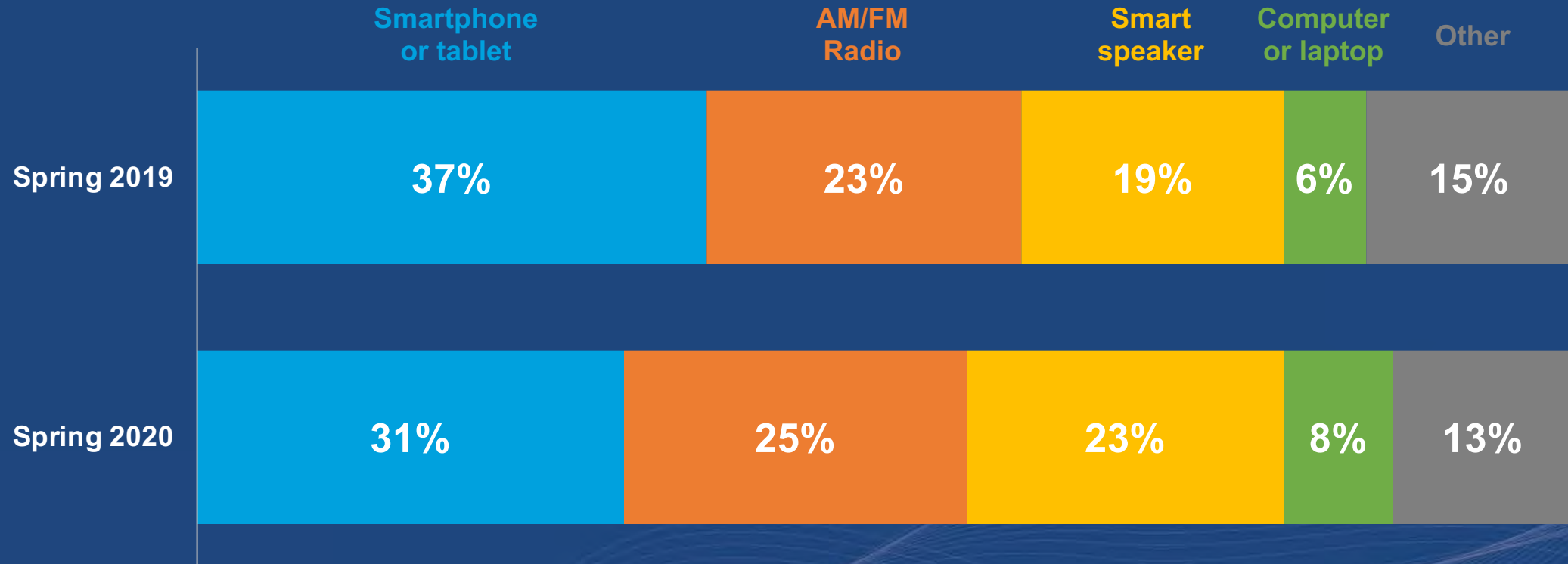


Spring 2020

Base: Own a smart speaker and own a smartphone

Smart speaker owners

How do you listen to audio most often?



Spring 2020

Base: Own a smart speaker

Voice Assistants

Total U.S. Online population 18+

Do you currently ever use a voice-operated personal assistant...



% saying yes

Spring 2020

Base: Total U.S. Online population 18+



56%

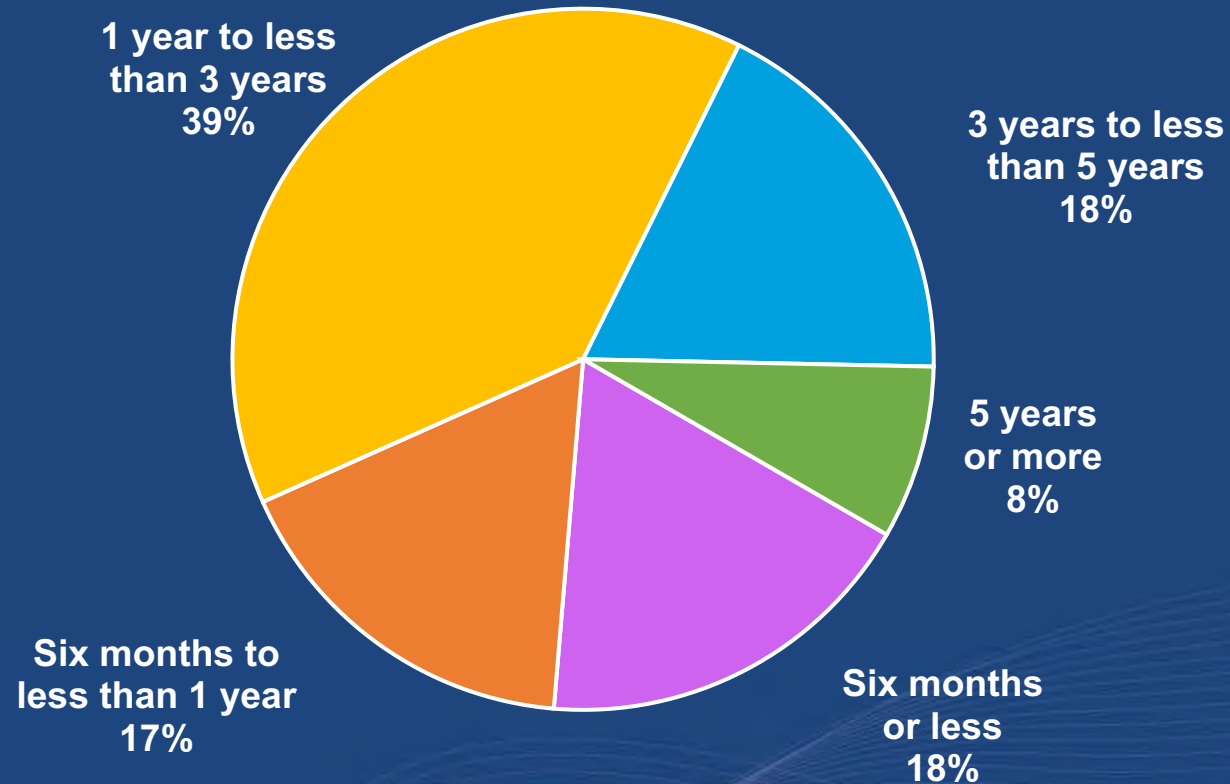
of those who **use a voice assistant on a smartphone**
say they keep the voice-operated personal assistant
on their smartphone turned on all the time

Spring 2020

Base: Use the voice assistant on smartphone

Voice-operated assistant users

How long have you been using voice-operated assistants of any kind?

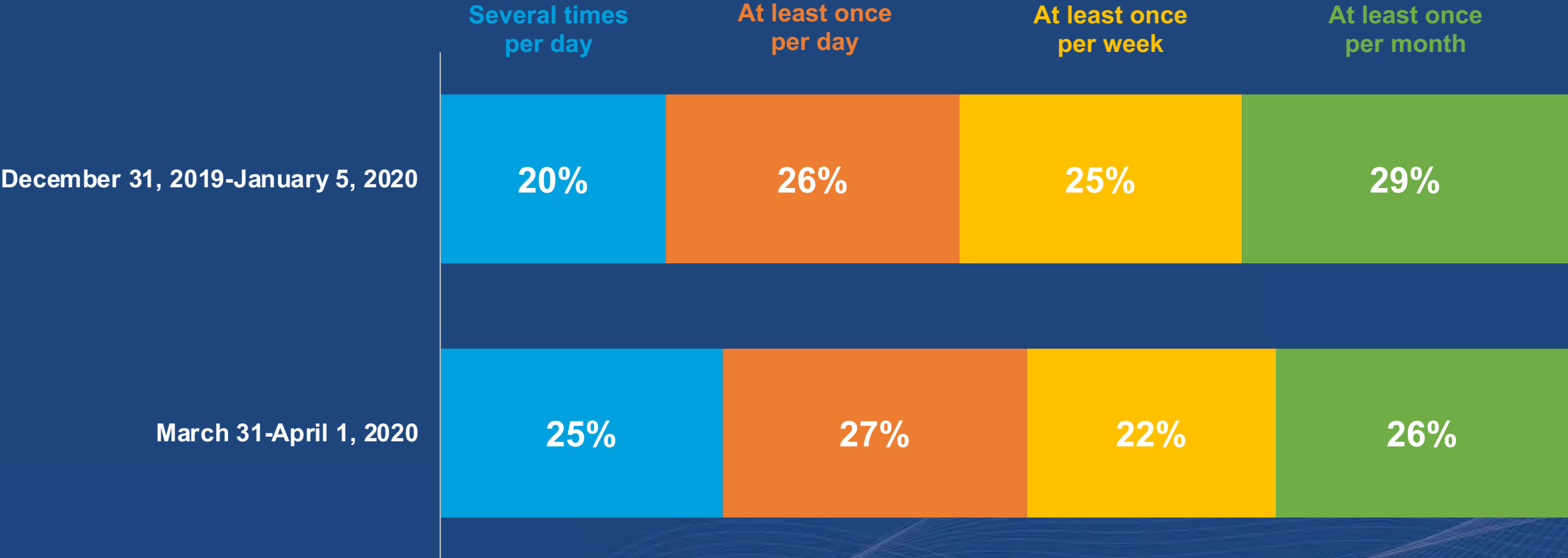


Spring 2020

Base: Ever use voice-operated personal assistants

Voice-operated assistant users

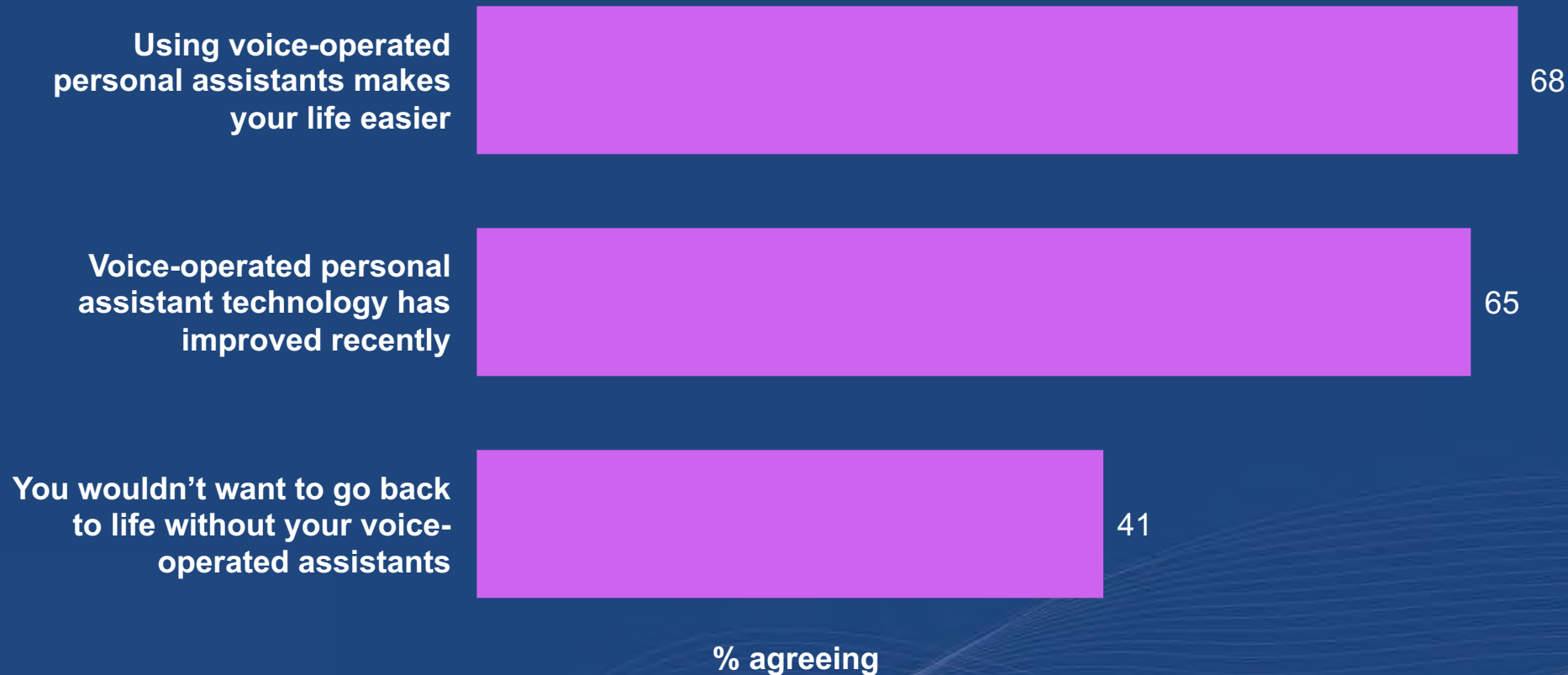
How often do you use voice commands?



Spring 2020

Base: Ever use voice-operated personal assistants

Voice-operated assistant users agree:



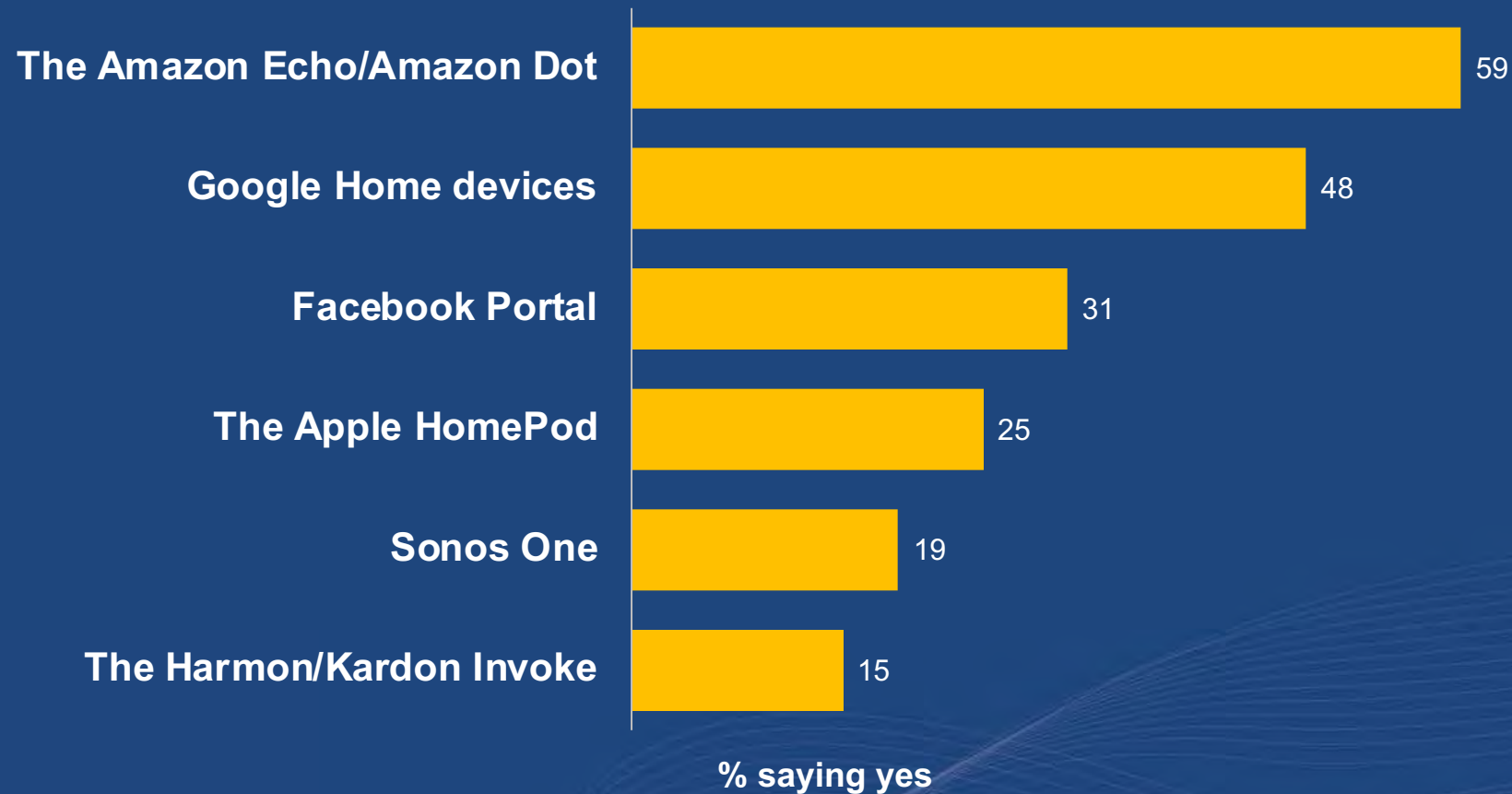
Spring 2020

Base: Smart speaker non-owners

Smart Speaker Non-Owners

Smart speaker non-owners

Have you ever heard of...?



Spring 2020

Base: Smart speaker non-owners

Smart speaker non-owners

% saying this is a reason they do not own a smart speaker

It bothers you that voice-enabled smart speakers are always listening

66

You worry that hackers could use voice-enabled smart speakers to get access to your home or personal information

65

You do not trust the companies that make the smart speakers to keep your information secure

58

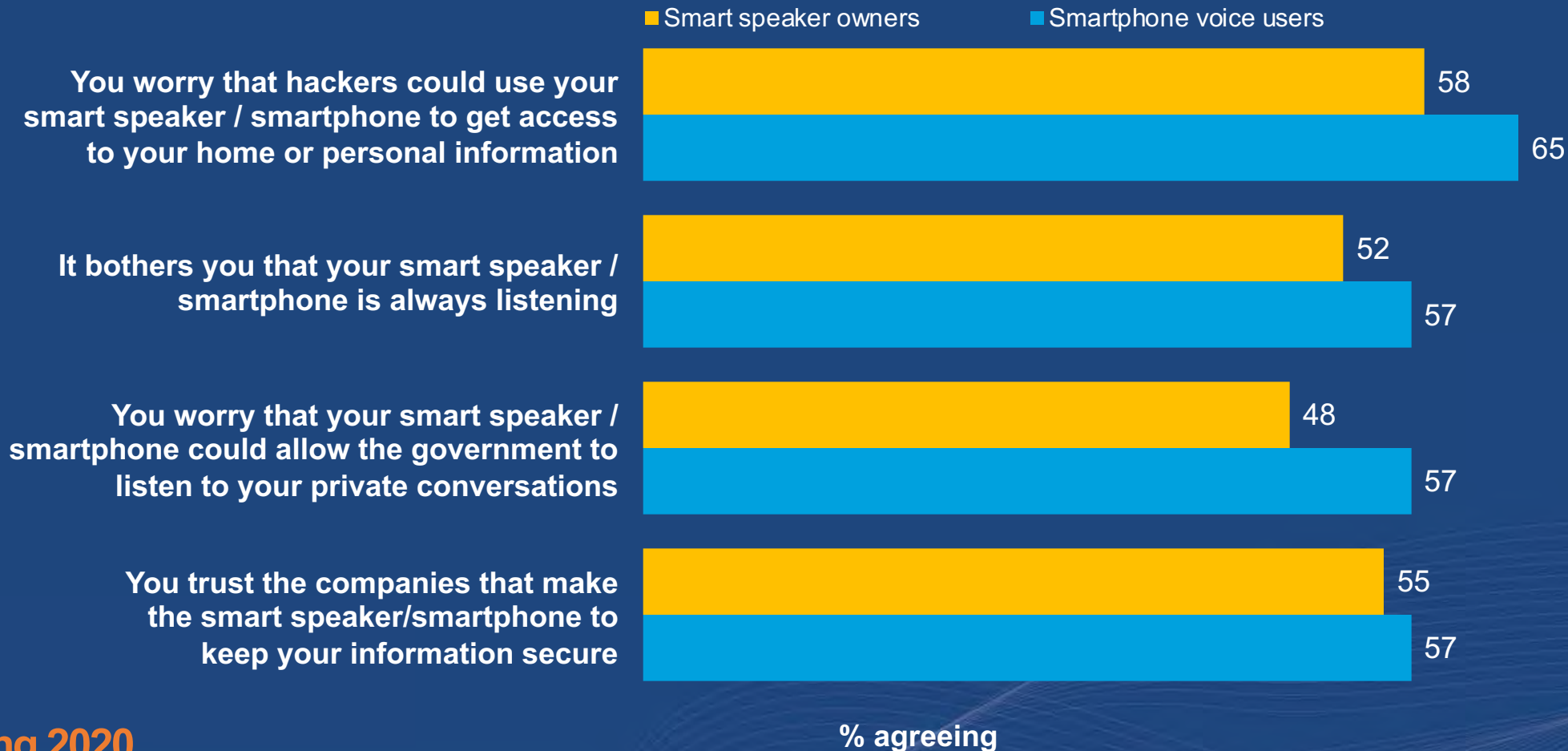
You worry that voice-enabled speakers could allow the government to listen to your private conversations

46

Spring 2020

Base: Smart speaker non-owners

Agree/Disagree: **Smart speaker owners** vs. **Smartphone voice users**



Spring 2020

Base: Own a smart speaker/Smartphone voice users

Smart speaker non-owners

How likely are you to purchase a voice-enabled speaker in the next six months?

Very likely Somewhat likely Not at all likely

All smart speaker non-owners

7%

27%

66%

Smart speaker non-owners who use voice commands

14%

38%

48%

Base: Own a smart speaker



46%

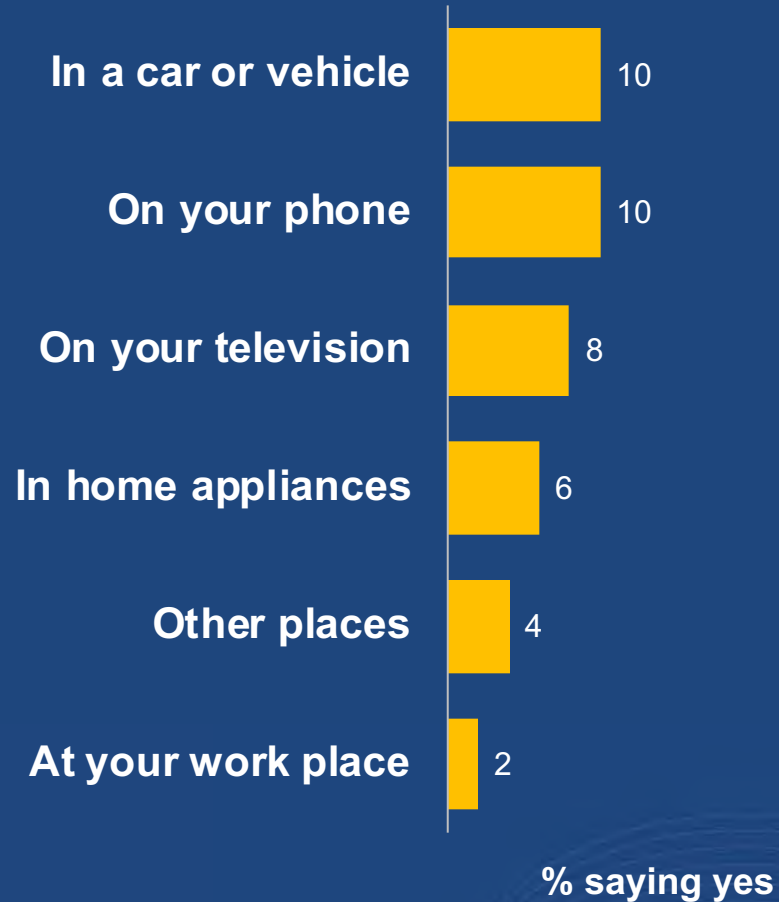
of **smart speaker owners who own a smartphone**
say they are using the voice-operated personal assistant
on their smartphone more since getting a smart speaker

Spring 2020

Base: Smart speaker owners who own a smartphone

Non voice-users

% interested in having voice assistant technology



% saying yes

Spring 2020

Base: Non-voice users

How are people using voice commands?

Smart speaker owners

How do you discover new skills or actions for your smart speakers?



Spring 2020

Base: Own a smart speaker



Own a smart speaker and use
voice assistant on smartphone:

59% say the
tasks completed using a **smartphone
voice-operated assistant** are mostly
different from the tasks completed
using a **smart speaker**

Spring 2020

Base: Smart Speaker owners who use voice assistant on smartphone

Smart speaker owners vs. Smartphone voice users: In a typical week do you request your [device] to...

Task	Smart speaker owners	Smartphone voice users	Difference
Play music	85%	51%	+34
Check the time	62%	40%	+22
Receive reminders	42%	21%	+21
Tell a joke or do something else funny	44%	25%	+19
Set a timer/alarm	65%	47%	+18
Get the weather	74%	58%	+16
Listen to an AM/FM radio station	45%	30%	+15
Play a game	24%	27%	-3
Read short stories	17%	20%	-3
Order food	18%	24%	-6
Find restaurants or businesses in your area	31%	38%	-7
Get the traffic	28%	36%	-8
Make a phone call	32%	46%	-14
Read or respond to a text message	Not asked	39%	--

Spring 2020

**Smartphone voice assistant
users request**

8.8

**different types of tasks on
their smartphone in a
typical week**

Base: Smartphone voice users

**Smart speaker
owners request**

10.8

**different types of tasks on
their smart speaker in a
typical week**

Base: Smart speaker owners

Spring 2020

2019:

Smart speaker owners request

9.4

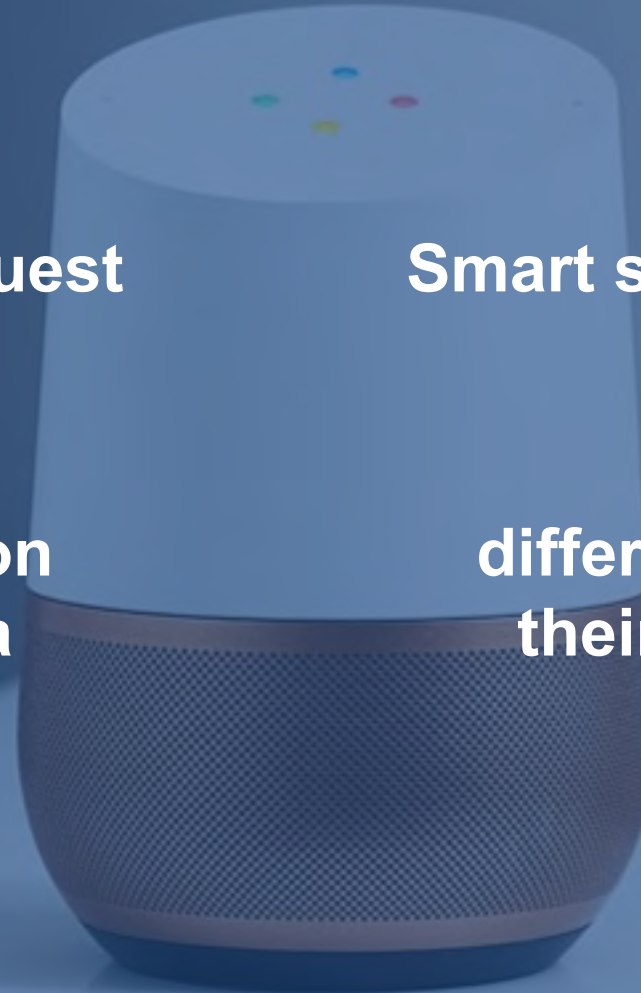
different types of tasks on
their smart speaker in a
typical week

2020:

Smart speaker owners request

10.8

different types of tasks on
their smart speaker in a
typical week



Spring 2020

Base: Own a smart speaker

Smart speaker owners and smartphone voice users agree:

The voice assistant on your smart speaker is better than the voice assistant on your smartphone

55

The voice assistant on your smartphone is better than the voice assistant on your smart speaker

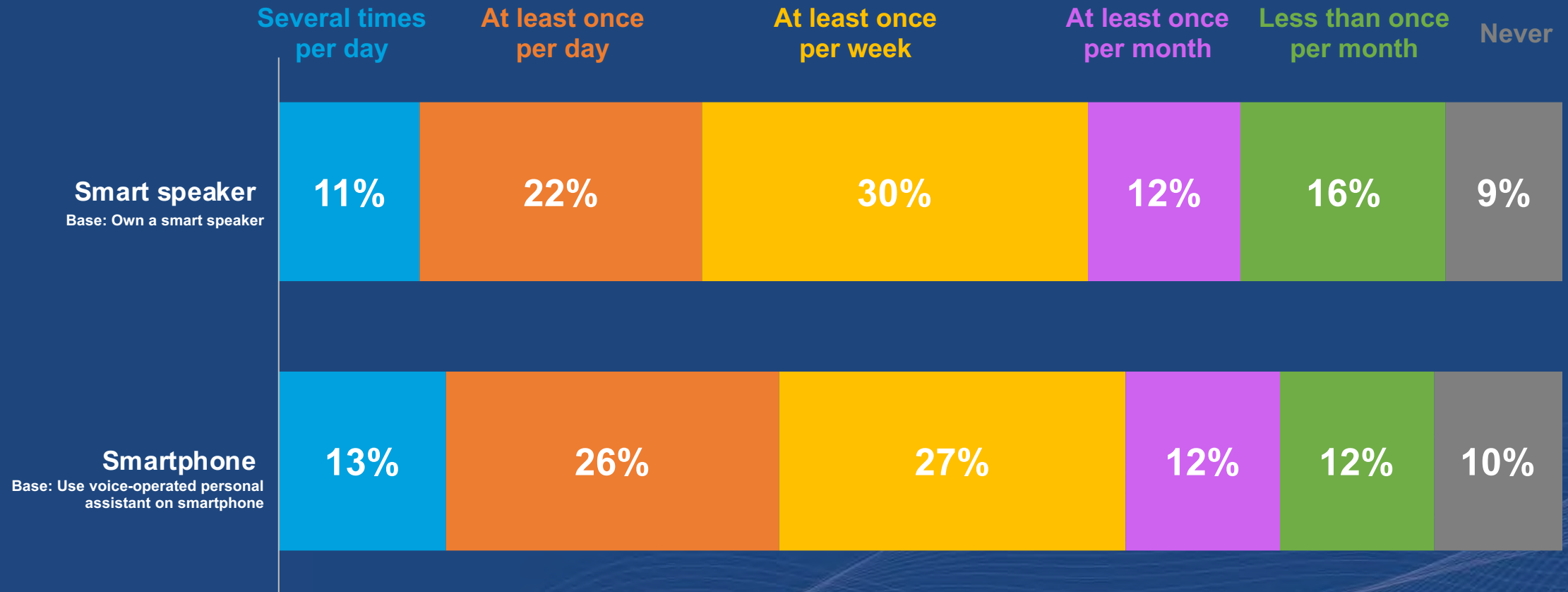
45

% agreeing

Spring 2020

Base: Use voice-operated assistant technology on smartphone and on a smart speaker (82%)

How often does [device] misunderstand or mishear a request?



Spring 2020

Smart speaker owners

“You don't know enough about your smart speaker to use all its features”

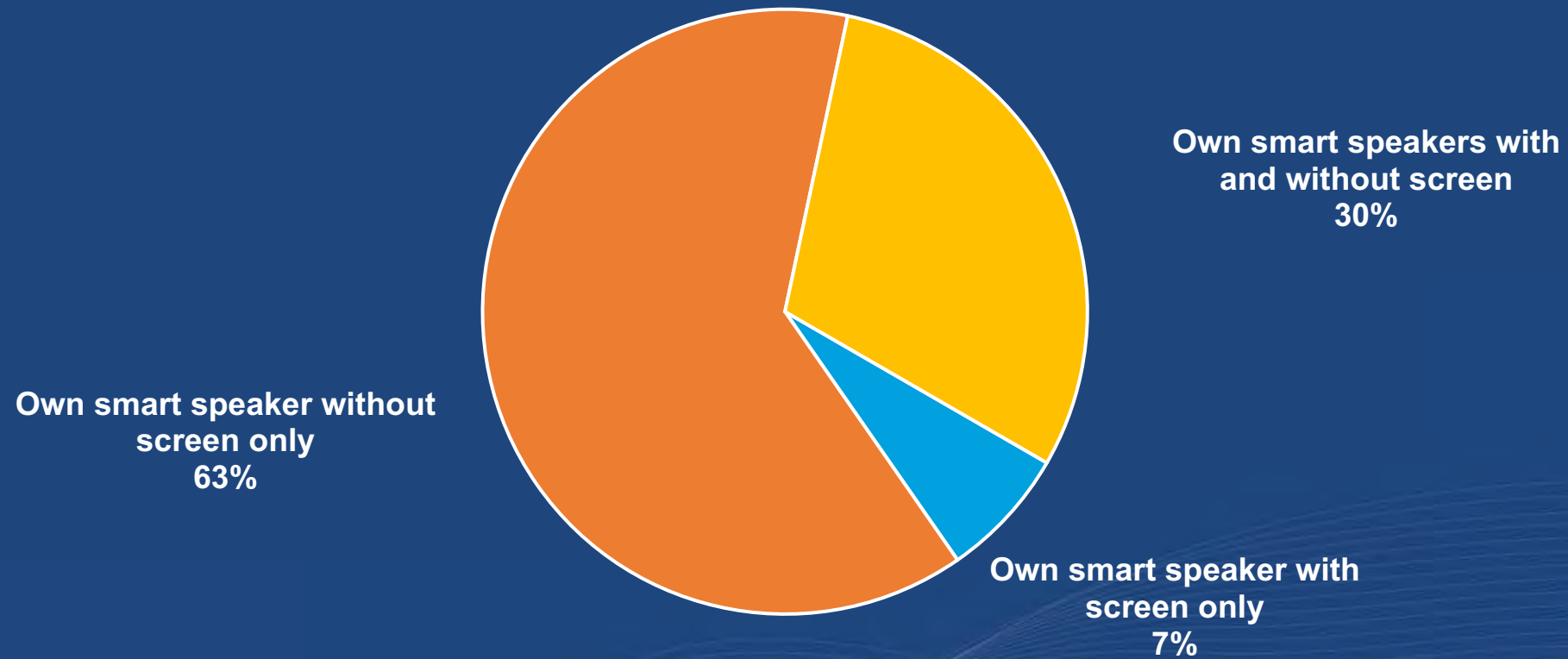


Base: Own a smart speaker

Smart Speakers with Screens

Smart speaker owners

Type of smart speaker owned

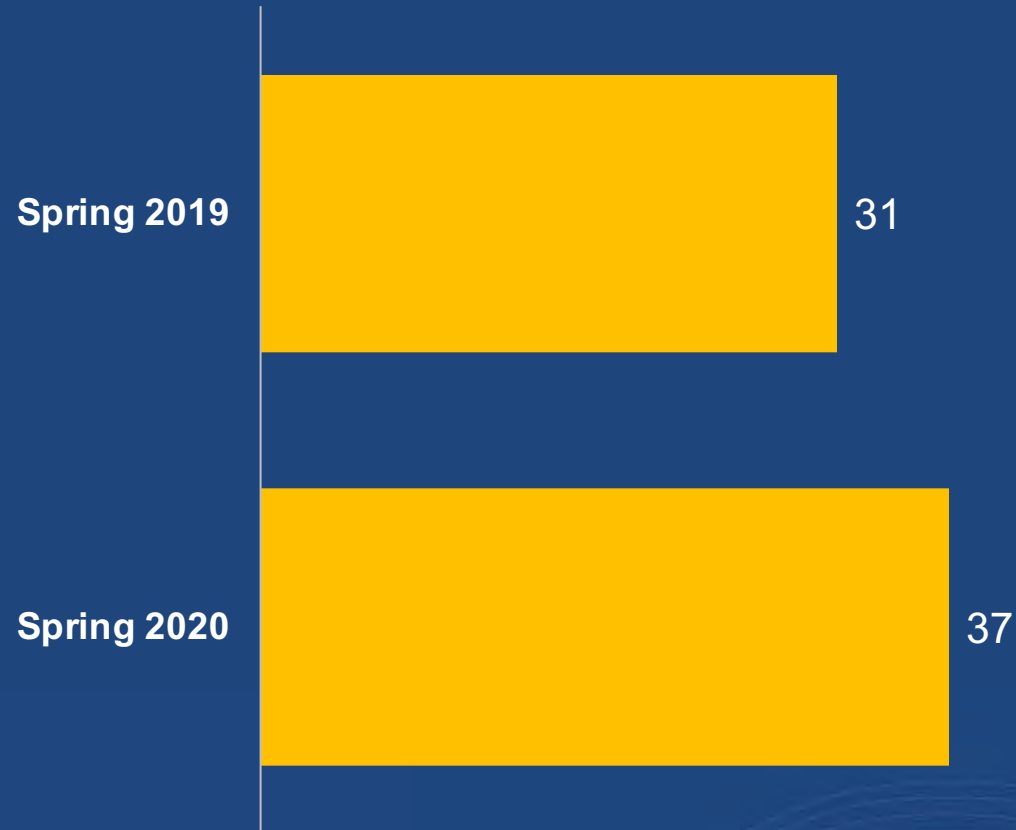


Spring 2020

Base: Smart speaker owners

Smart speaker owners

% owning a smart speaker with a screen:

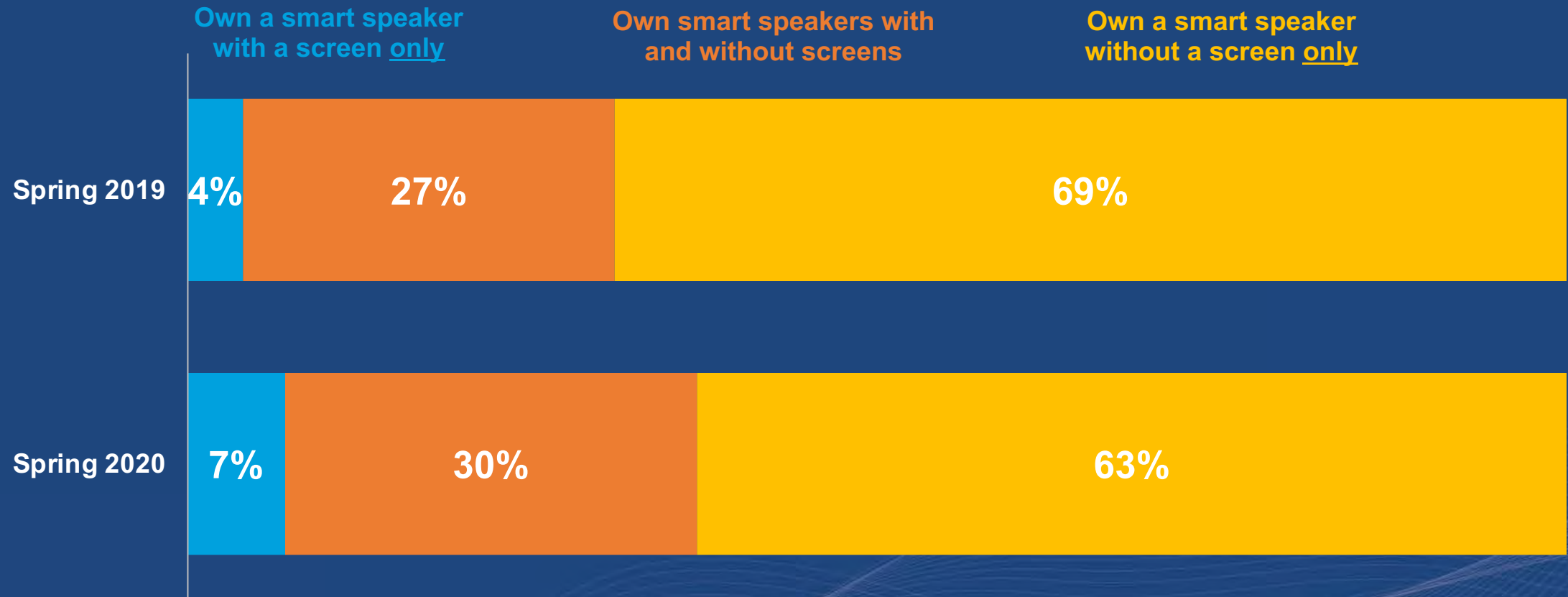


Spring 2019

Base: Own a smart speaker

Smart speaker owners

Type of smart speaker owned



Base: Own a smart speaker

Smart speaker non-owners

How likely are you to purchase a voice-enabled smart speaker that comes **with a video screen** in the next six months?

Very likely Somewhat likely Not at all likely

Smart speaker non-owners

8%

22%

70%

Smart speaker non-owners,
voice command users

15%

33%

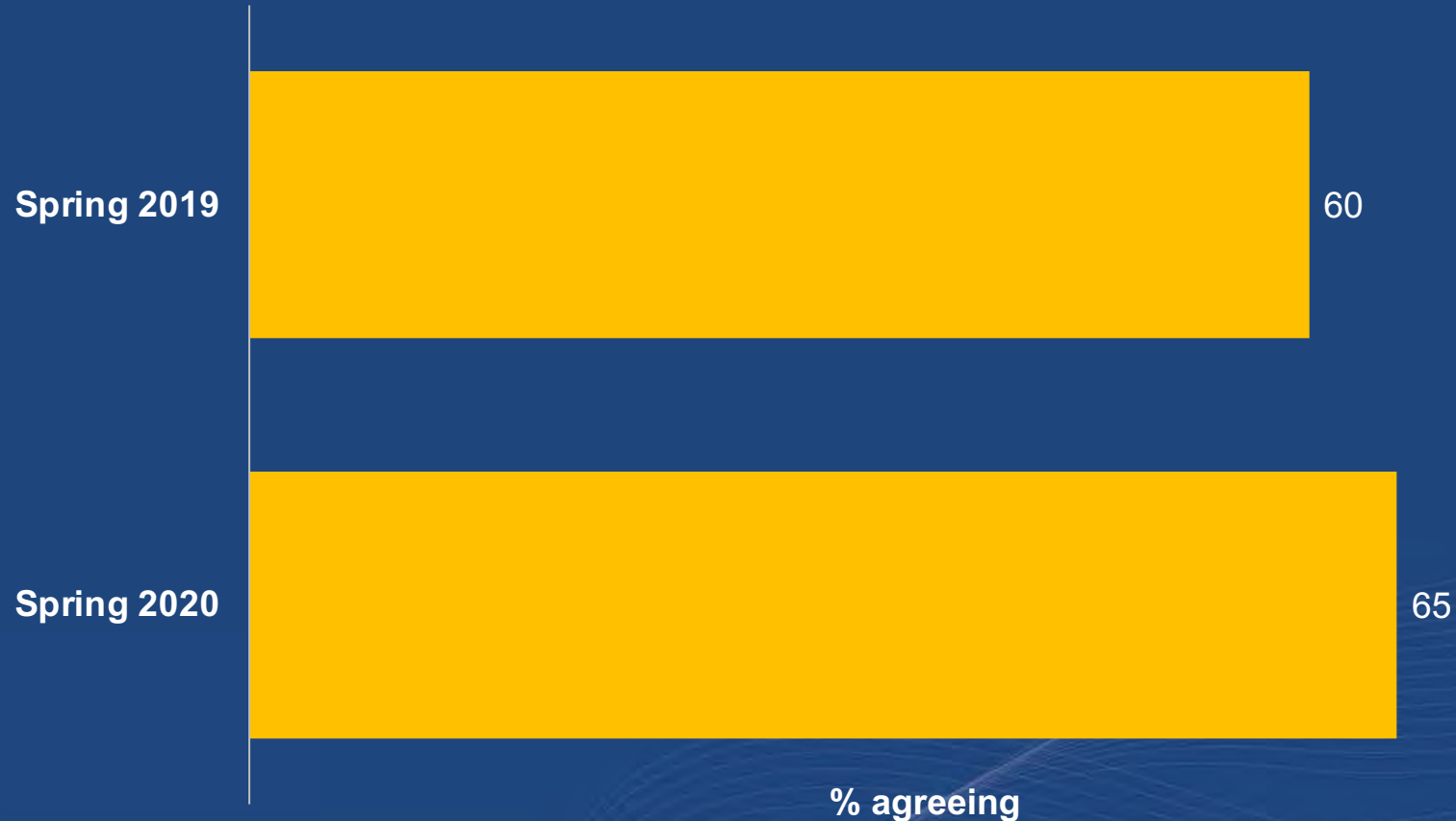
52%

Spring 2020

Base: Smart speaker non-owners

Own a smart speaker with a screen

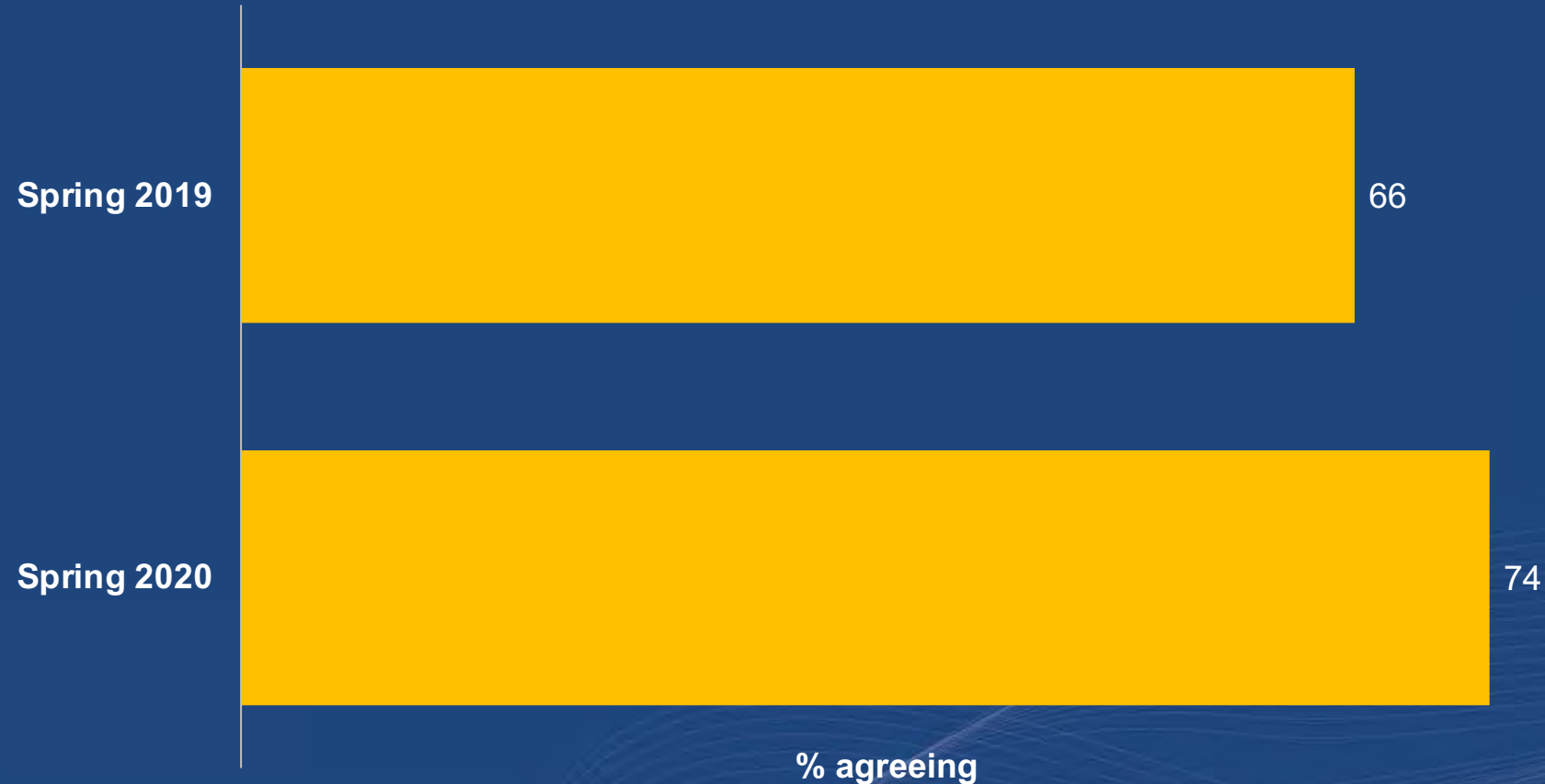
% who agree: “You prefer your smart speaker with a screen over a smart speaker without a screen”



Base: Own a smart speaker with a screen

Own a smart speaker with a screen

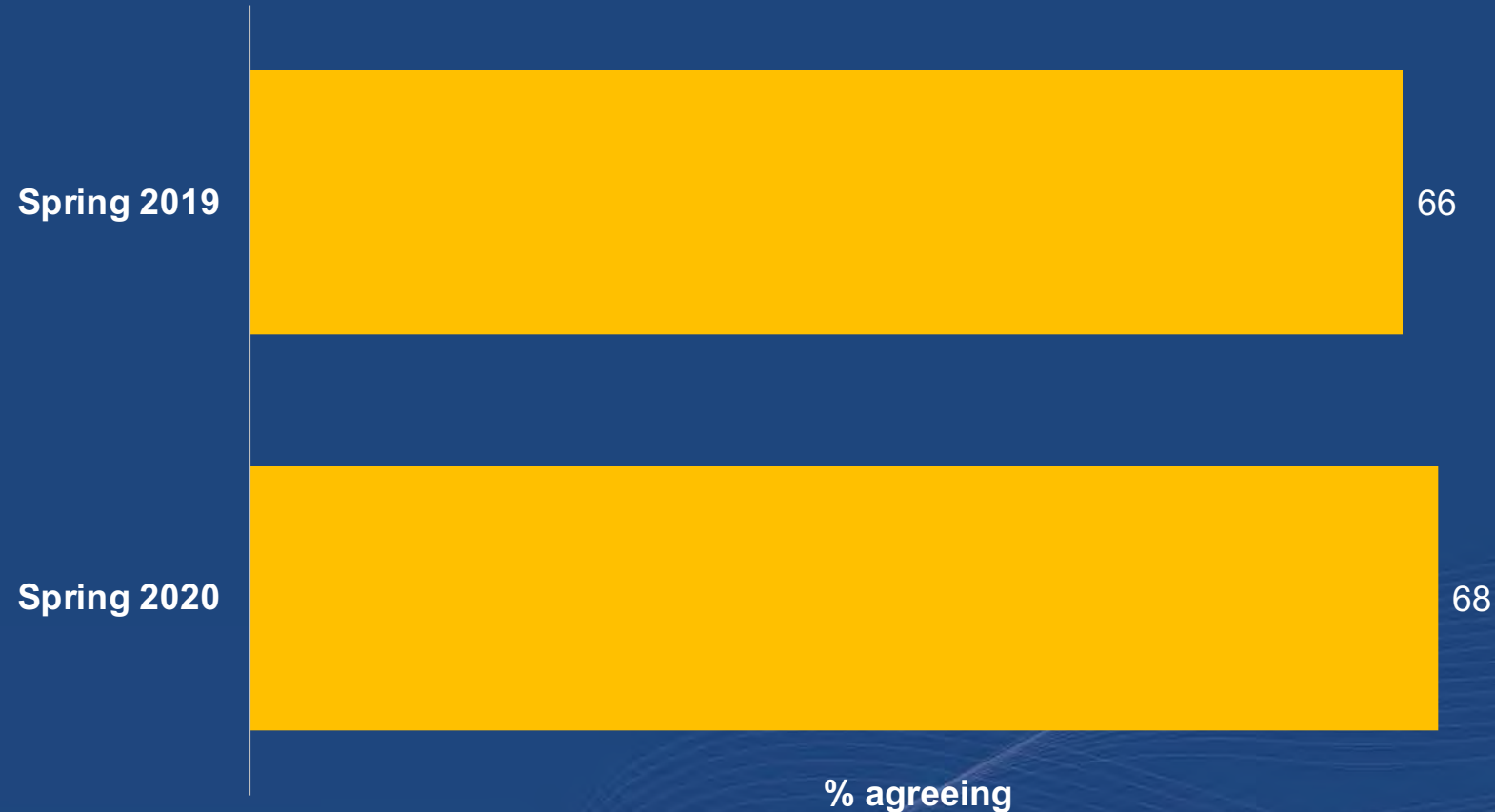
% who agree: “Having a screen on your smart speaker makes it easier to use”



Base: Own a smart speaker with a screen

Own a smart speaker with a screen

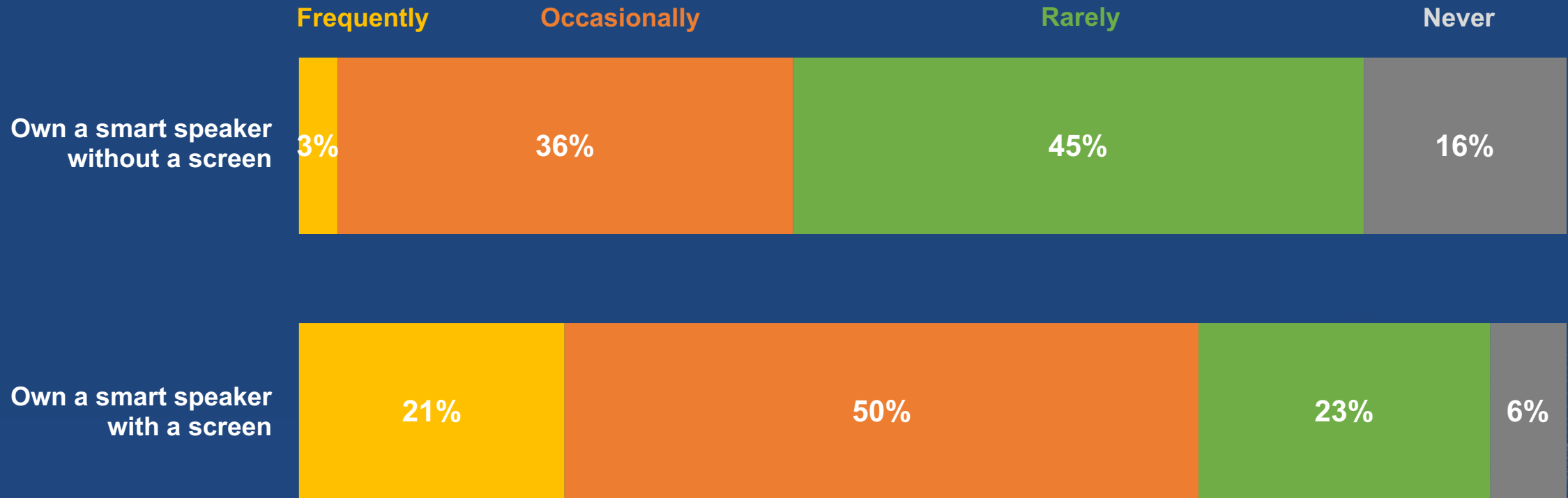
% who agree: “The screen on your smart speaker has made it easier to discover new content”



Base: Own a smart speaker with a screen

Smart speaker owners

How often do you seek out new skills or actions to use on your smart speaker?



Spring 2020

Base: Own a smart speaker

COVID-19 Related Behaviors

Smart speaker owners with children under 18 in household
**% who agree: “You want to buy another smart speaker
to entertain children in more rooms of the house”**



% saying “yes, applies to you”

Base: Smart speaker owners with children under 18 in household

Smart speaker owners % who ever listen to news on their smart speaker

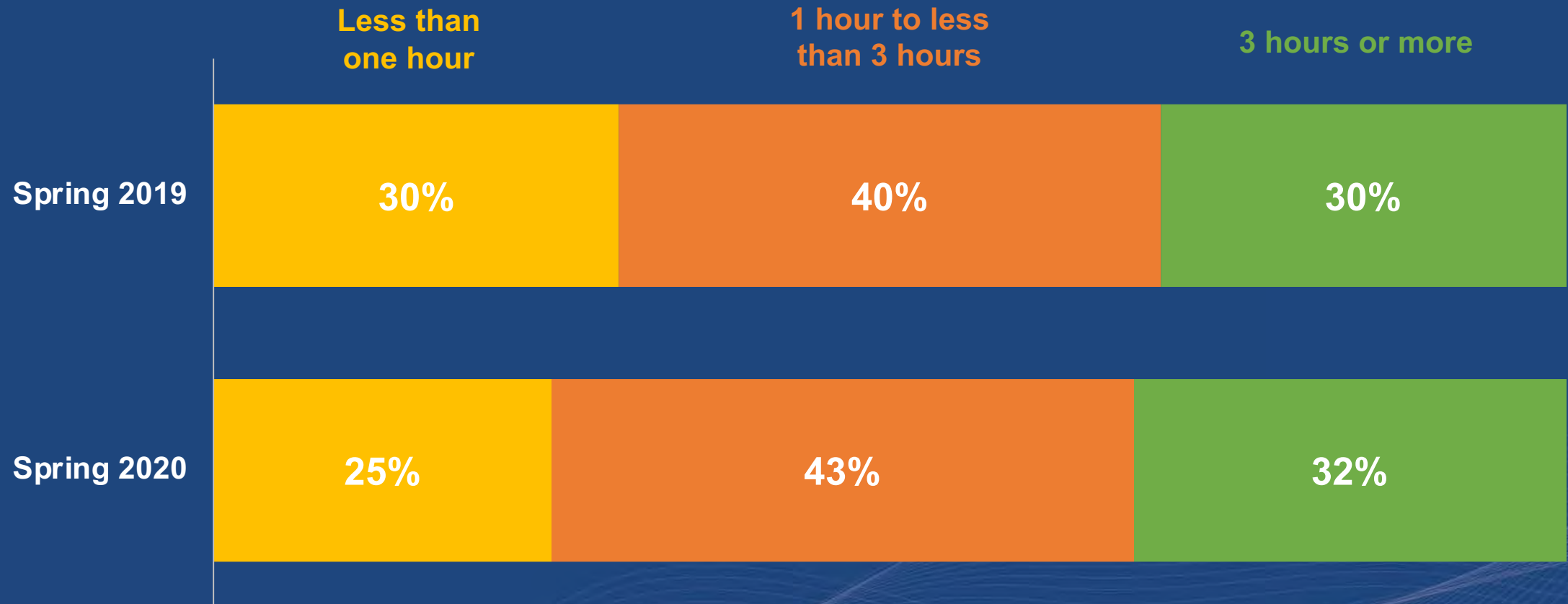


Spring 2020

Base: Own a smart speaker

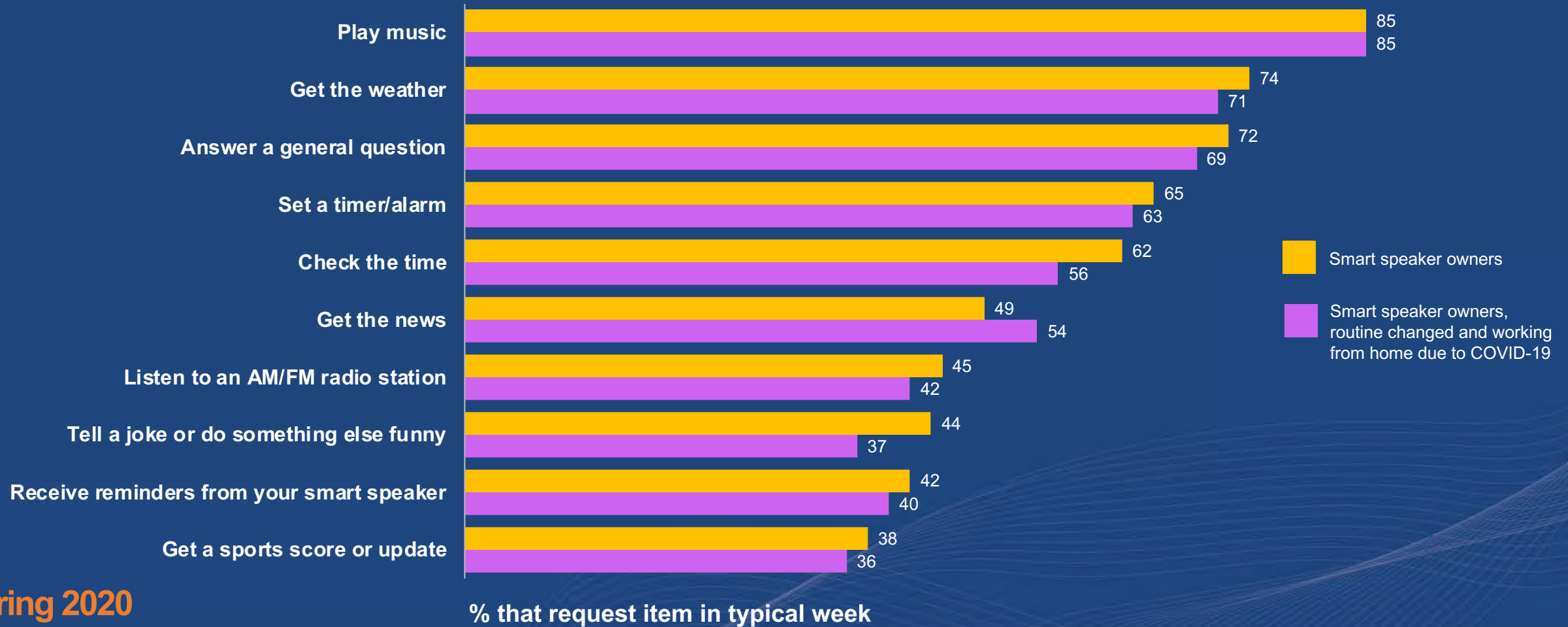
Smart speaker owners

Time spent listening to news on smart speaker in a typical week



Base: Own a smart speaker and listen to news programming on their smart speaker in a typical week

Top ten weekly smart speaker requests



Spring 2020

Base: Smart speaker owners

% who ever listen to news on their smart speaker

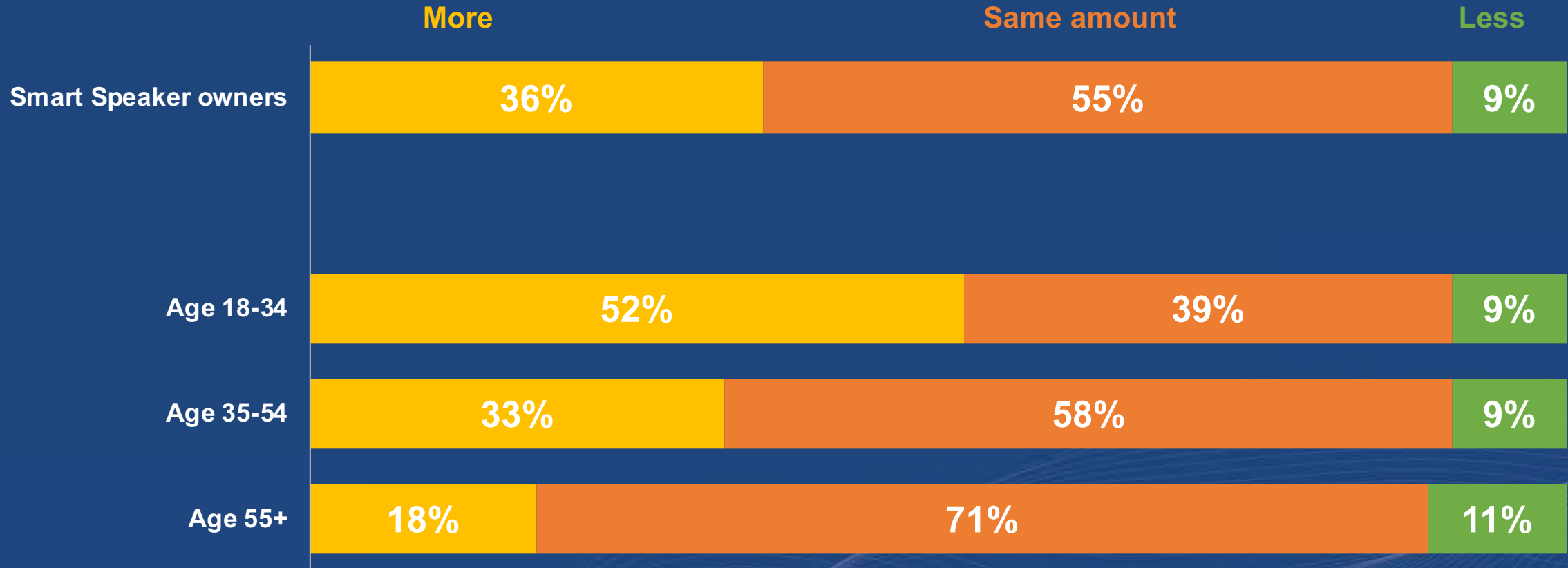


Spring 2020

Base: Own a smart speaker

Smart speaker owners

Compared to before the COVID-19 outbreak, would you say you are using your smart speaker to listen to music and entertainment more, less or about the same?

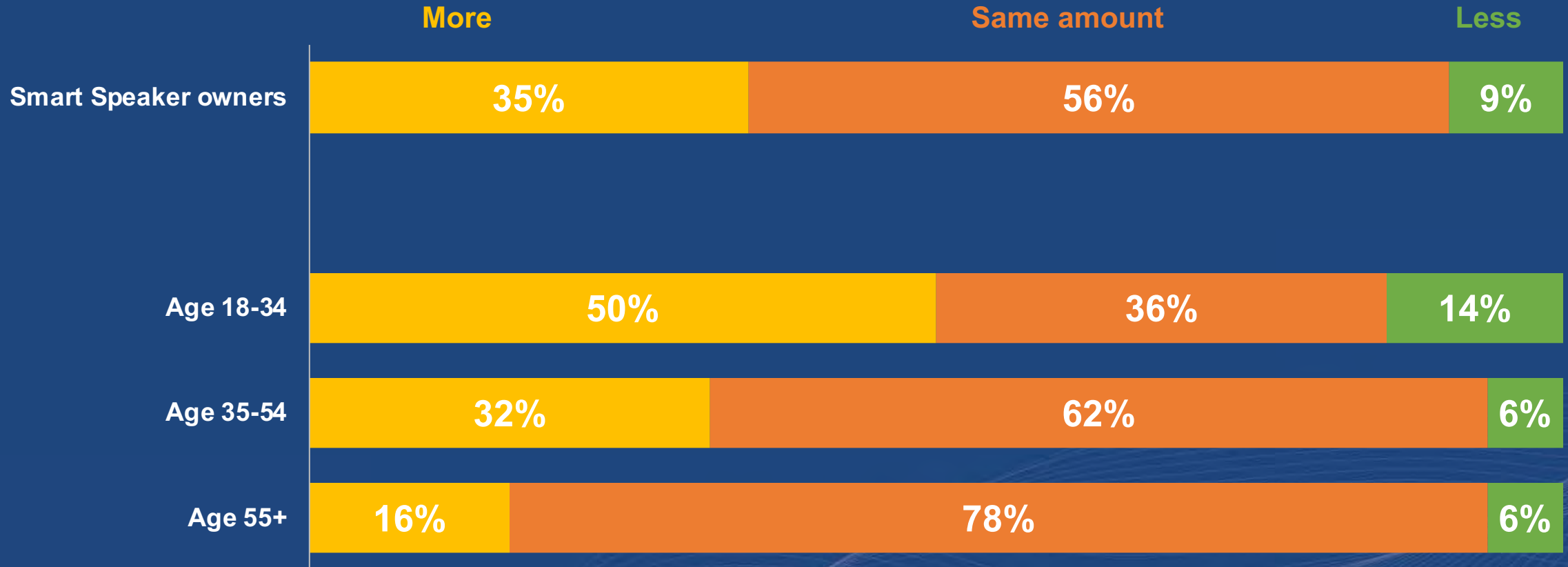


Spring 2020

Base: Own a smart speaker

Smart speaker owners

Compared to before the COVID-19 outbreak, would you say you are using your smart speaker to listen to news and information more, less or about the same?



Spring 2020

Base: Own a smart speaker

Takeaways

- Three-quarters of Americans 18+ say their routines have changed due to COVID-19, and their media habits have changed in response
- News is more important than ever with those who have experienced the most disruption
- Young people in particular are listening to more news as a result of the pandemic, with 50% of 18-34s indicating increased news consumption since the onset of COVID-19

Takeaways

- Voice assistant technology is in a state of evolution, not revolution, as people are using more kinds of skills and using the technology on speakers, TVs, and other devices
 - 63% of the total U.S. online population 18+ say they use voice-operated personal assistants of any kind
- The voice technology in smart speakers is seen as slightly better than the same technology in smartphones, which may be due to operating systems, and environmental factors

Takeaways

- Smart speakers with screens play a crucial role in skill discovery; audio-first devices are perceived as easier to use than audio-only devices
- Among non-owners of smart speakers, those who currently use voice assistants are 60% more likely than those who don't to purchase a smart speaker in the next six months



THE smart audio REPORT

npr.org/smartaudio



Exhibit 6

**UNITED STATES INTERNATIONAL TRADE COMMISSION
WASHINGTON, D.C.**

**Before the Honorable Cameron Elliot
Administrative Law Judge**

**IN THE MATTER OF CERTAIN AUDIO
PLAYERS AND COMPONENTS
THEREOF II**

Investigation No. 337-TA-1330

**DECLARATION OF DAVID V. ANDERSON IN SUPPORT OF COMPLAINANT'S
INITIAL MARKMAN BRIEF**

TABLE OF EXHIBITS

Exhibit No.	Description
1	U.S. Patent No. 9,632,748
2	U.S. Patent No. 9,812,128
3	U.S. Patent No. 11,024,311
4	U.S. Patent No. 11,050,615
5	<i>The Smart Audio Report</i> , NPR (2020)
6	
7	Fernández et al., <i>An Application of Recurrent Neural Networks to Discriminative Keyword Spotting</i> , ICANN 2007
8	U.S. Patent No. 9,318,107
9	Morgan and Bourlard, <i>Continuous Speech Recognition</i> , IEEE Signal Processing Magazine (May 1995)
10	
11	Keshet, <i>Automatic Speech and Speaker Recognition</i> (2009)
12	File History (Excerpts) of U.S. Patent No. 11,024,311

1. I, David V. Anderson, Ph.D., provide the following Declaration. If called to testify as a witness in this Investigation, I would testify competently regarding the matters set forth below.

I. INTRODUCTION

2. I have been retained by counsel for Complainant Google LLC (“Google”) as an expert witness in this litigation to provide opinions concerning certain claim terms in U.S. Patent Nos. 9,632,748 (“the ’748 patent”), 11,024,311 (“the ’311 patent”), and 9,812,128 (“the ’128 patent”). In particular, I have been asked to analyze what a person of ordinary skill in the art would understand certain claim terms to mean at the time of the invention, and to respond to the proposed constructions provided by Google and Respondent Sonos, Inc. (“Sonos”). I understand that Google has asserted claims 1-4, 7, 9-12, 14 and 15 of the ’748 patent, claims 1-15 of the ’128 patent and claims 1-3, 8-12, and 14-20 of the ’311 patent.

3. I am being compensated at my standard billing rate of \$500 per hour for time spent on this matter. My compensation is in no way dependent on the outcome of this matter or the particular testimony or opinions that I provide.

4. The conclusions expressed in this Declaration based on my review to date of the evidence produced at this stage of the case. My opinions are subject to change based on additional opinions that Sonos’s experts may present and information I may receive in the future. I reserve my right to amend or update my conclusions as appropriate in response to any future discovery or developments. With this in mind, based on the analysis I have conducted and for the reasons set forth below, I have preliminarily reached the conclusions set forth in this Declaration.

5. At a hearing I may use as exhibits various documents produced in this case that refer or relate to the matters discussed in this Declaration. I have not yet selected the particular exhibits that might be used. I may also rely on visual aids and may rely on analogies concerning elements of the patents discussed above, the accused products, the references cited in this report,

or any related technologies. In addition, I may create or assist in the creation of certain demonstrative evidence to assist me in testifying, such as demonstrations of devices and software to further support the positions in this Declaration.

II. EXPERT QUALIFICATIONS

6. My full qualifications, including my professional experience and education, can be found in my Curriculum Vitae, which includes a complete list of my publications, and is attached as Appendix A. Appendix A is incorporated into this Declaration in full.

7. I am a professor in the School of Electrical and Computer Engineering at the Georgia Institute of Technology (“Georgia Tech”) in Atlanta, Georgia. I have been a professor at Georgia Tech since 1999. In 2009 I served as a visiting professor in the Department of Computer Science at Korea University in Seoul, South Korea.

8. I received my Ph.D. in Electrical and Computer Engineering from Georgia Tech in 1999. I received my B.S. and M.S. in Electrical Engineering from Brigham Young University in 1993 and 1994, respectively.

9. In my employment prior to Georgia Tech as well as in my subsequent studies and research, I have worked extensively in areas related to research, design, and implementation of signal processing systems and human-centered computing applications. I have also taught graduate and undergraduate level courses at Georgia Tech on signal processing systems, machine learning, computer systems, and multimedia systems. For example, I have taught courses on machine learning for speech, pattern recognition, multimedia processing and systems, software design, and signal processing applications (covering topics in audio processing and speech recognition). I have also designed and taught a course on signal processing in the context of human perception. These courses and my research have covered many topics relevant to the subject matter of the ’748, ’128, ’311 patents and the prior art cited therein.

10. I also have extensive experience with software design, speech recognition systems, multiple microphone systems, processing noisy audio signals, signal detection, information theory, and related topics through my research and consulting.

11. I have served as principal investigator or co-principal investigator in numerous multi-disciplinary research projects including “Improved Speech Analysis, Coding, and Enhancement using Microradars,” “Blind Source Separation for Audio,” “Audio Classification,” “Auditory Scene Analysis,” “Hearing Aid Audio Processing,” “Speaker Driver Sound Enhancement,” “I-Vector Based Voice Quality,” “Analysis of Voice Exercise Using Signal Processing,” and “Smart Homes for Effective and Safe Remote Work During a Pandemic and Beyond.”

12. I have published over 200 book chapters and papers in reviewed journals and conferences. Topics include those such as “Speech recognition using filter bank features,” “Segmentation based speech enhancement using auxiliary sensors,” “A framework for estimation of clean speech by fusion of outputs from multiple speech enhancement systems,” “Distributed acquisition and processing systems for speech and audio,” “A missing data-based feature fusion strategy for noise-robust automatic speech recognition using noisy sensors,” “Learning distances to improve phoneme classification,” “Identification of voice quality variation using i-vectors,” “Varying time-constants and gain adaptation in feature extraction for speech processing,” “Low bit-rate coding of speech in harsh conditions using non-acoustic auxiliary devices,” “Multi-sensor spectro-temporal comb filtering for speech enhancement,” “Evaluation of a hearing compensation algorithm,” “Speech analysis and coding using a multi-resolution sinusoidal transform,” “Biologically inspired auditory sensing system interfaces on a chip,” “Cascade classifiers for audio classification,” and “Single acoustic channel speech enhancement based on glottal

correlation using non-acoustic sensors.” I have also contributed book chapters for treatises such as “Independent Component Analysis for Audio and Biosignal Applications,” and written a book on *Fixed-Point Signal Processing* which is related to the practical implementation of systems for processing sound and other signals.

13. I am the inventor on eight patents, including “Speech activity detector for use in noise reduction system, and methods therefor” (U.S. Patent No. 6,351,731), and “Analog audio signal enhancement system using a noise suppression algorithm” (U.S. Patent No. 7,590,250).

14. I am a Senior Member of the Institute of Electrical and Electronics Engineers (“IEEE”) and have been a Member of the IEEE Since 1991. I am also a Member of the IEEE Signal Processing Society and Communications Society. From 1994 to 2016 I was also a member of the Acoustical Society of America. In 2003, I served as the Co-Chair for the NSF Symposium on Next Generation Automatic Speech Recognition.

15. I have won numerous awards for my work including the Presidential Early Career Award for Scientists and Engineers, being named a US Frontiers of Engineering Fellow by the National Academy of Engineering, and being named a Frontiers of Science Fellow by the National Academy of Science.

III. LEGAL UNDERSTANDING

16. I understand that claim construction is a legal matter for the Administrative Law Judge.

17. I understand that claim terms should be given their ordinary and customary meaning within the context of the patent in which the terms are used, i.e., the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention in light of what the patent teaches.

18. I understand that to determine how a person of ordinary skill would understand a

claim term, one should look to those sources available that demonstrate what a person of skill in the art would have understood disputed claim language to mean. Such sources include the words of the claims themselves, the remainder of the patent's specification, the prosecution history of the patent (all considered "intrinsic" evidence), and "extrinsic" evidence concerning relevant scientific principles, the meaning of technical terms, and the state of the art.

19. I understand that words or terms should be given their ordinary and accepted meaning unless it appears that the inventors were using them to mean something else. This determination is made by consulting the claims, the patent specification, and the prosecution history. Additionally, the specification and prosecution history must be consulted to confirm whether the patentee has acted as their own lexicographer (i.e., provided their own special meaning to any disputed terms), or intentionally disclaimed, disavowed, or surrendered any claim scope. I also understand that the specification acts as a dictionary when it expressly defines claim terms or when it defines claim terms by implication.

20. I am informed that a claim construction analysis must begin and remain centered on the claim language itself. Additionally, the context in which a term is used in the asserted claim can be highly instructive. Likewise, other claims of the patent in question, both asserted and unasserted, can inform the meaning of a claim term. For example, because claim terms are normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims. Differences among claims can also be a useful guide in understanding the meaning of particular claim terms.

21. I understand that a person of ordinary skill in the art is deemed to read a claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification. For this reason, the words of the claim must be

interpreted in view of the entire specification. The specification is the primary basis for construing the claims. Ultimately, the interpretation to be given a term can only be determined and confirmed with a full understanding of what the inventors actually invented and intended to envelop with the claim as set forth in the patent itself.

22. In addition to consulting the specification, I understand that one should also consider the patent's prosecution history. The prosecution file history provides evidence of how both the Patent Office and the inventor(s) understood the terms of the patent, particularly in light of what was known in the prior art. Further, where the specification describes a claim term broadly, arguments and amendments made during prosecution may require a narrower interpretation.

23. I understand that although intrinsic evidence is of primary importance, extrinsic evidence, i.e., evidence external to the patent and prosecution history, including expert opinions, dictionaries, and learned treatises, can also be considered. For example, technical dictionaries may help one better understand the underlying technology and the way in which one of skill in the art might use the claim terms. Extrinsic evidence should not be considered, however, divorced from the context of the intrinsic evidence.

IV. LEVEL OF ORDINARY SKILL IN THE ART

24. I understand that claim interpretation is from the perspective of a person of ordinary skill in the art at the time of the invention.

25. I understand that the '748 patent issued from an application filed on July 29, 2014. I understand that Google has identified July 29, 2014 as the date of invention for the '748 patent claims. For purposes of this declaration, I treat the relevant time of the invention to be July 2014.

26. I am familiar with the knowledge of a person of ordinary skill in the field related to the subject matter of the '748 patent in the July 2014 time period. In forming my opinions, I considered several things, including the various approaches to voice processing employed in the

art, the type of problems encountered, and the rapidity with which innovations were made. I also considered the sophistication of the technology involved, and the educational background and experience of those actively working in the field. Finally, I placed myself back in the relevant time period, and considered the engineers that I had taught and worked with in the industry.

27. I understand that Sonos has proposed that a person of ordinary skill in the art relevant to the '748 patent in the July 2014 time period would have had "[t]he equivalent of a four-year degree from an accredited institution (typically denoted as a B.S. degree) in electrical or computer engineering, or a related field, and approximately 2-4 years of experience in the field of voice and/or audio processing, or an equivalent level of skill, knowledge, and experience." I agree that this level of skill is appropriate for the '748 patent.

28. I met these criteria in the July 2014 time period, and consider myself to have been a person with at least ordinary skill in the art pertaining to the '748 patent at that time. Furthermore, I believe that I am capable at this time of putting myself in the shoes of a person of ordinary skill in the July 2014 time period for purposes of assessing the meaning of the claims of the '748 patent to such an individual.

29. I understand that the '128 and '311 patents are related and share a common specification. I further understand the '311 patent claims priority to the application that issued as the '128 patent, which was filed on October 3, 2016, which claims priority as a "continuation-in-part" to an application filed on April 1, 2015, which in turn claims priority to a "provisional" application filed on October 9, 2014. I understand Google has identified that the dates of invention for the '311 and '128 patent claims are as follows: October 9, 2014 ('128 patent, claims 1-3, 6-8, and 11-13, and '311 patent, claims 1-3, 8-10, 12, 14-18, and 20); April 1, 2015 ('128 patent, claims 4, 9, and 14, and '311 patent, claim 19); and October 3, 2016 ('128 patent, claims 5, 10, and 15,

and '311 patent, claim 11).

30. I am familiar with the knowledge of a person of ordinary skill in the field related to the subject matter of the '128 and '311 patent in the October 2014 to October 2016 time period. In forming my opinion, I considered several things, including the various approaches to voice processing employed in the art, the type of problems encountered, and the rapidity with which innovations were made. I also considered the sophistication of the technology involved, and the educational background and experience of those actively working in the field. Finally, I placed myself back in the relevant time period, and considered the engineers that I had taught and worked with in the industry.

31. I understand that Sonos has proposed that a person of ordinary skill in the art relevant to the '128 and '311 patents at the time of their invention would have had “[t]he equivalent of a four-year degree from an accredited institution (typically denoted as a B.S. degree) in electrical or computer engineering, or a related field, and approximately 2-4 years of experience in the field of voice and/or audio processing, or an equivalent level of skill, knowledge, and experience.” I agree that this level of skill is appropriate for the '128 and '311 patents.

32. I met these criteria in the October 2014 to October 2016 time period, and consider myself to have been a person with at least ordinary skill in the art pertaining to the '128 and '311 patents at that time. Furthermore, I believe that I am capable at this time of putting myself in the shoes of a person of ordinary skill in the October 2014 to October 2016 time period for purposes of assessing the meaning of the claims of the '748 patent to such an individual.

V. BACKGROUND OF THE TECHNOLOGY

33. In this section I provide a summary of certain technologies that relate to the '748, '128, and '311 patents.

34. Voice assistants have become popular in recent years. *See Ex. 5 (The Smart Audio*

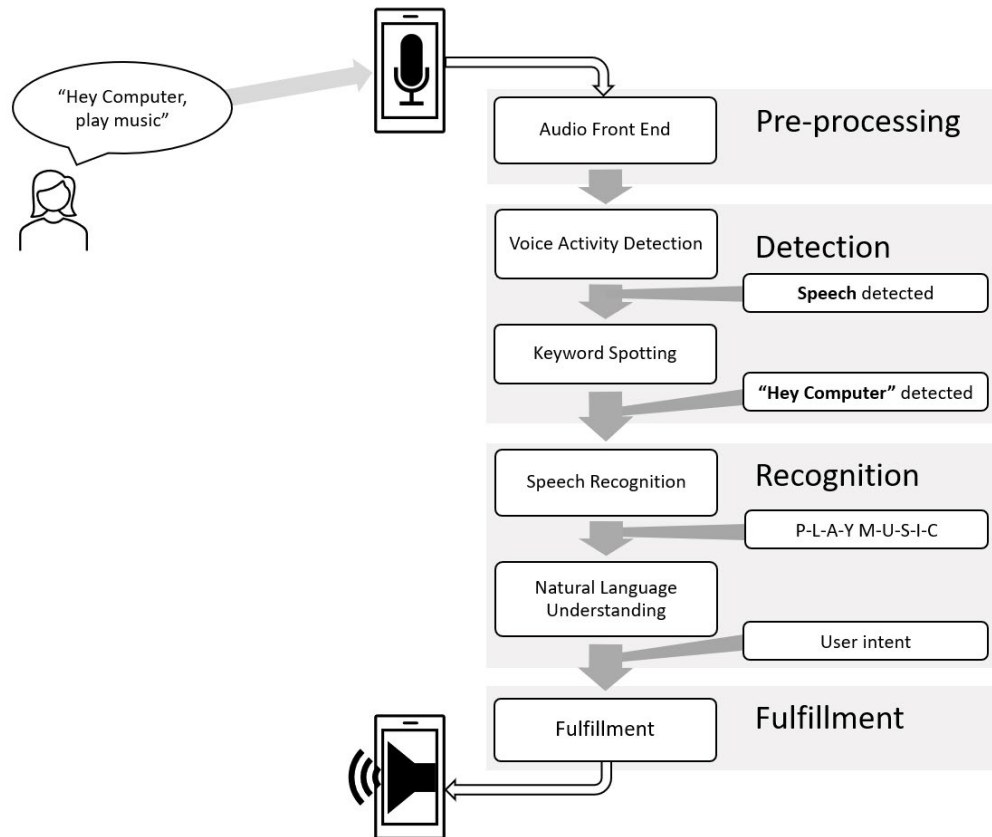
Report, NPR) at 5-6, 16 (highlighting the high adoption of devices supporting voice assistants). In this description, I use the term “voice assistant” to refer to software that generally allows interaction with, and control of, electronic devices via audible phrases and commands.

35. With the increased usage of voice assistant technology on different devices, voice assistants are increasingly used in multiple devices in the same location. For example, a household may have more than one device running a voice assistant application in a room or other area. Adoption of technology often depends on its ease of use, and therefore it is increasingly important that multiple devices in a particular environment coordinate in the execution of their respective voice assistant applications to avoid user confusion. Google’s ’748, ’128, and ’311 patents disclose solutions to enhance the management and coordination of voice control operations across such a multi-device network.

36. Voice assistants and the smart devices that support them process audio in multiple stages, often described as a “pipeline.” As the audio proceeds through the stages in such a pipeline, the computational intensity and processor usage for such processing generally increases. However, not every such pipeline necessarily includes every stage. A voice assistant pipeline typically starts with reception of a user’s utterance of a request by a device’s microphones and concludes with fulfillment of the request by the device that received it or by another component or device at the direction of the receiving device.

37. The stages of the pipeline can be grouped together by function, as illustrated in the figure below. In general, the pipeline will include pre-processing functions (“audio front end” stage), detection functions for detecting the utterance of a voice or keyword (or both) (“voice activity detection” stage and/or “keyword spotting” stage), recognition functions for recognizing and translating the words spoken by the user to text and processing the text to determine the user’s

intent (“speech recognition” stage and “natural language understanding” stage), and fulfillment functions for executing an action requested by a command following the keyword.



38. Pre-processing (Audio Front-End): A voice assistant pipeline typically starts with an “**audio front-end**,” which pre-processes an audio signal received from the device’s microphones order to clean the signal for subsequent stages, such as by suppressing or removing background noise and/or audio that the device itself is outputting via a speaker (which the device microphone will pick up in addition to background noise and voice utterances). Removal of audio that the device itself is outputting is commonly known in the art as acoustic echo cancellation.

39. Detection (Voice Activity Detection / Keyword Spotting): The next stage in a typical voice assistant pipeline involves detection of human speech or specific keywords in the signal. These forms of detection may be used as gating mechanisms for subsequent stages in the voice

processing pipeline, such that subsequent stages are only engaged when a keyword is detected in the audio signal. The asserted '128 and '311 patents further teach that information obtained during the detection stage may be used for “multi-device coordination,” when multiple devices detect a voice input (e.g., multiple devices in a room). The devices use this information to automatically negotiate leadership for interacting with the user, to reduce user confusion. For example, the patents' specification describes this as follows:

A location (e.g., a room or space within a home) may include multiple devices that include voice assistant systems and one or more users of such devices. As the devices are more than ever configured to be capable of accessing and presenting user-specific information, it is desirable for a voice assistant device to be able to handle multiple users gracefully and be able to provide user-specific information to the appropriate user. It is also desirable for there to be a leader amongst the voice assistant devices that is responsible for responding to the user's voice inputs, in order to reduce user confusion.

...

Accordingly, there is a need for an electronic device with a voice assistant system and/or a voice assistant server system that incorporates methods and systems for handling multiple users, for multi-device coordination to determine a leader, and for noise mitigation. The device and/or the server system may be configured to identify a speaker of a voice input based on comparisons to voice models, and to personalize responses to the identified speaker. The device and/or the server system may also be configured to negotiate leadership amongst multiple devices for response to a speaker.

'128 patent, 1:43-2:2.

40. **“Voice activity detection”** (VAD) is a process by which the signal from the audio front-end is checked for the presence of human speech. As a general matter, voice activity detectors continuously analyze an incoming audio signal as it is received, and output a prediction regarding whether speech is present. Voice activity detectors operate specifically to detect presence of speech in a received audio signal, which is a binary decision that does not involve conversion of speech to text or determination of the meaning of the speech. For example, if a user says “Hey Computer, play music,” the voice activity detector would detect the presence of speech

in the received audio signal, but would not determine which words were spoken by the user.

41. “**Keyword spotting**,” as its name suggests, allows for the detection of particular keywords in an audio signal. As described in *An Application of Recurrent Neural Networks to Discriminative Keyword Spotting* (Fernández et al., ICANN 2007), “[t]he goal of keyword spotting is to detect the presence of specific spoken words in (typically) unconstrained speech.” Ex. 7. Keywords are also known in some implementations as a “hotwords” or “wakewords” and may be a short phrase (e.g., “Hey Google” or “Hey Sonos”). Keywords can be used to signify a user’s desire to activate the voice assistant functionality of a smart device. For example, the ’128 patent describes the use of hotwords and wakewords for voice assistant devices:

In some implementations, an electronic device 190 is “awakened” (e.g., to activate an interface for the voice assistant service on the electronic device 190, to put the electronic device 190 into a state where the electronic device 190 is ready to receive voice requests to the voice assistant service) by a voice input that includes a hotword (also called a “wake word”). In some implementations, the electronic device 190 requires awakening if the electronic device 190 has been idle with respect to receipt of voice inputs for at least a predefined amount of time (e.g., 5 minutes); the predefined amount of time corresponds to an amount of idle time allowed before a voice interface session or conversation times out. The hotword may be a word or phrase, and may be a predefined default and/or may be customized by a user (e.g., a user may set a nickname for a particular electronic device 190 as the device’s hotword). In some implementations, there may be multiple hotwords that can awaken an electronic device 190. A user may speak the hotword, wait for an acknowledgement response from the electronic device 190 (e.g., the electronic device 190 outputs a greeting), and then make a first voice request. Alternatively, the user may combine the hotword and the first voice request in one voice input (e.g., the voice input includes the hotword followed by the voice request).

’128 patent at 11:44-67. U.S. Patent No. 9,318,107, which is incorporated by reference into the ’128 and ’311 patents further describes uses of hotwords and methods for detecting hotwords:

For a speech-enabled system, the users’ manner of interacting with the system is designed to be primarily, if not exclusively, by means of voice input. Consequently, the system, which potentially picks up all utterances made in the surrounding

environment including those not directed to the system, must have some way of discerning when any given utterance is directed at the system as opposed, e.g., to being directed at an individual present in the environment. One way to accomplish this is to use a hotword, which by agreement among the users in the environment, is reserved as a predetermined word that is spoken to invoke the attention of the system. In an example environment, the hotword used to invoke the system's attention are the words "OK computer." Consequently, each time the words "OK computer" are spoken, it is picked up by a microphone, conveyed to the system, which performs speech recognition techniques to determine whether the hotword was spoken and, if so, awaits an ensuing command or query. Accordingly, utterances directed at the system take the general form [HOTWORD] [QUERY], where "HOTWORD" in this example is "OK computer" and "QUERY" can be any question, command, declaration, or other request that can be speech recognized, parsed and acted on by the system, either alone or in conjunction with the server via the network.

When the user 102 speaks the utterance 104, "OK computer," each computing device that has a microphone in the vicinity of the user 102 detects and processes the utterance 104. Each computing device detects the utterance 104 through an audio input device such as a microphone. Each microphone provides audio data to a respective audio subsystem. The respective audio subsystem buffers, filters, and digitizes the audio data. In some implementations, each computing device may also perform endpointing and speaker identification on the audio data. The audio subsystem provides the processed audio data to a hotworder. The hotworder compares the processed audio data to known hotword data and computes a confidence score that indicates the likelihood that the utterance 104 corresponds to a hotword. The hotworder may extract audio features from the processed audio data such as filterbank energies or mel-frequency cepstral coefficients. The hotworder may use classifying windows to process these audio features such as by using a support vector machine or a neural network. Based on the processing of the audio features, the hotworder 124 computes a confidence score of 0.85, hotworder 126 computes a confidence score of 0.6, and hotworder 128 computes a confidence score of 0.45. In some implementations, the confidence score may be normalized to a scale of zero to one, with a higher number indicating a greater confidence that the utterance 104 includes a hotword.

Ex. 8 (U.S. Patent No. 9,318,107) at 1:47-2:3, 5:2-28.

42. Modern approaches to keyword spotting rely on the use of neural networks, which are machine learning configurations comprised of a number of individual nodes patterned after the neurons of the human brain. Prior to use, the neural network is "trained" on the specific keywords of interest. After the neural network has been trained, it can be deployed for use in predicting

whether an incoming audio signal contains a keyword, based on characteristics of the audio signal.

43. Keyword spotters operate specifically to detect presence of a keyword in a received audio signal; it is a binary decision. They do not determine the presence of other words in the audio signal, nor do they convert speech in the audio signal to text or seek to determine what is meant by the keyword. For example, if a user says “Hey Computer, play music” the keyword spotter might detect presence of the keyword (“Hey Computer”) in the received audio signal, but it would not detect other words that are not keywords.

44. Recognition (Speech Recognition and Natural Language Understanding): After the voice or keyword is detected by the preceding detection stage, the next step is to recognize the complete set of words that are spoken after the keyword, converting those words into text (“speech recognition”), and determine what they mean within the context of the system (“natural language understanding”). In other words, the recognition function is directed to more detailed processes that enable the voice assistant to determine text corresponding to the words spoken by the user, and determining from these words an intent associated with the execution of a function by a device.

45. “**Speech recognition**,” also called “automatic speech recognition,” involves the recognition of words in the signal and conversion of these words to text. In other words, “[t]he basic task of *Automatic Speech Recognition* (ASR) is to derive a sequence of words from a stream of acoustic information.” Ex. 9 (Morgan and Bourlard, *Continuous Speech Recognition*, IEEE Signal Processing Magazine (May 1995)). Speech recognizers first employ an acoustic model to evaluate a received audio signal against thousands of known sequences of “phonemes” (units of sound that distinguish one word from another), to determine the best candidate textual words for each spoken word. The speech recognition system then uses a language model to find the most likely word sequence from these candidates. This is done by evaluating the likelihood of each

possible sequence of words against a model that indicates the likelihood that words will appear together in a given context—for example, the likelihood that the words “plan to wait” will appear in a sentence, as opposed to “plant weight.” Following the example described above, if a user says “play music” after the keyword, the speech recognizer transcribes that utterance into text for further processing by subsequent stages in the voice assistant pipeline.

46. Speech recognition is more computationally expensive, requiring more memory and processing resources, than the keyword detection and voice activity detection technologies described above. For this reason, in certain implementations, the voice assistant pipeline may be designed so that speech recognition does not begin until a keyword has been detected.

47. **“Natural language understanding”** includes analysis of the recognized text corresponding to the user’s utterance to determine what the user wants the voice assistant to do, often referred to as an “intent” (e.g., a request to play music). This stage must handle variations on a request (e.g., “play some music,” “play some tunes,” “play Beethoven’s ninth”) and may be quite sophisticated. The variety of ways in which a request may be presented by the user further illustrates why speech recognition and natural language understanding stages are more computationally expensive than the voice activity detection and keyword spotting stages.

48. **Fulfillment**: The final stage in the voice assistant pipeline is fulfillment. At this stage, an action to be taken is determined based on the user’s intent determined from the original audio signal and potentially also based on additional information that the voice assistant receives in response to a request to the user for clarification. This stage also may include a determination by the voice assistant as to which device, or combination of devices, should fulfill the user’s request. For example, the ’128 and ’311 patents describe that a user might make a voice request on one device for playback of content on device that is different from the one that received the

voice request:

A user can issue a media play request via the microphone of electronic device 190, and ask the voice assistant service to play media content on the electronic device 190 itself or on another connected media output device. For example, the user can issue a media play request by saying to the Wi-Fi speaker “OK Google, play cat videos on my Living room TV.” The voice assistant service then fulfills the media play request by playing the requested media content on the requested device using a default or designated media application.

’128 patent, 9:34-43.

49. Thus, the user interacts with a first device and receives responses from that device. For example, the ’128 and ’311 patents describe that those responses can include requests from the voice assistant for clarification from the user, or a confirmation from the voice assistant to the user that action is being carried out (e.g., a confirmation that music will be played on another device).

Further, in some implementations, a user may issue, via the microphone of electronic device 190, a request for information or for performance of an action or operation. . . . The electronic device 190 responds to the request with voice message responses to the user, where the response may include, for example, requests for additional information to fulfill the request, confirmation that the request has been fulfilled, notice that the request cannot be fulfilled, and so forth.

’128 patent, 10:2-20, 24:61-25:7. The voice assistant then fulfills the user’s request by playing the requested content on the device.

VI. CONCLUSIONS REGARDING CLAIM TERMS IN THE ’748 PATENT

A. “[perform/performing] speech recognition” (All Claims)

50. Each of the independent claims of the ’748 patent recites “[perform/performing] speech recognition,” as follows:

[Claim 1] sending, by the first computing device and to at least a second computing device from the plurality of computing devices, a command to temporarily refrain from performing speech recognition;

determining, by the first computing device, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a predetermined spoken audio command; and

[Claims 7 and 11] send, to at least a second computing device from a plurality of computing devices, a command to temporarily refrain from performing speech recognition;

determine, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a predetermined spoken audio command; and

'748 patent, claims 1, 7, 11

51. I understand that the parties have offered the following competing constructions for this term:

Complainant's Proposed Construction	Respondent's Proposed Construction
No construction necessary (plain and ordinary meaning), i.e., "transforming speech to corresponding text"	"processing spoken audio to determine whether it includes a predetermined spoken audio command"

52. A person of ordinary skill in the art would understand that the term "speech recognition" had a well-understood meaning in the art and referred to a process for transforming speech to corresponding text. This well-understood meaning is reflected in the literature in the field of speech processing. For example, *Continuous Speech Recognition* explains that "[t]he basic task of Automatic Speech Recognition (ASR) is to derive a sequence of words from a stream of acoustic information." Ex. 9 at 3-4. As another example, *Automatic Speech and Speaker Recognition* explains that the goal of "automatic speech recognition" is "to transcribe a recorded speech utterance into its corresponding sequence of words." Ex. 11 at 3.

53. A person of ordinary skill in the art would also understand based on the specification that the '748 patent uses the term "speech recognition" consistent with its ordinary meaning of transforming text to speech. The '748 patent states that the claimed computing devices' "audio input modules may be operable to utilize *speech recognition techniques to*

determine text from obtained audio input.” ’748 patent, 4:57-59 (emphasis added).

54. As I understand Sonos’s proposed construction, it would encompass voice activity detection and keyword (hotword detection) as well as ***any*** processing “spoken audio to determine whether it includes a predetermined spoken audio command.” Such a construction is inconsistent with what a person of ordinary skill in the art would understand “speech recognition” to mean in the context of the claims and specification of the ’748 patent. Sonos’s proposed construction of “speech recognition” appears to include any other kind of processing, including keyword or hotword detection, which are different technologies than speech recognition. Keyword or hotword detection is, generally speaking, a process whereby a system distinguishes between utterances in which a given keyword is spoken and utterances where that keyword is not spoken. *See, e.g.,* Ex. 11 (Keshet, Automatic Speech and Speaker Recognition (2009)) at 13.

55. A person of ordinary skill in the art would understand based on the specification of the ’748 patent that “speech recognition” as recited in the claims does not include keyword or hotword detection. The specification unequivocally describes speech recognition as a technique ***“to determine text from obtained audio input.”*** ’748 patent, 4:57-59. And the ***only*** time the ’748 patent uses the term “hotword” is in the context of monitoring for a ***predetermined audio command***, not speech recognition. *See* ’748 patent, 2:66-67.

56. Further, a person of ordinary skill in the art would understand that the specification distinguishes between two ways to “process” audio input. The specification states “[i]n some examples, monitoring and execution module 38 may process the audio input by performing ***one or more speech-to-text transformations to obtain textual data corresponding to the received audio input.*** In other examples, monitoring and execution module 38 may perform other audio analysis to recognize and identify speech, ***such as performing frequency analysis to identify tonal***

characteristics or other sonic identifiers present in the audio input.” ’748 patent, 15:30-38 (emphasis added). A person of ordinary skill in the art would understand that the first example refers to speech recognition, and the second refers to voice activity detection (i.e., whether the audio received by the device was a voice).

57. As I understand Sonos’s construction, it would encompass both of these ways to “process” audio input by the claimed computing devices, but only the former is “speech recognition.” Thus, a person of ordinary skill in the art would understand that Sonos’s construction is overbroad.

58. Finally, a person of ordinary skill in the art would recognize that Sonos’s proposed construction of “speech recognition” would render the claim language redundant with other recited claim elements. For example, claim 1 includes the limitation “determining, by the first computing device, based on *performing speech recognition* on the spoken audio input, *whether the spoken audio input includes a predetermined spoken audio command.*” ’748 patent, cl. 1. (emphasis added). If “performing speech recognition” meant “processing spoken audio to determine whether it includes a predetermined spoken audio command,” then it would be redundant to say that the computing device “determin[es] ... whether the spoken audio input includes a predetermined spoken audio command” “based on performing speech recognition.”

**B. “a command to temporarily refrain from performing speech recognition”
(All Claims)**

59. Each of the independent claims of the ’748 patent recites “a command to temporarily refrain from performing speech recognition,” as follows:

[Claim 1] sending, by the first computing device and to at least a second computing device from the plurality of computing devices, a command to temporarily refrain from performing speech recognition;

[Claims 7 and 11] send, to at least a second computing device from a plurality of computing devices, a command to temporarily refrain from performing speech recognition;

60. I understand that the parties have offered the following competing constructions for this term:

Complainant's Proposed Construction	Respondent's Proposed Construction
No construction necessary (plain and ordinary meaning)	"a message instructing the second computing device to temporarily cease performing any and all speech recognition"

61. A person of ordinary skill in the art reading the '748 claims in the context of the claims and the specification would understand the plain and ordinary meaning of this term. A person of ordinary skill would understand that the claim language uses common English words and employs them using their standard meanings. Each independent claim requires, "responsive to determining that the first computing device should perform speech recognition on the spoken audio input," "send[ing], [by the first computing device and] to at least a second computing device from a plurality of computing devices, a command to temporarily refrain from performing speech recognition." '748 patent, cls. 1, 7, 11. Under their plain meaning, the claims require that the claimed first computing device send a message to at least the second computing device that at least the second computing device should not perform speech recognition for a limited amount of time.

62. A person of ordinary skill in the art would further that the claim term should be given its plain and ordinary meaning based on the specification of the '748 patent. The specification explains the meaning of the term in the context of the claims. The '748 patent explains that, "[i]n general, techniques of the present disclosure may enable one or more computing devices from among a plurality of computing devices to designate a particular computing device to monitor for and/or act on audio input" '748 patent, 2:36-41. Once that

device is designated, the other devices may “temporarily cease to monitor for audio input. *Id.* at 2:52-57. This can be accomplished in several different ways, one of which is an instruction or command. Specifically, the designated computing device can send instructions (a command) to the other, non-designated computing devices to temporarily cease monitoring for audio input. *Id.* at 9:36-41 (“In some examples, the designated particular computing device may output instructions to computing devices other than the particular computing device to cause the respective computing devices other than the particular computing device to temporarily deactivate respective audio input modules.”).

63. Sonos’s proposed construction substitutes some words of the claim and adds others. Specifically, Sonos changes “command” to “message instructing,” “refrain” to “cease,” and “speech recognition” to “any and all speech recognition.” A person of ordinary skill in the art would easily understand the plain meaning of the claim language itself, so adding and substituting claim terms would provide no clarity to a person of ordinary skill in the art and could only distort the meaning of the claims.

C. “predetermined spoken audio command” (All Claims)

64. Each of the independent claims of the ’748 patent recites “a predetermined spoken audio command,” as follows:

determining, by the first computing device, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a predetermined spoken audio command; and

responsive to determining that the spoken audio input includes the predetermined spoken audio command, executing, by the first computing device, an action associated with the predetermined spoken audio command.

65. I understand that the parties have offered the following competing constructions for this term:

Complainant’s Proposed Construction	Respondent’s Proposed Construction

No construction necessary (plain and ordinary meaning)	“any audio input usable to initiate interaction with a computing device, such as a spoken hotword”

66. A person of ordinary skill in the art would understand that the term “predetermined spoken audio command” has a plain and ordinary meaning in the context of the claims and the specification. In this context, “predetermined spoken audio command” means a spoken utterance that includes a directive for the claimed computing device to execute an associated action .

67. A person of ordinary skill in the art would understand that the claims require the “predetermined spoken audio command” to include a directive that can be executed by the device. Each of independent claims 1, 7, and 10 recites “determin[e/ing], by the first computing device, based on performing speech recognition on the spoken audio input, whether the spoken audio input includes a ***predetermined spoken audio command***; and responsive to determining that the spoken audio input includes the predetermined spoken audio command, ***executing***[, by the first computing device], ***an action associated with the predetermined spoken audio command.***” ’748 patent, cls. 1, 7, 10.

68. The specification provides further evidence that would lead a person of ordinary skill in the art to conclude that the spoken audio must include a directive for the claimed computing device to execute an associated action. The ’748 patent ties predetermined audio commands with corresponding actions throughout the specification. For example, the ’748 patent states that “[i]n some cases, a computing device may monitor an environment for audio input that includes a predetermined audio command and, ***if the predetermined audio command is detected, perform a corresponding action,***” that “one or more of audio input modules 6 may ***associate a predetermined audio command with an action to be performed*** by the respective computing device.” ’748 patent at 1:15-18; 4:8-10. The specification provides numerous examples of predetermined spoken audio

commands and their associated actions. The '748 patent explains that “[o]ne or more of audio input modules 6 may associate a predetermined audio command with an action to be performed by the respective computing device.” ’748 patent at 4:8-10. For example, predetermined audio commands can include “answer” (to answer a telephone call), “navigation” or “directions” (to request directions), “send email,” “send text message,” and other directive phrases. *Id.* at 4:14-21. Rather than simply “directions,” the phrase “[g]ive me directions” may also be a predetermined spoken audio command. *Id.* at 9:61-62 (“The user may speak a predetermined audio command (e.g., ‘Give me directions’).”). In another example, “the audio input includes the predetermined audio command ‘what’s the time.’” *Id.* at 20:15-18. “Go silent” is another example. *Id.* at 24:36-39. The specification also provides an example where “the predetermined audio command may be the word ‘play’ and the subsequent audio input may be the title of a song, TV show, or other media content item.” *Id.* at 24:41-46.

69. As I understand Sonos’s proposed construction, it is directed to “initiat[ing] interaction” with a computing device, and would therefore does not include the requirement that the predetermined spoken audio command actually include a **command**, i.e., an instruction to execute an action, as recited in the asserted claims. A person of ordinary skill in the art would understand that such a reading is inconsistent with the claims and specification. Further, Sonos’s proposed construction would seemingly exclude audio inputs other than a hotword or similar construction used to “initiate interaction.” The claims and specification make clear that the claim language encompasses predetermined audio commands associated with a variety of actions to be carried out by the device beyond just entry into an audio input mode (as would be the case with a hotword), such as answering telephone calls and sending messages. *See, e.g.,* ’748 patent, 4:8-30. Therefore a person of ordinary skill in the art would not understand “predetermined spoken audio

command” to be limited as Sonos suggests.

D. “[determine/determining] ... based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input” (All Claims)

70. Each of the independent claims of the ’748 patent recites “[determine/determining] ... based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input” as follows:

determining, by the first computing device and based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input; and (claims 1)

determine, based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input (claims 7, 11)

71. I understand that the parties have offered the following competing constructions for this term:

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“determining whether the first computing device should perform speech recognition on the spoken audio input without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command”

72. A person of ordinary skill in the art would understand the plain and ordinary meaning of this term based on the claims and the specification of the ’748 patent. For example, the specification explains that “one or more computing devices from a plurality of computing devices may designate a particular computing device from among the plurality of computing devices to process audio input to determine whether the audio input includes a predetermined audio command.” ’748 patent at 6:61-66. The computing devices may determine which of them should

be designated as the “particular computing device” using an arbitration process, or each can make the decision individually. *Id.* at 7:30-36. The specification states that the determination can be “based at least in part on” a variety of criteria, such as a configured hierarchy based on type of device (7:37-64), audio input quality of each device’s environment (7:65-8:13), the number of predetermined audio commands each device is capable of handling (8:14-25), the “context” of each device such as location, the time, applications installed on or executing on the device, or other contextual information (8:26-52); or other information (8:53-62). The claim language is consistent with these examples.

73. Further, a person of ordinary skill in the art would not understand the claim term to mean, as Sonos contends, “determining whether the first computing device should perform speech recognition on the spoken audio input without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command.” This construction replaces the latter half of the claim term “based at least in part on information received from the one or more additional computing devices” with “without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command.” A person of ordinary skill in the art would understand that those two phrases do not mean the same thing in the context of the ’748 claims and specification (or in any context). A person of ordinary skill in the art would understand that such a construction would change the meaning of the claim in two ways. First, it would remove the requirement that the determination of “whether the first computing device should perform speech recognition” be “based at least in part on information received from the one or more additional computing devices.” Second, it would add a requirement that the “determination” of whether the first computing device perform speech recognition “be done without any “processing” of the spoken audio input to

determine whether it includes a predetermined audio command. Both changes are unsupported by the claims and specification of the '748 patent.

74. A person of ordinary skill in the art would further understand that Sonos's construction narrows the scope of the claims by adding a negative limitation that the "determining" step be performed "without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command." A person of ordinary skill in the art would understand that this negative limitation is not supported by the claims or the specification and that the claims could be practiced whether or not the non-designated computing devices process the spoken audio input to determine whether it includes a predetermined audio command." This is because nothing in the claims or specification precludes the non-designated computing devices from determining whether the spoken audio input includes a predetermined audio command before the "determining . . . whether the first computing device should perform speech recognition" at issue. To the contrary, the specification states that each of the computing devices may monitor the environment and perform some processing on the received audio input whether or not that device is designated. For example, the '748 patent explains that where the designation decision is based on "a level of audio input quality associated with one or more of computing devices 2," "one or more of computing devices 2 may obtain audio input from the environment . . . and communicate with others of computing devices 2 to compare the quality of the audio input and/or of speech included therein." *Id.* at 8:2-7. Determining and communicating the quality of an audio input may require some processing of the audio input, and nothing in the specification states that the associated processes could not include determining whether the audio input includes a predetermined audio command.

75. A person of ordinary skill in the art would further understand that the claims are

directed to and focused on issuing an instruction to not perform speech recognition, not **how and when** that device operates in response to the instruction. As explained above with respect to the term “a command to temporarily refrain” in Section VI.B, in some embodiments the designated device may send a command to the other devices to refrain from performing speech recognition on the spoken audio input. When the designated device sends an instruction to the non-designated devices to “temporarily deactivate respective audio input modules,” the ’748 patent does not specify that the deactivation must be immediate or that it must occur before any of the spoken audio input is processed. *Id.* at 9:36-50. Accordingly, a person of ordinary skill in the art would understand that there is no basis to construe the claims to require that the determination whether to perform speech recognition must be made “without any of the plurality of computing devices processing the spoken audio input to determine whether it includes a predetermined audio command.”

VII. CONCLUSIONS REGARDING CLAIM TERMS IN THE ’128 AND ’311 PATENTS

A. “other devices of the plurality of electronic devices” / “quality scores generated by the other devices of the plurality of electronic devices” (’128 All Claims)

76. The terms “other devices of the plurality of electronic devices” and “quality scores generated by the other devices of the plurality of electronic devices” appear in each of independent claims 1, 6, and 11 of the ’128 patent. Each of the independent claims contains the same following limitations:

communicating the quality score to the other devices of the plurality of electronic devices;

receiving quality scores generated by the other devices of the plurality of electronic devices for detection of the voice input by the other devices;

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received

quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input; and

'128 patent, claims 1, 6, 11 (emphasis added).

77. I understand that the parties have offered the following competing constructions for these terms:

Complainant's Proposed Construction	Respondent's Proposed Construction
No construction necessary (plain and ordinary meaning)	"other devices" means "two or more other devices"; and "quality scores" means "two or more quality scores"

78. A person of ordinary skill in the art reading the '128 claims in the context of the specification would understand the plain and ordinary meaning of these terms, and would understand it to be that "other devices" refers to one or more devices" and "quality scores" refers to one or more "quality scores."

79. First, a person of ordinary skill would understand this to be the case based on the language of the claims themselves. The independent claims each recite "a first electronic device of a plurality of electronic devices." A person of ordinary skill would understand "plurality" to mean "at least two" and thus that a "plurality of electronic devices" to mean two or more electronic devices, one of which is the recited "first electronic device." A person of ordinary skill would therefore understand that "other devices of the plurality of electronic devices" may refer to one or more "other devices." In other words, a person of ordinary skill in the art would understand the independent claims to encompass a two-device configuration comprising the "first electronic device" and one "other electronic device." Based on this understanding, a person of ordinary skill in the art would further understand that the "quality scores generated by the other devices of the plurality of electronic devices" may refer to one or more "quality scores." For example, there need be only one "other device" and one "quality score" received from that "other device" in order to

meet the claim limitations.

80. Second, a person of ordinary skill would understand that the '128 specification confirms that the claimed “other devices” and “quality scores” can refer to a single device and score in a two-device system. For example, the specification describes Figure 6 as follows:

FIG. 6 is a flow diagram illustrating a method 1200 of negotiating device leadership among multiple voice interface devices, in accordance with some implementations. The method 1200 is implemented at an electronic device (e.g., device 1002-1, 1002-2, or 1002-3) of **two or more electronic devices** 190 (electronic devices 1002), where each of the **two or more electronic devices** includes one or more microphones, a speaker, one or more processors and memory storing one or more programs for execution by the one or more processors.

'128 patent, Fig. 6, 28:63-29:5. Figure 6, shown below, tracks the elements of the independent claims:

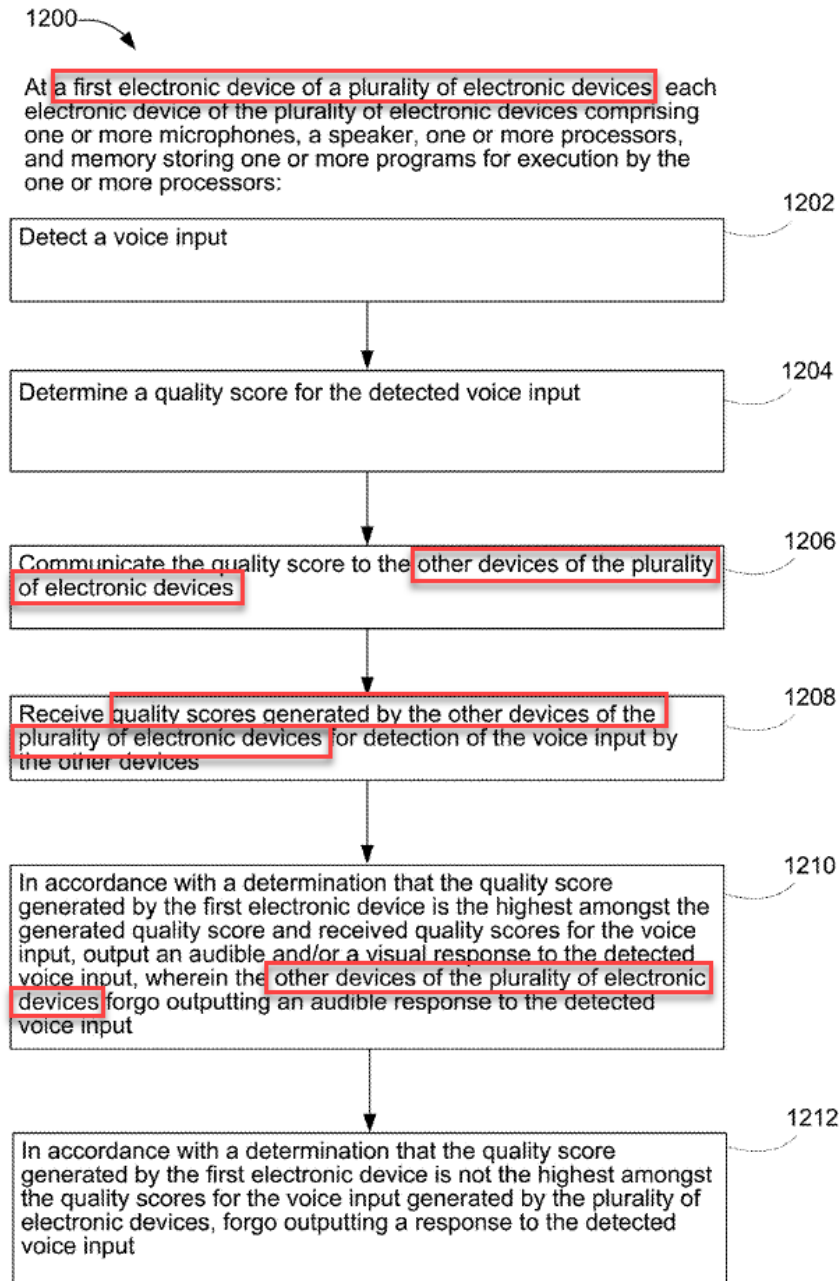


Figure 6

81. A person of ordinary skill would understand, based on the above description of Figure 6 in the specification, that the “plurality of electronic devices” referenced in Figure 6 is comprised of “two or more electronic devices.” ’128 patent, Fig. 6, 28:63-29:5. For the same

reasons described above in relation to the claims, the person of ordinary skill would therefore also understand that the “other devices” in the plurality of electronic devices described in steps 1206, 1208, and 1210 of Figure 6 refers to one or more devices, and “quality scores” received from “other devices” described in step 1208 in Figure 6 refers to one or more “quality scores.” A person of ordinary skill in the art would understand the independent claims of the ’128 patent in the context of Figure 6 and the associated description, and would interpret the claims to be consistent with it.

82. For all the reasons described above, a person of ordinary skill in the art would not understand these terms as proposed by Sonos, which would exclude two-device configurations from the claimed “plurality of electronic devices” (by requiring two or more “other devices” in addition to the “first electronic device” of the “plurality of electronic devices”).

B. “second values ... from other electronic devices” / “second values from other electronic devices of the plurality of electronic devices” (’311 All Claims)

83. The terms “second values ... from other electronic devices” and “second values from other electronic devices of the plurality of electronic devices” appear in claims 1, 10, and 16 of the ’311 patent as follows:

receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices (claims 1 and 10)

receive respective second values from other electronic devices of the plurality of electronic devices (claim 16)

’311 patent, cls. 1, 10, 16 (emphasis added).

Complainant’s Proposed Construction	Respondent’s Proposed Construction
No construction necessary (plain and ordinary meaning)	“second values” means “two or more second values”; and “electronic devices” means “two or more electronic devices”

84. Independent claims 1 and 10 of the ’311 patent recite “a first electronic device of a plurality of electronic devices.” Independent claim 16 of the ’311 patent recites “a plurality of

electronic devices.” A person of ordinary skill would understand “plurality” to mean two or more electronic devices, one of which is the “first electronic device” recited in claims 1 and 10.

85. For the same reasons I described above in relation to the ’128 patent, a person of ordinary skill in the art reading the ’311 claims in the context of the specification would understand the plain and ordinary meaning of these terms, and would understand it to be that “other devices” may refer to one or more devices and “second values” may refer to one or more “second values.”

86. First, a person of ordinary skill would understand this to be the case based on the language of the claims themselves. Independent claims 1 and 10 each recite “a first electronic device of a plurality of electronic devices.” A person of ordinary skill would understand based on use of the term “plurality” that this refers to two or more electronic devices, one of which is the recited “first electronic device.” In claim 16, a person of ordinary skill would also understand that the “plurality” of electronic devices means two or more devices, one of which is the “device” that “determines a first value” and one or more of which is the “other device” from which a “respective second value” is received.

87. A person of ordinary skill would therefore understand that “other devices of the plurality of electronic devices” may refer to one or more “other devices” in the plurality. In other words, as with the claims of the ’128 patent, a person of ordinary skill in the art would understand the independent claims of the ’311 patent to encompass a two-device configuration. Based on this understanding, a person of ordinary skill in the art would further understand “second values” received from the other devices of the plurality of electronic devices to refer to one or more “second values,” one for each of the one or more “other devices.”

88. Second, a person of ordinary skill would understand that the same portions of the ’311 specification described above in relation to the ’128 patent confirms that the claimed

“other devices” and “second values” can refer to a single device and value in a two-device system, for the same reasons I describe above for the ’128 patent. ’311 patent, Fig. 6, 29:8-17.

89. For the same reasons described above, a person of ordinary skill in the art would not understand these terms as proposed by Sonos, which would exclude two-device configurations from the claimed “plurality of electronic devices” (by requiring two or more “other devices” in addition to the “first electronic device” of the “plurality of electronic devices”).

90. For all the reasons described above, a person of ordinary skill in the art would not interpret these terms in the manner proposed by Sonos, which would exclude two-device configurations from the claimed “plurality of electronic devices” (by requiring three or more devices in the “plurality of electronic devices”).

**C. “forgo outputting an audible response” / “forgoing outputting a response”
(’128 All Claims) / “[forgoing / forgo] responding” (’311 Claims 2, 3, 12)**

91. The terms “forgo outputting an audible response” and “forgoing outputting a response” appear claims 1, 6, and 11 of the ’128 patent. Each of the independent claims contains the same following limitations:

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input;

in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic device, forgoing outputting a response to the detected voice input.

92. The term “[forgoing / forgo] responding” appears in claims 2, 3, and 12 of the ’311 patent as follows (I have included the independent claims for context):

1. A method performed at a first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more

microphones, one or more processors, and memory storing one or more programs for execution by the one or more processors, the method comprising:

detecting a voice input from a user;

determining a first value associated with the voice input;

receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices; and

in accordance with a determination that the first value is higher than the second values, responding to the detected input.

2. The method of claim 1, further comprising:

in accordance with a determination that the first value is not higher than the second scores, forgoing responding to the detected input.

3. The method of claim 2, wherein forgoing responding to the detected input further comprises changing a state of operation of the first electronic device.

10. A first electronic device of a plurality of electronic devices, the first electronic device comprising:

one or more microphones;

one or more processors; and

memory storing one or more programs to be executed by the one or more processors, the one or more programs comprising instructions for:

detecting a voice input from a user;

determining a first value associated with the voice input;

receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices; and

in accordance with a determination that the first value is higher than the second values, responding to the detected input.

12. The first electronic device of claim 10, the one or more programs further comprising instructions for:

communicating to the other devices of the plurality of electronic devices to forgo responding to the detected output.

93. I understand that the parties have offered the following competing constructions for

these terms:

Complainant's Proposed Construction	Respondent's Proposed Construction
No construction necessary (plain and ordinary meaning)	"[refrain]/[refraining] from outputting any and all [audible] responses"

94. A person of ordinary skill in the art reading the '128 and '311 claims in the context of the specification would understand the plain and ordinary meaning of these terms.

95. A person of ordinary skill would understand that the "forgo/forgoing" terms in the '128 claims require that if the first electronic device of the plurality of electronic devices is determined to have the highest quality score, then the other devices of the plurality do not output an audible response to the detected voice input. If, on the other hand, the first electronic device is determined to not have the highest quality score, then the first electronic device does not output a response to the detected voice input.

96. A person of ordinary skill would understand that the "forgoing" terms in '311 claims 2 and 3 require that if the first electronic device of the plurality of electronic devices is determined to not have the highest value, then the first electronic device does not respond to the detected voice input. And a person of ordinary skill would understand that the "forgo" term in '311 claim 12 requires that if the first electronic device of the plurality of electronic devices is determined to have the highest quality score, then it communicates to the other devices of the plurality that they should not respond to the detected voice input.

97. A person of ordinary skill in the art would understand based on the claim language references above and the specification that the claimed "forgo[ing]" is tied to a response *to the detected voice input*. For example, the specification describes that "[i]f a score is not the best (e.g. highest amongst the scores, the electronic device *forgoes outputting responses...to the voice input*." '128 patent, 21:61-65 (emphasis added); *see also id.*, 23:4-7, 29:59-30:12. The patents

provide examples of responses to a detected voice input, including audible responses (e.g., “voice confirmation of the user request, reading out of the user-requested information”) and/or a visual responses (e.g., “a notification or a LED light pattern”). ’128 patent, 29:32-54. The patents then describe how devices can forgo responding, for example by “powering down a speaker or associated amplifier.” ’128 patent, 21:61-65 (“If its score is not the best (e.g., highest) amongst the scores, the electronic device forgoes outputting responses (e.g., responses generated by the server system 140) to the voice input (e.g., by powering down or ‘turning off’ the speaker 520).”); 23:4-7 (“Electronic devices 1002-2 and 1002-3 determine that their quality scores are not the best amongst the three electronic devices 1002, and power down their speakers 520 and forgo outputting responses to the user.”).

98. The patents further describe how a device “forgoes” responding to a voice input, if they are not determined to be the leader, as follows:

In accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, the electronic device 190 forgoes outputting a response to the detected voice input. If a device determines that its score is not the highest amongst the devices with respect to the voice input (e.g., devices 1002-2 and 1002-3), the device does not output a response to the voice input. For example, the devices with the lower scores power down their speakers 520.

Id. 29:59-30:12.

99. By negotiating device leadership such that a “leader” will be responsible for responding to the user’s voice inputs and other devices that are not the “leader” forgo responding, the voice assistant devices reduce user confusion. ’128 patent, 1:43-53.

100. Sonos’s proposal broadens the scope of the claimed “response[s]” to encompass actions that a person of ordinary skill in the art would not consider to be “responding to the voice input,” as that is described in the specification.

101. A person of ordinary skill in the art would also understand from the specification that “responding” to a voice input is distinct from the process of “fulfilling” a user’s request. A person of ordinary skill in the art would understand that fulfillment is a process in which an action or operation is taken in response to an instruction relating to the user’s request, and that in the context of the patent, fulfillment may take place on a different device. For example, the patents describe that a user may make “a request for information or for performance of an action or operation,” and “[t]he electronic device 190 **responds to the request** with voice message responses to the user, where the response may include, for example, requests for additional information to **fulfill the request**, confirmation that the request has been **fulfilled**, notice that the request cannot be **fulfilled**, and so forth. ’128 patent, 10:2-20 (emphases added). Similarly, the patents describe that a device may determine a request from a voice input and “generate **a response that is responsive to the request** (e.g., an audible acknowledgement of the request, an audible inquiry to the user for more information needed to **fulfill the request**).” ’128 patent, 24:61-25:7 (emphases added).

102. The patents further describe that, when the user requests an operation to be performed on a **different** device, the “response” to the voice input is the “instruction” sent to another device to perform that action, rather than the operation performed by the other device. For example the patent explains that “in response to a hotword voice input to awaken the electronic device 190, the electronic device 190 generates and presents (e.g., outputs) a voice greeting that includes the first user’s name (e.g., ‘Hello David,’ ‘Good morning Mary’). The **response may include a voice output and/or instructions to another device to perform an operation.**” ’128 patent, 24:45-51.

103. A person of ordinary skill would understand based on the above descriptions in the

specification that a device may respond to a voice input itself, for example by outputting a greeting or playing media, or the device may respond by issuing instructions to another device to perform an operation. Thus, the specification distinguishes responses to a detected voice input (e.g., requests for additional information or confirmation messages) from acts devices might take to fulfill an action requested within that voice input.

104. The patent specifications also offer a detailed example distinguishing fulfillment from responding. The patents describe that “the user can issue a media play request by saying to the Wi-Fi speaker ‘OK Google, play cat videos on my Living room TV.’ The voice assistant service then **fulfils the media play request** by playing the requested media content on the requested device using a default or designated media application.” ’128 patent, 9:38-43 (emphasis added). In this example, if a device other than “Living room TV” detected the voice input, the “response” to the detected voice input would include the instruction sent to another device from the detecting device to the Living room TV (i.e., the fulfilling device) instructing that TV to perform that the requested action; the fulfillment performed by the Living room TV (i.e., playing cat videos) would not be the “response to the detected voice input.” *Id.*, 24:45-51 (“For example, in response to a hotword voice input to awaken the electronic device 190, the electronic device 190 generates and presents (e.g., outputs) a ...**response [which] may include a voice output and/or instructions to another device to perform an operation..**”) (emphasis added). Thus, the ’128 and ’311 patents describe responding to a voice input as separate and distinct from fulfillment, and the claims do not preclude a device that forgoes outputting a response to a voice input from fulfilling an action based on an instruction from the device that does respond to the detected voice input.

105. A person of ordinary skill would also understand that the patents describe devices can perform still other actions, but still “forgo” responding to a detected voice input. For example,

the patents describe that, “[i]n some implementations, **each of the electronic devices 1002 outputs a visual indication** (e.g., a predefined LED light pattern) **that it is listening** whenever it is listening to a user (e.g., receiving and detecting the user's voice input), and then **just the leader device outputs the response to the voice input** per the leadership determination.” ’128 patent, 23:35-45 (emphases added). Based on this description, a person of ordinary skill would appreciate that outputting a visual indication that a device is merely listening to a user is not a “response to the voice input.”

106. Because Sonos’s proposal broadens the scope of the claimed “response[s]” to encompass more than just responding to a detected voice input, which would foreclose a device from performing actions described in the patents in relation to fulfilling a user’s request or outputting a visual indication that the device is listening, a person of ordinary skill in the art would not understand the claims as Sonos has proposed (e.g., a person of ordinary skill would not understand the terms to require that the device “refrain from outputting **any and all** responses”).

107. For all the reasons described above, a person of ordinary skill in the art would understand the plain and ordinary meaning of these terms, and would not interpret the terms in the manner proposed by Sonos.

D. “in accordance with a determination that the quality score generated by the first electronic device is the highest ... in accordance with a determination that the quality score generated by the first electronic device is not the highest” (’128 Claims 1, 2, 3, 5) / “in accordance with a determination that the first value is higher than the second values ... in accordance with a determination that the first value is not higher than the second scores” (’311 Claim 2)

108. The “in accordance with a determination that...” terms appear claim 1 of the ’128

patent,¹ as follows:

in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting an audible and/or a visual response to the detected voice input, wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input;

in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic device, forgoing outputting a response to the detected voice input.

109. In the '311 patent, one of the “in accordance with a determination that...” terms appears in claim 1, and the other appears in claim 2, which depends from claim 1,² as follows:

[claim 1] in accordance with a determination that the first value is higher than the second values, responding to the detected input.

[claim 2] in accordance with a determination that the first value is not higher than the second scores, forgoing responding to the detected input.

110. I understand that the parties have offered the following competing constructions for these terms:

Complainant's Proposed Construction	Respondent's Proposed Construction
No construction necessary (plain and ordinary meaning)	Indefinite

111. A person of ordinary skill in the art reading the '128 and '311 claims in the context of the specification would understand the plain and ordinary meaning of these terms with reasonable certainty.

112. I understand that Sonos contends in its invalidity contentions that the claims are

¹ Claims 2, 3, and 5 of the '128 patent depend from claim 1. The other independent claims of the '128 patent (claims 6 and 11) recite similar claim language, but I understand the parties' proposed constructions for term are only applicable to claim 1.

² The other independent claims of the '311 patent (claims 6 and 11) also recite similar claim language, but I understand this term has been identified for construction only in claim 1.

indefinite because “it is impossible for an ‘electronic device’ to perform both of these method steps for a given voice input.” This description is not consistent with how a person of ordinary skill in the art would understand these terms.

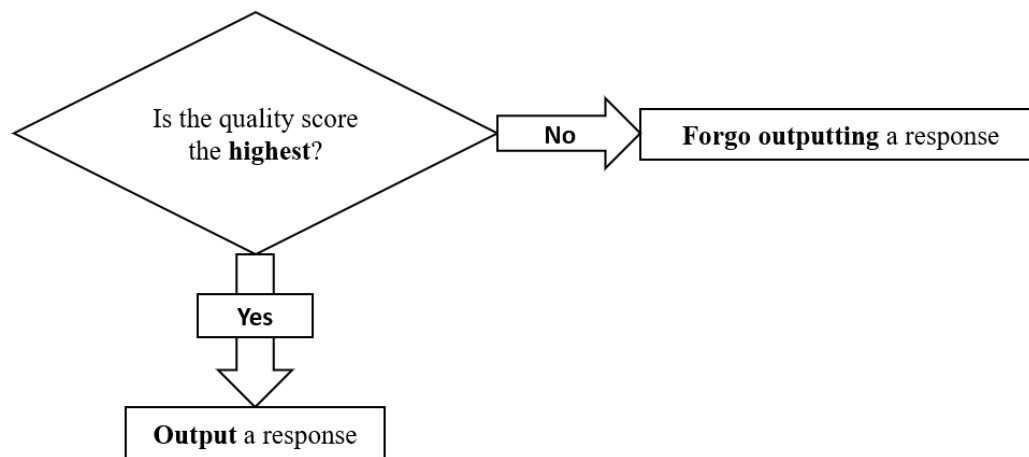
113. A person of ordinary skill would readily understand that the above-referenced claim limitations recite steps presented as alternatives, whose performance depends on the conditions stated in the claims. In other words, one or the other step may be taken, depending on whether the conditions precedent are met.

114. With regard to claim 1 of the ’128 patent, a person of ordinary skill would understand that the step of “outputting an audible and/or a visual response to the detected voice input” is performed “in accordance with a determination that the quality score generated by the first electronic device is **the highest** amongst the generated quality score and received quality scores for the voice input.” Similarly, a person of ordinary skill would understand that the following limitation recites an alternative step of “forgoing outputting a response to the detected voice input,” which is performed “in accordance with a determination that the quality score generated by the first electronic device is **not the highest** amongst the quality scores for the voice input generated by the plurality of electronic devices.” ’128 patent, claim 1 (emphasis added).

115. In claim 2 of the ’311 patent, a person of ordinary skill would similarly understand that the step of “responding to the detected input” is performed “in accordance with a determination that the first value **is higher** than the second values.” And the person of ordinary skill would understand that claim 2 recites an alternative step of “forgoing responding to the detected input,” which is performed “in accordance with a determination that the first value **is not higher** than the second scores.”

116. Thus, a person of ordinary skill would understand that the “in accordance with a

determination that...” language in the claims introduces a condition precedent which, if satisfied, results in the step being performed. If the condition is not satisfied, the step is not performed. The logic of the claim can be illustrated as follows (using the “highest” language from claim 1 of the ’128 patent as an example):



117. A person of ordinary skill would understand that the claim language “in accordance with a determination that...” introduces a condition precedent based on the plain language of the claim itself. Their understanding on that issue would be further confirmed by the specification, which explains that “if” and “in accordance with a determination that” mean the same thing: “As used herein, the term ‘*if*’ is, optionally, construed to mean ‘when’ or ‘upon’ or ‘in response to determining’ or ‘in response to detecting’ or ‘*in accordance with a determination that,*’ depending on the context.” ’128 Patent, 35:6-9 (emphases added). A person of ordinary skill in the art would understand the plain meaning of these conditional steps.

118. A person of ordinary skill in the art would also understand that the conditional language in the claims is consistent with the stated purpose of the invention described in the ’128 and ’311 patents. For example, the patents describe that “[a] location (e.g., a room or space within a home) may include multiple devices that include voice assistant systems and one or more users

of such devices,” and that it is “desirable for there to be a leader amongst the voice assistant devices that is responsible for responding to the user’s voice inputs, in order to reduce user confusion.” ’128 patent, 1:43-45, 50-53. Along these lines, the patents recognized “a need for an electronic device with a voice assistant system and/or a voice assistant server system that incorporates methods and systems for handling multiple users, for multi-device coordination to determine a leader, and for noise mitigation.” ’128 patent, 1:59-63. Based on this description of the purpose of the invention in the ’128 and ’311 patents, a person of ordinary skill would expect the claimed invention to encompass steps that a device takes if it is the “leader,” and steps that it alternatively takes if it is not the “leader.”

119. A person of ordinary skill would further understand the conditional framework of the claims based on other descriptions in the specification, where the patents use “in accordance with a determination that” to introduce the same functionality recited in the claims, and describe the steps as alternatives. For example:

In accordance with a determination that the quality score generated by the first electronic device is ***the highest*** amongst the generated quality score and received quality scores for the voice input, ***the electronic device outputs (1210) an audible response and/or a visual response*** (e.g., a notification or a LED light pattern) to the detected voice input, and the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input.

...

In accordance with a determination that the quality score generated by the first electronic device is ***not the highest*** amongst the quality scores for the voice input generated by the plurality of electronic devices, ***the electronic device 190 forgoes outputting a response to the detected voice input. If*** a device determines that its score is ***not the highest*** amongst the devices with respect to the voice input (e.g., devices 1002-2 and 1002-3), ***the device does not output a response to the voice input.***

’128 patent, 29:32-40, 59-63 (emphases added). These descriptions in the specification confirm to a person of ordinary skill that the ’128 and ’311 claim limitations recite conditional steps.

120. As stated above, I understand that Sonos contends in its invalidity contentions that the claims are indefinite because “it is impossible for an ‘electronic device’ to perform both of these method steps for a given voice input.” A person of ordinary skill would readily appreciate that **both** of the conditions stated in the ’128 and ’311 claims cannot be met simultaneously. In other words, a person of ordinary skill in the art reading claim 1 of the ’128 patent in the context of the specification would not understand it to require a determination that the first electronic device’s quality score is both “the highest” and “not the highest” for a given voice input, and that the first electronic device must therefore both output a response to the voice input and forgo outputting a response to the voice input. Similarly, a person of ordinary skill in the art reading claims 1 and 2 of the ’311 patent in the context of the specification would not understand claim 2 to require a determination that the first electronic device’s first value is both “higher” than the second values and “not higher” than the second values for a given voice input, and that the first electronic device must therefore both respond to the voice input and forgo responding to the voice input. For that reason, in addition to those described above, a person of ordinary skill would not interpret the claims as Sonos suggests.

121. For all the reasons described above, a person of ordinary skill in the art would understand the “in accordance with a determination that...” limitations of the ’128 and ’311 claims with reasonable certainty.

E. “cause the electronic device” (’128 Claims 11, 12, 13, 15)³

122. The term “cause the electronic device” appears in claims 11 and 15 of the ’128 patent. In claim 11, the preamble states the following:

A non-transitory computer readable storage medium storing one or more programs,

³ Although this phrase does not appear in claims 12 and 13, a construction would affect these claims because each depends from claim 11.

the one or more programs comprising instructions, which, when executed by a first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more microphones, a speaker, and one or more processors, cause the electronic device to perform operations comprising:

'128 patent, claim 11.

In claim 15, the preamble recites the following:

The computer readable storage medium of claim 11, further comprising instructions which, when executed by the electronic device, cause the electronic device to perform operations comprising:

'128 patent, claim 15.

123. I understand that the parties have offered the following competing constructions for these terms:

Complainant's Proposed Construction	Respondent's Proposed Construction
No construction necessary (plain and ordinary meaning)	Indefinite

124. A person of ordinary skill in the art reading the '128 claims in the context of the specification would understand the plain and ordinary meaning of this term. It is my understanding that Sonos contends this term is indefinite because the preamble of claim 11 does not specify whether "the electronic device" in the disputed phrase is the same device as the "first electronic device." A person of ordinary skill in the art would understand that "the electronic device" is the same device as the "the first electronic device" recited in the preamble.

125. First, a person of ordinary skill would understand this to be the case based on the plain language of the preambles of claims 11 and 15. Claim 11 recites "[a] non-transitory computer readable storage medium storing one or more programs, the one or more programs comprising instructions, which, when executed by *a first electronic device* of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more microphones, a speaker, and one or more processors, *cause the electronic device* to perform [the operations recited in the

claim].” ’128 patent, claim 11 (emphases added). A person of ordinary skill would recognize that the only device of the “plurality of electronic devices” that claim 11 specifies is “the first electronic device,” and would therefore understand the only “electronic device” to which “the electronic device” could refer to is the “first electronic device. By the same reasoning, a person of ordinary skill in the art would understand that “the electronic device” in claim 15 refers to the “first electronic device.”

126. Further, a person of ordinary skill would understand that the “electronic device” refers to the “first electronic device” based on the context of claim 11 as a whole. Claim 11 recites “instructions” that are executed by “a first electronic device.” The claim states that those instructions, when executed, cause “the electronic device” to perform a number of operations, and the claim states that those operations are carried out by the “first electronic device.” ’128 patent, claim 11. For example, the claims refer to “the quality score generated by the *first electronic device.*” (emphasis added). A person of ordinary skill in the art would understand that “the electronic device” that performs the recited operations is the same as the “first electronic device” that “execute[s]” the recited instruction.

127. A person of ordinary skill in the art would also understand based on the specification that “the electronic device” of claim 11 is the recited “first electronic device.” For example, Figure 6 of the ’128 patent illustrates a series of operations that correspond to those recited in claim 11. Figure 6 shows that those operations are performed “at a first electronic device of a plurality of electronic devices.”

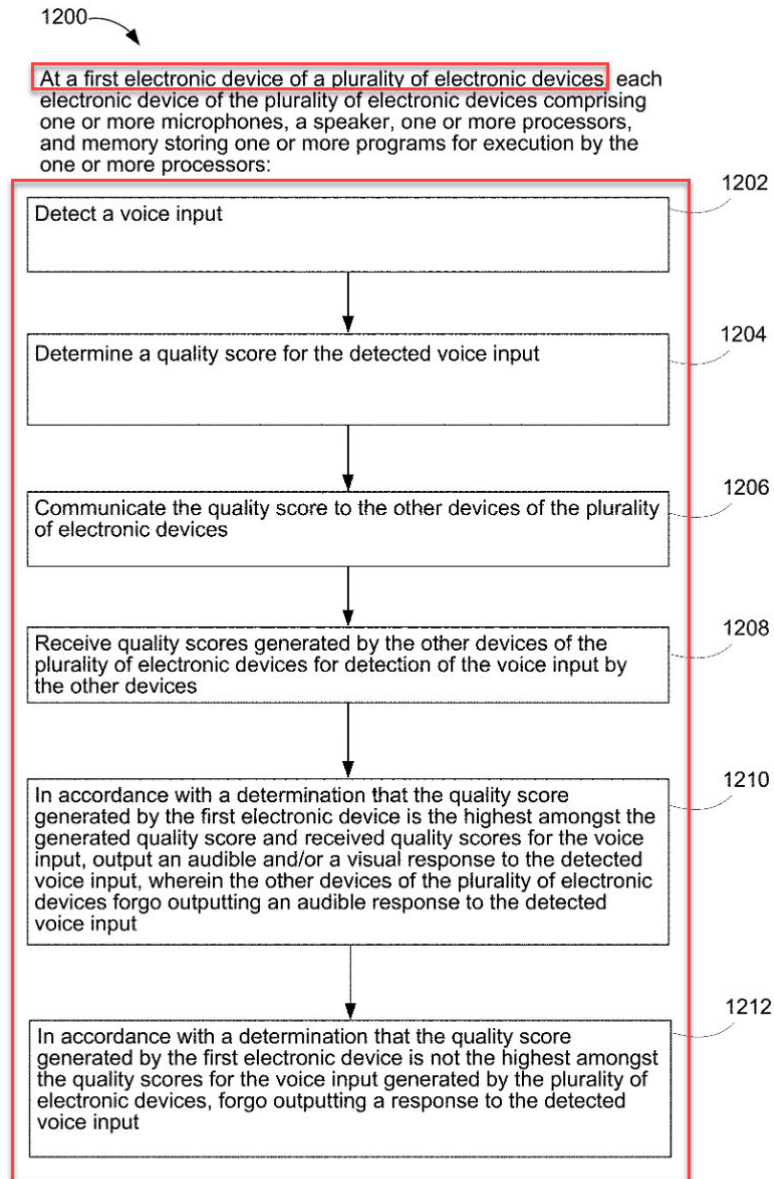


Figure 6

128. The specification explains that Figure 6 shows a method that is “implemented at an *electronic device*” of two or more electronic devices. ’128 patent, Fig. 6, 28:65-29:5. The specification then describes how “the electronic device” carries out the series of operations illustrated in Figure 6 that correspond to those recited in claim 11. *Id.*, 29:6-30:12.

129. Based on this description, a person of ordinary skill in the art would understand that

the electronic device recited in claim 11, which is caused to perform the same operations illustrated in Figure 6, is the same as the “first electronic device” recited in claim 11.

F. “the second scores” (’311 Claims 2, 3)⁴

130. The term “the second scores” appears in claim 2 of the ’311 patent. Claim 2 states the following:

The method of claim 1, further comprising:
in accordance with a determination that the first value is not higher than the second scores, forgoing responding to the detected input.

’311 patent, cl. 2.

131. I understand that the parties have offered the following competing constructions for these terms:

Complainant’s Proposed Construction	Respondent’s Proposed Construction
“the second values”	Indefinite

132. A person of ordinary skill in the art reading the ’311 claims in the context of the specification would understand the plain and ordinary meaning of these terms. A person of ordinary skill in the art would understand that “the second scores” means “the second values.” Specifically, in the context of the claims and the specification, a person of ordinary skill in the art would readily understand that the use of “the second scores” in claim 2, rather than “the second values,” is a clerical error, and a person of ordinary skill would understand with reasonable certainty that the “the second scores” refers to “the second values.”

133. A person of ordinary skill in the art would understand based on the claim language itself that “second scores” was intended to mean “second values.” A person of ordinary skill would

⁴ Although the term “the second scores” does not appear in claim 3, a construction of “the second scores” would affect claim 3 because it depends from claim 2.

appreciate that claim 2, which recites: “in accordance with a determination that the first value is **not higher** than the second scores, **forgoing responding** to the detected input,” is effectively the inverse of the last element of its claim 1, which recites, “in accordance with a determination that the first value is **higher** than the second values, **responding** to the detected input.” ’311 patent, claims 1-2 (emphasis added).

134. From the parallel structure of the language in claims 1 and 2, a person of ordinary skill would understand that “the second scores” (of claim 1) is intended to refer to the same “second values” of independent claim 1. A person of ordinary skill would appreciate that claims 1 and 2 together recite a comparison with two alternative operations: claim 1 describes what the device should do if the first value is higher than the second value, and claim 2 adds what the device should do if the second is higher than the first. A person of ordinary skill in the art would understand that the claims are not intended to recite comparing two different **types** of metrics (i.e., scores with values), especially when claims 1 and 2 are directed to the same comparison to determine whether the first electronic device should respond to the detected input.

135. A person of ordinary skill in the art would also understand that claim 4, which also depends on claim 1, further confirms that “the second scores” in claim 2 was an error. Claim 4 recites the same limitation as claim 2, referring to the correct term “the second values,” rather than “scores”: “The method of claim 1, further comprising: in accordance with a determination that the **first value is not higher than the second values**, identifying a criterion associated with the voice input.” (emphasis added).

136. Finally, a person of ordinary skill in the art would understand that the prosecution history supports that “the second scores” means “the second values.” In a preliminary amendment, then-pending claim 1 was revised to replace the term “score” with “value,” and new claims were

added, including those that issued as claims 2 and 3:

1. (Currently Amended) A method, ~~comprising~~ performed at a first electronic device of a plurality of electronic devices, each ~~electronic device~~ of the plurality of electronic devices comprising one or more microphones, ~~a speaker~~, one or more processors, and memory storing one or more programs for execution by the one or more processors, the method comprising:
 - detecting a voice input from a user;
 - ~~determining a state of the first electronic device and a relevance of the voice input to the first electronic device;~~
 - ~~identifying a subset of electronic devices from the plurality to which the voice input is relevant;~~
 - ~~in accordance with a determination that the subset includes the first electronic device;~~
 - determining a first value score of a criterion associated with the voice input; and
 - receiving second values associated with the voice input scores of the criterionfrom other electronic devices of the plurality of electronic devices in the subset; and
 - in accordance with a determination that the first value score is higher than the second values scores, responding to the detected input.
2. (New) The method of claim 1, further comprising:
 - in accordance with a determination that the first score is not higher than the second scores, forgoing responding to the detected input.
3. (New) The method of claim 2, wherein forgoing responding to the detected input further comprises changing a state of operation of the first electronic device.

137. After the claims were allowed, the Applicant corrected claim 2 “for clarity and to correct for informalities,” Ex. 12 (*311 File History) at 26 (Nov. 25, 2020 Response to Office Action). In that Office Action Response, the Applicant corrected the first instance of “score” to “value,” without correspondingly correcting the second instance of “scores” to “values” to be consistent with claims 1 and 4.

2. (Currently Amended) The method of claim 1, further comprising:
in accordance with a determination that the first value score is not higher than the
second scores, forgoing responding to the detected input.

138. Based on this history, a person of ordinary skill in the art would conclude that the Applicant's failure to amend "the second scores" during prosecution – as it did with the first instance of "score" in the claim – was an obvious error and would understand with reasonable certainty that "second scores" in claim 2 means "second values."

G. "the detected output" ('311 Claim 12)

139. The term "the detected output" appears in claim 12 of the '311 patent. Claim 12 states the following:

12. The first electronic device of claim 10, the one or more programs further comprising instructions for:

communicating to the other devices of the plurality of electronic devices to
forgo responding to the detected output.

'311 patent, cl. 12.

140. I understand that the parties have offered the following competing constructions for these terms:

Complainant's Proposed Construction	Respondent's Proposed Construction
"the detected input"	Indefinite

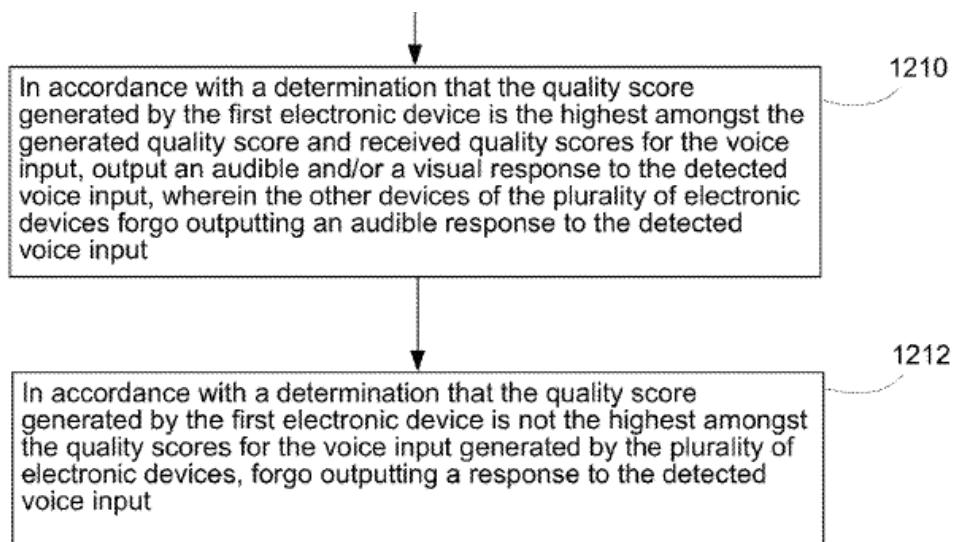
141. A person of ordinary skill in the art reading the '311 claims in the context of the specification would understand the plain and ordinary meaning of these terms. A person of ordinary skill in the art would understand that "the detected output" means "the detected input." Specifically, in the context of the claims and the specification, a person of ordinary skill in the art would readily understand that the use of "the detected output" in claim 12, rather than "the detected input," is a clerical error, and a person of ordinary skill would understand with reasonably certainty

that the “the detected output” refers to “the detected input.”

142. First, a person of ordinary skill would understand from the language of the claims themselves that “the detected output” means “the detected input.” For example, with the exception of the error in claim 12, the ’311 patent claims only recite “voice input[s],” and never recite any “outputs.” ’311 patent, claims 1-4, 8-11, 16, 20 (all reciting “a[/the] detected input,” not “output”).

143. A person of ordinary skill in the art would also appreciate that claim 10, from which claim 12 depends, recites “detecting a voice input,” “determining a first value associated with the voice input,” “receiving second values associated with the voice input,” and then “responding to the detected input.” Thus, a person of ordinary skill would understand that the first electronic device of claim 10 responds to the detected input, and that the added limitation of dependent claim 12 describes that the first electronic device “communicat[es] to the other devices of the plurality of electronic devices to forgo responding” to the same thing, the “detected input” to which the first electronic device responded. A person of ordinary skill in the art would understand that these claims would not make sense together if the first electronic device told the other devices to forgo detecting an output instead of an input, and thus that the recitation of “the detected output” was nothing more than a clerical error.

144. A person of ordinary skill in the art would also understand based on the specification that “the detected output” means “the detected input.” In every instance in which the specification describes a device forgoing a response, the device forgoes responding to a “detected voice input,” never to an output. See ’311 patent, Fig. 6 (step 1210, “wherein the other devices of the plurality of electronic devices forgo outputting an audible response to the detected voice input,” step 1212, “forgo outputting a response to the detected voice input”):



'311 patent, 3:8-15, 3:32-39, 3:57-64, 22:18-20 (“the leaders forgo receiving, detecting, and processing subsequent voice inputs from the user”). Although the specification does include embodiments that forgo “outputting responses” in response to a detected input, no where does it disclose **detecting** an output. See '311 patent, 10:9-10, 22:4-7, 23:16-17, 29:50-52, 30:7-8, 30:20-24, 30:58-60. Further, the '311 patent uses the word “detect[ed/ing]” in connection with “voice input” more than forty times, and only uses the phrase “detected voice output” once – in the clerical error in claim 12. Based on this overwhelming evidence from the specification, a person of ordinary skill would understand that “the detected output” can only mean “the detected input.”

145. A person of ordinary skill in the art would also understand that the prosecution history of the '311 patent supports the conclusion that “the detected output” means “the detected input.” Claim 12, including the language “the detected output” was added during prosecution in a March 19, 2020 preliminary amendment and then allowed on the next office action, pending a terminal disclaimer. Ex. 12 ('311 File History) at 6 (Mar. 19, 2020 Prelim. Amdmt.), 15 (Nov. 5, 2020 Non-Final Office Action).

12. (New) The first electronic device of claim 10, the one or more programs further comprising instructions for:

communicating to the other devices of the plurality of electronic devices to forgo responding to the detected output.

146. In this context it is clear to a person of ordinary skill in the art that “the detected output” was merely a clerical error.

I declare under penalty of perjury that the foregoing is true and correct. Executed on December 12, 2022, in Alpharetta, Georgia.

A handwritten signature in blue ink, appearing to read "David V. Anderson", is written over a solid black horizontal line.

David V. Anderson, Ph.D.

Appendix 1

David V. Anderson

December 1, 2022

Earned Degrees

- 1999 **Ph.D. in Electrical and Computer Engineering.**
Georgia Institute of Technology
- 1994 **M.S. in Electrical Engineering.**
Brigham Young University
- 1993 **B.S. in Electrical Engineering.**
Brigham Young University

II. Employment History

- 2012-present **Professor**, *School of Electrical & Computer Engineering*, Georgia Institute of Technology, Atlanta, Georgia.
- 2011-present **Partner**, *Ratrix Technologies, LLC*, Atlanta, Georgia.
High-tech start-up with NSF SBIR funding to develop wireless data receivers
- 2005-2012 **Associate Professor**, *School of Electrical & Computer Engineering*, Georgia Institute of Technology, Atlanta, Georgia.
- 2009 **Visiting Professor**, *Department of Computer Science*, Korea University, Seoul, South Korea.
- 1999-2005 **Assistant Professor**, *School of Electrical & Computer Engineering*, Georgia Institute of Technology, Atlanta, Georgia.
- 1999 **Education Specialist**, *Texas Instruments, Inc.*, Dallas, Texas.

III. Honors and Awards (selected)

- 2004 Presidential Early Career Award for Scientists and Engineers (PECASE)
- 2004 NSF Faculty Early Career Development (CAREER) Award
- 2006 US Frontiers of Engineering Fellow, National Academy of Engineering
- 2006 Frontiers of Science Fellow, National Academy of Science
- 2009 Nominated by the School of Electrical and Computer Engineering for the Institute 2009 Outstanding Faculty Leadership Award for the Development of Graduate Research Assistants
- 2022-2024 Georgia Tech Provost Teaching and Learning Fellow

IV. Research, Scholarship, and Creative Activities

A. Published Books, Book Chapters, and Edited Volumes

A.1. Published Books

- [1] Wayne T. Padgett and David V. Anderson. *Fixed-point Signal Processing*. Synthesis Lectures on Signal Processing. Morgan & Claypool Publishers, 2009.

A.2. Published Parts of Books

- [1] J. H. McClellan, R. W. Schafer, and M. A. Yoder. *DSP First: A Multimedia Approach*. Prentice Hall, 1998. Assisted in the preparation of the multi-media CD-ROM that accompanies the text. Wrote and revised the laboratory projects included in the text.
- [2] Sheng-Yu Peng, Paul E. Hasler, and David Anderson. An analog programmable multi-dimensional radial basis function based classifier. In *IFIP WG 10.5 International Conference on Very Large Scale Integration of System-on-Chip*, pages 13–18, Atlanta, GA, October 2007.
- [3] Jorge Marin-Hurtado and David V. Anderson. *Independent Component Analysis for Audio and Biosignal applications*, chapter Preservation of Localization Cues in BSS-Based Noise Reduction: Application in Binaural Hearing Aids. InTech, 2012. Ganesh R Naik, ed.

B. Refereed Publications and Submitted Articles

B.1. Refereed Journal Publications

- [1] Paul Hasler, Abhishek Bandyopadhyay, and David V. Anderson. High fill-factor imagers for neuromorphic processing enabled by floating-gate circuits. *EURASIP Journal on Applied Signal Processing*, 2003(7):676–689, June 2003. Invited paper for special issue on neuromorphic signal processing and implementation.
- [2] Venkatesh Krishnan and David V. Anderson. Joint design of channel-optimized multistage vector quantizer. *IEEE Signal Processing Letters*, 11(1):5–7, January 2004.
- [3] Venkatesh Krishnan, David V. Anderson, and Kwan Truong. Optimal multistage vector quantization of LPC parameters over noisy channels. *IEEE Transactions on Speech and Audio Processing*, 12(1):1–8, January 2004.
- [4] Paul S. Hong, David V. Anderson, Doug B. Williams, Thomas P. Barnwell III, Joel R. Jackson, Monson H. Hayes III, Ronald W. Schafer, and John D. Echard. “DSP for practicing engineers:” A case study in internet course delivery. *IEEE Transactions on Education*, 47(3):301–310, August 2004.
- [5] Abhishek Bandyopadhyay, Paul Hasler, and David V. Anderson. A CMOS floating-gate matrix transform imager. *IEEE Sensors*, 5(3):455–462, 2005.
- [6] Paul Hasler, Paul D. Smith, Rich Ellis, David Graham, and David Anderson. Analog floating-gate, on-chip auditory sensing system interfaces. *IEEE Sensors*, 5:1027–1034, October 2005.

- [7] Sourabh Ravindran, Kristopher Schlemmer, and David V. Anderson. A physiologically inspired method for audio classification. *EURASIP Journal on Applied Signal Processing*, 2005(9):1374–1381, 2005.
- [8] Tyson S. Hall, Christopher M. Twigg, Paul Hasler, and David V. Anderson. Developing large-scale field-programmable analog arrays for rapid prototyping. *International Journal for Embedded Systems*, 1(3/4):179–192, 2005.
- [9] Sourabh Ravindran, Paul Smith, David Graham, Varinthira Duangudom, David Anderson, and Paul Hasler. Towards biologically inspired on-chip auditory processing. *EURASIP Journal on Applied Signal Processing*, 2005(7):1082–1092, 2005.
- [10] Daniel J. Allred, David V. Anderson, Walter Huang, Venkatesh Krishnan, and Heejong Yoo. LMS adaptive filters using distributed arithmetic for high throughput. *IEEE Transactions on Circuits and Systems*, 52(7):1327–1337, July 2005.
- [11] Tyson S. Hall and David V. Anderson. A framework for teaching real-time digital signal processing with field-programmable gate arrays. *IEEE Transactions on Education*, 48(3):551 – 558, August 2005.
- [12] Tyson S. Hall, Christopher M. Twigg, Jordan D. Gray, Paul Hasler, and David V. Anderson. Large-scale field-programmable analog arrays for analog signal processing. *IEEE Transactions on Circuits and Systems*, 52(11):2298 – 2307, November 2005.
- [13] Faik Baskaya, Sasank Reddy, Sung Kyu Lim, and David V. Anderson. Placement for large-scale floating-gate field programable analog arrays. *IEEE Transactions on Very Large Scale Integration Systems*, 14(8):906–910, August 2006.
- [14] Sheng-Yu Peng, Paul Hasler, and David V. Anderson. An analog programmable multi-dimensional radial basis function based classifier. *IEEE Transactions on Circuits and Systems I*, 54(10):2148–2158, October 2007.
- [15] Erhan Ozalevli, Walter Huang, Paul E. Hasler, and David V. Anderson. A reconfigurable mixed-signal VLSI implementation of distributed arithmetic used for finite-impulse response filtering. *IEEE Transactions on Circuits and Systems I*, 55(2):510–521, March 2008.
- [16] Kofi M. Odame, David V. Anderson, and Paul Hasler. A bandpass filter for inherent gain adaptation for hearing applications. *IEEE Transactions on Circuits and Systems I*, 55(3):786–795, April 2008.
- [17] Wei Zhang, Xiaoli Ma, Brian Gestner, and David V. Anderson. Designing low-complexity equalizers for wireless systems. *IEEE Communications Magazine*, 47(1):56–62, January 2009.
- [18] Faik Baskaya, David V. Anderson, and Sung Kyu Lim. Net sensitivity based optimization of large-scale field programmable analog array (FPAA) placement

and routing. *IEEE Transactions on Circuits and Systems II*, 56(7):565–569, July 2009.

- [19] L.J. Karam, I. Alkamal, A. Gatherer, G.A. Frantz, D.V. Anderson, and B.L. Evans. Trends in multicore DSP platforms. *IEEE Signal Processing Magazine*, 26(6):38–49, November 2009.
- [20] Bo Marr, Jason George, Brian Degnan, David V. Anderson, and Paul E. Hasler. Error immune logic for low power probabilistic computing. *VLSI Design*, 2010. doi:10.1155/2010/460312.
- [21] Walter Huang and David V. Anderson. Modified sliding-block distributed arithmetic with offset binary coding for adaptive filters. *Journal of Signal Processing Systems*, April 2010. Available: <http://dx.doi.org/10.1007/s11265-010-0479-4>.
- [22] David V. Anderson. Storytelling—the missing art in engineering presentations. *IEEE Signal Processing Magazine*, 28(2):109–111, March 2011.
- [23] Brian Gestner, Wei Zhang, Xiaoli Ma, and David V. Anderson. Lattice reduction for MIMO detection: From theoretical analysis to hardware realization. *IEEE Transactions on Circuits and Systems I*, 58(4):813–826, April 2011.
- [24] J. Marin Hurtado and David V. Anderson. FFT-based block processing in speech enhancement: Potential artifacts and solutions. *IEEE Transactions on Audio, Speech and Language Processing*, 19(8):2527–2537, November 2011.
- [25] Brian Gestner, Xiaoli Ma, and David V. Anderson. Incremental lattice reduction: Motivation, theory, and practical implementation. *IEEE Transactions on Wireless Communications*, 11(1):188–198, January 2012.
- [26] Bo Marr, Brian Degnan, Paul Hasler, and David V. Anderson. Scaling energy per operation via an asynchronous pipeline. *IEEE Transactions on VLSI*, PP(99):1–5, 2012.
- [27] J.I. Marin-Hurtado, D.N. Parikh, and D.V. Anderson. Perceptually inspired noise-reduction method for binaural hearing aids. *Audio, Speech, and Language Processing, IEEE Transactions on*, 20(4):1372–1382, May 2012.
- [28] B. Gestner, Xiaoli Ma, and D.V. Anderson. Incremental lattice reduction: Motivation, theory, and practical implementation. *Wireless Communications, IEEE Transactions on*, 11(1):188–198, January 2012.
- [29] A.A. Kressner, D.V. Anderson, and C.J. Rozell. Evaluating the generalization of the hearing aid speech quality index (hasqi). *Audio, Speech, and Language Processing, IEEE Transactions on*, 21(2):407–415, February 2013.
- [30] S. Ramakrishnan, A. Basu, Leung Kin Chiu, J. Hasler, D. Anderson, and S. Brink. Speech processing on a reconfigurable analog platform. *Very Large*

Scale Integration (VLSI) Systems, IEEE Transactions on, 22(2):430–433, Feb 2014.

- [31] Nashlie H Sephus, Aaron D Lanterman, and David V Anderson. Modulation spectral features: In pursuit of invariant representations of music with application to unsupervised source identification. *Journal of New Music Research*, 44(1):58–70, 2015.
- [32] Jinwoo Kang, David V. Anderson, and Monson H. Hayes. Face recognition for vehicle personalization with near infrared frame differencing. *IEEE Transactions on Consumer Electronics*, 62(3):316–324, August 2016.
- [33] B. T. Carroll, B. M. Whitaker, W. Dayley, and D. V. Anderson. Outlier learning via augmented frozen dictionaries. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(6):1207–1215, June 2017.
- [34] Muhammad Rizwan and David V. Anderson. A weighted accent classification using multiple words. *Neurocomput.*, 277(C):120–128, February 2018.
- [35] Chieh-Feng Cheng, Abbas Rashidi, Mark A. Davenport, and David V. Anderson. Activity analysis of construction equipment using audio signals and support vector machines. *Automation in Construction*, June 2017.
- [36] Bradley M Whitaker, Pradyumna Byappanahalli Suresha, Chengyu Liu, Gari D Clifford, and David V Anderson. Combining sparse coding and time-domain features for heart sound classification. *Physiological Measurement*, 38:1701–1729, 08 2017.
- [37] Chris Sabillon, Abbas Rashidi, Biswanath Samanta, Mark A. Davenport, and David V. Anderson. An audio-based Bayesian model for productivity estimation of construction cyclic activities. *Journal of Computing in Civil Engineering*, 34(1), 2020.
- [38] Muhammed Rizwan, Bradley M Whitaker, and David V Anderson. AF detection from ECG recordings using feature selection, sparse coding, and ensemble learning. *Physiological measurement*, 39(12), December 2018.
- [39] M. N. Sahadat, N. Sebkhi, D. Anderson, and M. Ghovanloo. Optimization of tongue gesture processing algorithm for standalone multimodal tongue drive system. *IEEE Sensors Journal*, 19(7):2704–2712, April 2019.
- [40] Kaitlin L. Fair, Daniel R. Mendat, Andreas G. Andreou, Christopher J. Rozell, Justin Romberg, and David V. Anderson. Sparse coding using the locally competitive algorithm on the truenorth neurosynaptic system. *Frontiers in Neuroscience*, 13:754, 2019.
- [41] Chris Sabillon, Abbas Rashidi, Biswanath Samanta, Mark Davenport, and David Anderson. Audio-based bayesian model for productivity estimation of cyclic construction activities. *Journal of Computing in Civil Engineering*, 34:04019048, 01 2020.

- [42] N. Sebkhi, A. Bhavsar, D. V. Anderson, J. Wang, and O. T. Inan. Inertial measurements for tongue motion tracking based on magnetic localization with orientation compensation. *IEEE Sensors Journal*, 21(6):7964–7971, March 2021.
- [43] Nordine Sebkhi, Arpan Bhavsar, Md NAZMUS Sahadat, Jesse Baldwin, Erica Walling, Amber Biniker, Michelle Hoefnagel, Geneva Tonuzi, Raine Osborne, David V. Anderson, and Omer Inan. Evaluation of a head-tongue controller for power wheelchair driving by people with quadriplegia. *IEEE Transactions on Biomedical Engineering*, 2021.
- [44] Mohammad M. Salut and David V. Anderson. Online tensor robust principal component analysis. *IEEE Access*, 10:69354–69363, 2022.

B.2. Conference Presentations with Proceedings (Refereed)

- [1] David V. Anderson, Richard W. Harris, and Douglas M. Chabries. Evaluation of a hearing compensation algorithm. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 3531–3533, Detroit, May 1995.
- [2] Douglas M. Chabries, David V. Anderson, Thomas G. Stockham, Jr., and Richard W. Christiansen. Application of a human auditory model to loudness perception and hearing compensation. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 3527–3530, Detroit, May 1995.
- [3] David V. Anderson. Speech analysis and coding using a multi-resolution sinusoidal transform. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1037–1040, Atlanta, May 1996.
- [4] David V. Anderson, James H. McClellan, Ronald W. Schafer, Jeffrey B. Schodorf, and Mark A. Yoder. DSP First - a first course in ECE. In *Proceedings of the 1996 Asilomar Conference on Circuits, Systems, and Computers*, volume 1, pages 226–230, Pacific Grove, CA, May 1996.
- [5] David V. Anderson and Mark A. Clements. Audio signal noise reduction using multi-resolution sinusoidal modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 805–808, Phoenix, May 1999.
- [6] David V. Anderson, Lonnie Harvel, Monson H. Hayes III, Y. Ishiguro, Joel R. Jackson, and Maria Pimentel. Internet course delivery-making it easier and more effective. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, volume 1, pages 84–87, New York, July 2000.
- [7] David V. Anderson, Thomas P. Barnwell, Monson H. Hayes III, and Joel R. Jackson. Effective and efficient distance learning over the internet:

- Implementation of an on-line DSP course. In *Proc. International Conference on Engineering Education*, Taiwan, August 2000.
- [8] Joel R. Jackson, David V. Anderson, and Monson H. Hayes III. Effective and efficient distance learning over the internet: Tools and techniques. In *Proc. International Conference on Engineering Education*, Taiwan, August 2000.
 - [9] Monson H. Hayes III, Joel R. Jackson, and David V. Anderson. Producing effective internet courses with *inFusion*. In *Proceedings Learning '00*, Madrid, Spain, October 2000.
 - [10] David V. Anderson, Thomas P. Barnwell, John Echard, Monson H. Hayes III, Joel R. Jackson, Ronald W. Schafer, and Douglas B. Williams. An on-line DSP course for practicing engineers. In *Signal Processing Education Workshop*, Hunt, TX, October 2000. Web Proceedings:
<http://spib.ece.rice.edu/SPTM/DSP2000/>.
 - [11] David V. Anderson and Paul Hasler. Cooperative analog/digital signal processing. In *World Conference on Systemics, Cybernetics, and Informatics*, Orlando, FL, July 2001. *Invited Paper, Best Paper Award*.
 - [12] Matt Kucic, Paul Hasler, Jeff Dugger, and David V. Anderson. Programmable and adaptive analog filters using arrays of floating-gate circuits. In Erik Brunvand and Chris Myers, editors, *2001 Conference on Advanced Research in VLSI*, pages 148–162, Salt Lake City, March 2001. IEEE Computer Society.
 - [13] Joel R. Jackson, Thomas P. Barnwell, Douglas B. Williams, Monson H. Hayes III, David V. Anderson, and Ronald W. Schafer. DSP for practicing engineers: an online continuing DSP education course. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume V, pages 2721–2724, Salt Lake City, UT, May 2001.
 - [14] Joel R. Jackson, Thomas P. Barnwell, Douglas B. Williams, Monson H. Hayes III, David V. Anderson, and Ronald W. Schafer. Online DSP education: DSP for practicing engineers. In *American Society for Engineering Education Annual Conference*, Albuquerque, NM, June 2001.
 - [15] Joel R. Jackson, Thomas P. Barnwell, David V. Anderson, and Monson H. Hayes III. *inFusion*: Simplifying online course creation. In *American Society for Engineering Education Annual Conference*, Albuquerque, NM, June 2001.
 - [16] Paul Smith, Matt Kucic, Rich Ellis, Paul Hasler, and David V. Anderson. Cepstrum frequency encoding in analog floating-gate circuitry. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume IV, pages 671–674, Phoenix, AZ, May 2002.
 - [17] Paul Hasler and David V. Anderson. Cooperative analog-digital signal processing. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume IV, pages 3972–3975, Orlando, FL, May 2002.

- [18] Heejong Yoo, David V. Anderson, and Paul Hasler. Continuous-time audio noise suppression and real-time implementation. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume IV, pages 3980–3983, Orlando, FL, May 2002.
- [19] Richard Ellis, Heejong Yoo, and David V. Anderson. An analog floating-gate IC for audio noise suppression. In *Proceedings of the International Symposium on Circuits and Systems*, volume II, pages 728–31, Phoenix, AZ, May 2002. Invited Paper.
- [20] Paul Hasler, Paul Smith, Richard Ellis, David Graham, and David V. Anderson. Biologically inspired auditory sensing system interfaces on a chip. In *2002 IEEE Sensors Conference*, Orlando, FL, June 2002. *Invited Paper, nominated for Best Paper of Conference*.
- [21] Paul Hasler, Abhishek Bandyopadhyay, and David V. Anderson. Low-power analog image processing using transform imagers. In *IEEE Midwest Circuits and Systems*, Tulsa, OK, August 2002.
- [22] Paul Hasler, Paul Smith, Chris Duffy, Christal Gordon, Jeff Dugger, and David Anderson. A floating-gate vector-quantizer. In *IEEE Midwest Circuits and Systems*, Tulsa, OK, August 2002.
- [23] Tyson S. Hall, Paul Hasler, and David V. Anderson. Field-programmable analog arrays: A floating-gate approach. In *12th International Conference on Field Programmable Logic and Applications*, Montpellier, France, September 2002.
- [24] Paul Hasler, Abhishek Bandyopadhyay, and David Anderson. Low-power analog image processing using transform imagers. In *Digital Signal Processing Workshop*, pages 333–338, Pine Mountain, Georgia, October 2002.
- [25] David Anderson, Paul Hasler, Rich Ellis, Heejong Yoo, David Graham, and Mat Hans. A low-power, continuous-time system for audio noise suppression and a VLSI implementation. In *Digital Signal Processing Workshop*, pages 327–332, Pine Mountain, Georgia, October 2002.
- [26] Tyson S. Hall and David V. Anderson. From algorithms to gates: Developing a pedagogical framework for teaching DSP hardware design. In *Signal Processing Education Workshop*, pages 157–161, Pine Mountain, Georgia, October 2002.
- [27] P. Spencer Whitehead, David V. Anderson, and Mark A. Clements. Adaptive acoustic noise suppression for speech enhancement. In *Proceedings 2003 International Conference on Multimedia and Expo*, volume 1, pages 565–568, Baltimore, MD, June 2003.
- [28] Cenk Demiroglu and David V. Anderson. Noise robust digit recognition with missing frames. In *8th European Conference on Speech Communication and Technology*, pages 2165–2168, Geneva, Switzerland, September 2003.
- [29] Venkatesh Krishnan and David V. Anderson. Robust jointly optimized multistage vector quantization for speech coding. In *8th European Conference*

on Speech Communication and Technology, pages 1093–1096, Geneva, Switzerland, September 2003.

- [30] Sunil D. Kamath and David V. Anderson. Signal processing in digital and floating-gate analog circuits; design trade-offs. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2003.
- [31] Heejong Yoo, David V. Anderson, and Paul Hasler. On delay structures for the analog adaptive filters with long filter taps. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2003.
- [32] Daniel Allred, Venkatesh Krishnan, Walter Huang, and David V. Anderson. Implementation of an LMS adaptive filter on an FPGA employing multiplexed multiplier architecture. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2003.
- [33] Sourabh Ravindran, Cenk Demiroglu, and David V. Anderson. Speech recognition using filter bank features. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2003.
- [34] Thomas Barnwell III, Mark Clements, David Anderson, Elliot Moore, Matthew Lee, Erdem Ertan, Venkatesh Krishnan, Woosuk Choi, James Hu, Cenk Demiroglu, Spencer Whitehead, and Adriane Durey. Low bit-rate coding of speech in harsh conditions using non-acoustic auxiliary devices. In *Special Workshop in Maui: Lectures by the Masters in Signal Processing*, Maui, HI, January 2004.
- [35] Michael Healy, Sourabh Ravindran, and David V. Anderson. Effects of varying parameters in asymmetric adaboost on the accuracy of a cascade audio classifier. In *IEEE SoutheastCon 2004*, Greensboro, NC, March 2004.
- [36] Tyson S. Hall, Christopher M. Twigg, Paul Hasler, and David V. Anderson. Developing large-scale field-programmable analog arrays. In *11th Reconfigurable Architectures Workshop (RAW 2004), 18th International Parallel and Distributed Processing Symposium*, pages 142–147, Santa Fé, NM, April 2004.
- [37] Tyson S. Hall, Christopher M. Twigg, Paul Hasler, and David V. Anderson. Application performance of elements in a floating-gate FPAA. In *Proceedings of the International Symposium on Circuits and Systems*, Vancouver, Canada, May 2004.
- [38] Heejong Yoo, David Graham, David V. Anderson, and Paul Hasler. C4 band-pass delay filter for continuous-time subband adaptive tapped-delay filter. In *Proceedings of the International Symposium on Circuits and Systems*, Vancouver, Canada, May 2004.
- [39] Sourabh Ravindran and David V. Anderson. Boosting as a dimensionality reduction tool for audio classification. In *Proceedings of the International Symposium on Circuits and Systems*, Vancouver, Canada, May 2004.

- [40] Cenk Demiroglu and David V. Anderson. Two-sensor noise-robust ASR with missing frames for Aurora2 task. In *Proceedings of the International Symposium on Circuits and Systems*, Vancouver, Canada, May 2004.
- [41] Shyam Subramanian, David V. Anderson, and Paul Hasler. Synthesis of static multiple input multiple output MITE networks. In *Proceedings of the International Symposium on Circuits and Systems*, Vancouver, Canada, May 2004.
- [42] Haw-Jing Lo, Guillermo Serrano, Paul Hasler, David Anderson, and Bradley Minch. Programmable multiple input translinear elements. In *Proceedings of the International Symposium on Circuits and Systems*, Vancouver, Canada, May 2004.
- [43] Sourabh Ravindran, David V. Anderson, and Malcolm Slaney. Low-power audio classification for ubiquitous sensor networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004.
- [44] Daniel J. Allred, Heejong Yoo, Venkatesh Krishnan, and David V. Anderson. A novel high performance distributed arithmetic adaptive filter implementation on an FPGA. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004.
- [45] Daniel J. Allred, Walter Huang, Venkatesh Krishnan, Heejong Yoo, and David V. Anderson. An FPGA implementation for a high throughput adaptive filter using distributed arithmetic. In *FCCM*, pages 324–325, 2004.
- [46] Tyson S. Hall and David V. Anderson. Merging theory and implementation: A framework for teaching DSP hardware design. In *2004 ASEE Annual Conference*, Salt Lake City, UT, June 2004.
- [47] Rongqiang Hu and David V. Anderson. Audio noise suppression based on neuromorphic saliency and phoneme adaptive filtering. In *IEEE Digital Signal Processing Workshop*, Taos, NM, August 2004.
- [48] Sunil Kamath, Sourabh Ravindran, and David V. Anderson. Independent component analysis for audio classification. In *IEEE Digital Signal Processing Workshop*, Taos, NM, August 2004.
- [49] Sourabh Ravindran and David V. Anderson. Cascade classifiers for audio classification. In *IEEE Digital Signal Processing Workshop*, Taos, NM, August 2004.
- [50] Cenk Demiroglu and David V. Anderson. Noise robust digit recognition using a glottal radar sensor for voicing detection. In *International Conference on Spoken Language Processing*, Jeju, Korea, October 2004.
- [51] Rongqiang (James) Hu and David V. Anderson. Single acoustic channel speech enhancement based on glottal correlation using non-acoustic sensors. In

International Conference on Spoken Language Processing, Jeju, Korea, October 2004.

- [52] Cenk Demiroglu and David V. Anderson. A soft decision MMSE amplitude estimator as a noise preprocessor to speech coders using a glottal sensor. In *International Conference on Spoken Language Processing*, Jeju, Korea, October 2004.
- [53] Venkatesh Krishnan, Christopher Duffy, David V. Anderson, and Paul Hasler. Optimal quantization employing programmable flash analog to digital converters. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2004. Invited paper.
- [54] Rongqiang Hu and David V. Anderson. Improved perceptually inspired speech enhancement using a psychoacoustic model. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2004.
- [55] Cenk Demiroglu, Sunil Kamath, David V. Anderson, Mark A. Clements, and Thomas P. Barnwell III. Segmentation based speech enhancement using auxiliary sensors. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2004.
- [56] Teahyung Lee and David V. Anderson. The behavioral simulation results of the optical flow estimation algorithm for CADSP system design. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, Pacific Grove, CA, November 2004.
- [57] Heejong Yoo and David V. Anderson. Hardware-efficient distributed arithmetic architecture for high-order digital filters. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 125–128, Philadelphia, Pennsylvania, March 2005.
- [58] Cenk Demiroglu, Sunil Kamath, and David V. Anderson. Segmentation-based speech enhancement for intelligibility improvement in MELP coders using auxiliary sensors. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 797–800, Philadelphia, Pennsylvania, March 2005.
- [59] Faik Baskaya, Sasank Reddy, Sung Kyu Lim, Tyson Hall, and David V. Anderson. Mapping algorithm for large-scale field programmable analog array. In *ACM International Workshop on Physical Design*, pages 152–158, San Francisco, April 2005. ACM Press.
- [60] Sourabh Ravindran and David V. Anderson. Audio classification and scene recognition for hearing aids. In *Proceedings of the International Symposium on Circuits and Systems*, volume 2, pages 830–863, Kobe, Japan, May 2005.
- [61] Shyam Subramanian, David V. Anderson, Paul Hasler, and Bradley A. Minch. Synthesis of MITE log-domain filters with unique operating points. In

Proceedings of the International Symposium on Circuits and Systems, volume 2, pages 996–999, Kobe, Japan, May 2005.

- [62] I. Faik Baskaya, Sasank Reddy, Sung Kyu Lim, and David Anderson. Hierarchical placement for large-scale FPAA. In *International Conference on Field Programmable Logic and Applications*, pages 421–426, Tampere, Finland, August 2005.
- [63] Sourabh Ravindran, David Anderson, and James Rehg. Cascade jump support vector machine classifiers. In *Machine Learning in Signal Processing 2005*, pages 135–139, Mystic, Connecticut, September 2005.
- [64] Rongqiang Hu, Venkatesh Krishnan, and David V. Anderson. Speech bandwidth extension by improved codebook mapping towards increased phonetic classification. In *Interspeech 2005*, pages 1501–1504, Lisbon, Portugal, September 2005.
- [65] Rongqiang Hu, Sunil Kamath, and David V. Anderson. Speech enhancement using non-acoustic sensors. In *Interspeech 2005*, pages 2305–2308, Lisbon, Portugal, September 2005.
- [66] Venkatesh Krishnan, Thomas P. Barnwell-III, and David V. Anderson. Using dynamic codebook re-ordering to exploit inter-frame correlation in MELP coders. In *Interspeech 2005*, pages 2717–2720, Lisbon, Portugal, September 2005.
- [67] Venkatesh Krishnan, Phil S. Whitehead, David V. Anderson, and Mark A. Clements. A framework for estimation of clean speech by fusion of outputs from multiple speech enhancement systems. In *Interspeech 2005*, pages 2317–2320, Lisbon, Portugal, September 2005.
- [68] Sourabh Ravindran, Sunil Kamath, and David Anderson. Implementation of low-power automatic audio classification for hearing-aids on TI c5510. In *Texas Instruments Developers Conference*, Dallas, Texas, February 2006.
- [69] Venkatesh Krishnan, Sabat M. Siniscalchi, David V. Anderson, and Mark A. Clements. Noise robust Aurora-2 speech recognition employing a codebook-constrained Kalman filter preprocessor. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume I, pages 781–784, Toulouse, France, May 2006.
- [70] Teahyung Lee and David V. Anderson. Architecture for hierarchical block motion estimation using variable block sizes. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume III, pages 960–963, Toulouse, France, May 2006.
- [71] Teahyung Lee and David V. Anderson. Performance analysis of a correlation-based optical flow algorithm under noisy environments. In *Proceedings of the International Symposium on Circuits and Systems*, pages 4699–4702, Kos, Greece, May 2006.

- [72] Walter Huang, Venkatesh Krishnan, and David V. Anderson. Conjugate distributed arithmetic adaptive FIR filters and their hardware implementation. In *IEEE International Midwest Symposium on Circuits and Systems*, San Juan, Puerto Rico, August 2006.
- [73] R. Velmurugan, S. Subramanian, V. Cevher, D. Abramson, K. M. Odame, J.D. Gray, H.J. Lo, J. H. McClellan, and D. V. Anderson. On low-power analog implementation of particle filters for target tracking. In *Proceedings of the 14th European Signal Processing Conference*, Florence, Italy, September 2006. Invited Paper.
- [74] Nikolaos Vasiloglou, Alexander G. Gray, and David V. Anderson. Parameter estimation for manifold learning, through density estimation. In *IEEE International Workshop on Machine Learning for Signal Processing*, pages 211–216, Maynooth, Ireland, September 2006.
- [75] Sourabh Ravindran, David V. Anderson, and Malcolm Slaney. Improving the noise robustness of mel-frequency cepstral coefficients. In *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition*, pages 48–52, Pittsburgh, PA, September 2006.
- [76] David V. Anderson and Sourabh Ravindran. Distributed acquisition and processing systems for speech and audio. In *Allerton Conference on Communication, Control, and Computing*, Monticello, IL, September 2006.
- [77] Teahyung Lee and David V. Anderson. Checkerboard-type filtering for a low-power gradient-based optical flow estimation system. In *Proceedings of the IEEE International Conference on Image Processing*, pages 3285–3288, Atlanta, GA, October 2006.
- [78] Teahyung Lee and David V. Anderson. The wavelet-based multi-resolution motion estimation using temporal aliasing detection. In Chang Wen Chen, Dan Schonfeld, and Jiebo Luo, editors, *Proceedings SPIE Electronic Imaging Science and Technology*, volume 6508 of *Visual Communications and Image Processing 2007*, pages 2s–1, San Jose, CA, January 2007. SPIE.
- [79] David V. Anderson, Sourabh Ravindran, and Malcolm Slaney. Varying time-constants and gain adaptation in feature extraction for speech processing. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 669–672, April 2007.
- [80] Chris Twigg, Paul Hasler, and David Anderson. Large-scale FPAA devices for signal processing applications. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 69–72, April 2007.
- [81] Cenk Demiroglu, David Anderson, Mark Clements, and Thomas Barnwell. Multi-sensor spectro-temporal comb filtering for speech enhancement. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 589–592, April 2007.

- [82] Faik Baskaya, Brian Gestner, Chris Twigg, Sung Kyu Lim, David V. Anderson, and Paul Hasler. Rapid prototyping of large-scale analog circuits with field programmable analog array. In *IEEE Symposium on Field-Programmable Custom Computing Machines*, pages 319–320, April 2007.
- [83] Brian Gestner, Jason Tanner, and David Anderson. Glass break detector analog front-end using novel classifier circuit. In *Proceedings of the International Symposium on Circuits and Systems*, pages 3586–3589, May 2007.
- [84] Shyam Subramanian, Paul Hasler, Bradley Minch, and David Anderson. Optimal synthesis of MITE translinear loops. In *Proceedings of the International Symposium on Circuits and Systems*, pages 2822–2825, May 2007.
- [85] Shyam Subramanian and David Anderson. 2-MITE product-of-power-law networks. In *Proceedings of the International Symposium on Circuits and Systems*, pages 1041–1044, May 2007.
- [86] Erhan Ozalevli, Walter Huang, David V. Anderson, and Paul Hasler. VLSI implementation of a reconfigurable mixed-signal finite impulse response filter. In *Proceedings of the International Symposium on Circuits and Systems*, pages 2168–2171, May 2007.
- [87] Cenk Demiroglu, David V. Anderson, and Mark A. Clements. A missing data-based feature fusion stragey for noise-robust automatic speech recognition using noisy sensors. In *Proceedings of the International Symposium on Circuits and Systems*, pages 965–968, May 2007.
- [88] Tyson S. Hall and David V. Anderson. Teaching hardware design of fixed-point digital signal processing systems. In *Proceedings of the American Society for Engineering Education Annual Conference*, volume AC, page 2711, Honolulu, HI, June 2007.
- [89] T. Lee, L. K. Chiu, D. V. Anderson, R. Robucci, and P. Hasler. Rapid algorithm verification for cooperative analog-digital imaging systems. In *Proceedings of the IEEE Midwest Symposium on Circuits and Systems*, 2007.
- [90] Brian Gestner and David V. Anderson. Automatic generation of ModelSim-Matlab interface for RTL debugging and verification. In *IEEE International Midwest Symposium on Circuits and Systems*, 2007.
- [91] Varinthira Duangudom and David V. Anderson. Using auditory saliency to understand complex auditory scenes. In *Proceedings of the European Signal Processing Conference*, pages 1206–1210, 2007.
- [92] Sheng-Yu Peng, Paul E. Hasler, and David Anderson. An analog programmable multi-dimensional radial basis function based classifier. In *IFIP WG 10.5 International Conference on Very Large Scale Integration of System-on-Chip*, pages 13–18, Atlanta, GA, October 2007.

- [93] Jungwon Lee, Teahyung Lee, and David V. Anderson. Improved wavelet-based embedded image coding using a dynamic index reordering vector quantizer. In *Data Compression Conference*, page 530, Snowbird, UT, USA, March 2008.
- [94] Ryan Robucci, Leung Kin Chiu, Jordan Gray, Justin Romberg, Paul Hasler, and David V. Anderson. Compressive sensing on a CMOS separable transform image sensor. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 5125–5128, Las Vegas, Nevada, March 2008.
- [95] Teahyung Lee and David V. Anderson. A spatially recursive optical flow estimation framework using adaptive filtering. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 789–792, Las Vegas, Nevada, March 2008.
- [96] Sheng-Yu Peng, Yu Tsao, Paul Hasler, and David V. Anderson. A programmable analog radial-basis-function based classifier. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1425–28, Las Vegas, Nevada, March 2008.
- [97] Brian Gestner, Wei Zhang, Xiaoli Ma, and David V. Anderson. VLSI implementation of a lattice reduction algorithm for low-complexity equalization. In *IEEE International Conference on Circuits and Systems for Communications*, Shanghai, May 2008.
- [98] Haw-Jing Lo, Heejong Yoo, and David V. Anderson. A reusable distributed arithmetic architecture for FIR filtering. In *Proceedings of the IEEE Midwest Symposium on Circuits and Systems*, pages 233–236, August 2008.
- [99] Brian Gestner and David V. Anderson. Single newton-raphson iteration for integer-rounded divider for lattice reduction algorithms. In *Proceedings of the IEEE Midwest Symposium on Circuits and Systems*, pages 966–969, August 2008.
- [100] Harry B. Marr, Jason George, David V. Anderson, and Paul Hasler. Increased energy efficiency and reliability of ultra-low power arithmetic. In *Proceedings of the IEEE Midwest Symposium on Circuits and Systems*, pages 366–369, August 2008.
- [101] Nikolaos Vasiloglou, Alexander G. Gray, and David V. Anderson. Scalable semidefinite manifold learning. In *IEEE Workshop on Machine Learning for Signal Processing*, pages 368–373, Cancun, Mexico, October 2008.
- [102] Bo Marr, Stephen Brink, Paul Hasler, and David Anderson. A reconfigurable analog system for efficient stochastic biological computation. In *IEEE Biomedical Circuits and Systems Conference*, pages 293–296, Baltimore, Maryland, November 2008.
- [103] Nikolaos Vasiloglou, Alexander G. Gray, and David V. Anderson. Learning isometric separation maps. In *NIPS Workshop on Kernel Learning: Automatic Selection of Optimal Kernels*, Vancouver, British Columbia, December 2008.

- [104] Nikolaos Vasiloglou, David V. Anderson, and Alexander Gray. Learning the intrinsic dimensions of the TIMIT speech database with maximum variance unfolding. In *IEEE Digital Signal Processing Workshop*, Marco Island, Florida, January 2009.
- [105] Jungwon Lee and David V. Anderson. Wavelet-based embedded image coding using dynamic index reordering vector quantization. In *IEEE Digital Signal Processing Workshop*, Marco Island, Florida, January 2009.
- [106] Nikolaos Vasiloglou, Alexander Gray, and David V. Anderson. Non-negative matrix factorization, convexity and isometry. In *The 2009 SIAM International Conference on Data Mining*, Sparks, Nevada, April 2009.
- [107] Haw-Jing Lo and David V. Anderson. A hardware-efficient implementation of the fast affine projection algorithm. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, April 2009.
- [108] Walter Huang and David V. Anderson. Adaptive filters using modified sliding-block distributed arithmetic with offset binary coding. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, April 2009.
- [109] Brian Gestner, Wei Zhang, Xiaoli Ma, and David V. Anderson. VLSI implementation of an effective lattice reduction algorithm with fixed-point considerations. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, April 2009.
- [110] Krishna T. Malladi and David V. Anderson. Analog implementation of SNR based gain adaptation for denoising. In *Proceedings of the International Symposium on Circuits and Systems*, pages 269–272, May 2009.
- [111] Bo Marr, Brian Degnan, Paul Hasler, and David Anderson. An asynchronously embedded datapath for performance acceleration and energy efficiency. In *Proceedings of the International Symposium on Circuits and Systems*, pages 3046–3049, May 2009.
- [112] Nikolaos Vasiloglou, Alexander G. Gray, and David V. Anderson. Learning isometric separation maps. In *IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6, September 2009.
- [113] Leung Kin Chiu and David V. Anderson. Psychoacoustic bass enhancement system on reconfigurable analog signal processor. In *52nd IEEE International Midwest Symposium on Circuits and Systems*, pages 164–167, August 2009.
- [114] Devangi N. Parikh, Sourabh Ravindran, and David V. Anderson. Gain adaptation based on signal-to-noise ratio for noise suppression. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 185–188, October 2009.

- [115] Faik Baskaya, David V. Anderson, Paul Hasler, and Sung Kyu Lim. A generic reconfigurable array specification and programming environment (grasper). In *Circuit Theory and Design, 2009. ECCTD 2009. European Conference on*, pages 619–622, August 2009.
- [116] Jorge Marin-Hurtado and David V. Anderson. Distortions in speech enhancement due to block processing. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Dallas, TX, March 2010.
- [117] Devangi Parikh, Muhammad Ikram, and David V. Anderson. Implementation of blind source separation and a post-processing algorithm for noise suppression in cell-phone applications. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Dallas, TX, March 2010.
- [118] David V. Anderson. A modulation view of audio processing for reducing audible artifacts. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Dallas, TX, March 2010.
- [119] Jason George, Bo Marr, Aniruddha Dasgupta, and David V. Anderson. Fixed-point arithmetic on a budget: Comparing probabilistic and reduced-precision addition. In *53rd IEEE International Midwest Symposium on Circuits and Systems*, pages 1258–1261, August 2010.
- [120] Devangi Parikh and David V. Anderson. Perceptual artifacts in speech noise suppression. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, pages 99–103, Pacific Grove, CA, November 2010.
- [121] Visar Berisha, Ailar Javadi, Richard Hammett, David V. Anderson, and Alex Gray. Making decisions about unseen data: Semi-supervised learning at different levels of specificity. In *Proceedings of the Asilomar Conference on Circuits, Systems, and Computers*, pages 75–79, Pacific Grove, CA, November 2010.
- [122] Devangi N. Parikh and David V. Anderson. Blind source separation with perceptual post processing. In *Digital Signal Processing Workshop and 6th IEEE Signal Processing Education Workshop, 2011. DSP/SPE 2011. IEEE 14th*, pages 321–325, January 2011.
- [123] Leung Kin Chiu, David V. Anderson, and Benjamin Hoomes. Audio output enhancement algorithms for piezoelectric loudspeakers. In *Digital Signal Processing Workshop and 6th IEEE Signal Processing Education Workshop, 2011. DSP/SPE 2011. IEEE 14th*, pages 317–320, January 2011.
- [124] Leung Kin Chiu, Brian Gestner, and David V. Anderson. Design of analog audio classifiers with adaboost-based feature selection. In *Proceedings of the International Symposium on Circuits and Systems*, May 2011.
- [125] P. Spencer Whitehead and David V. Anderson. Robust Bayesian analysis applied to Wiener filtering of speech. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2011.

- [126] Leung Kin Chiu, N.V. Parrish, and D.V. Anderson. A perceptually transparent audio power reduction algorithm for loudspeaker power management. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 325–328, May 2011.
- [127] Brian Gestner, Xiaoli Ma, and David V. Anderson. Are all basis updates for lattice-reduction-aided MIMO detection necessary? In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2011.
- [128] Jorge Marin-Hurtado, Devangi Parikh, and David Anderson. Binaural noise-reduction method based on blind source separation and perceptual post processing. In *Interspeech 2011*, August 2011.
- [129] Jorge I. Marin and David Anderson. Perceptually-inspired processing for multichannel wiener filter. In *Interspeech 2011*, August 2011.
- [130] J.I. Marin-Hurtado and D.V. Anderson. Reduced-bandwidth and low-complexity multichannel wiener filter for binaural hearing aids. In *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2011 IEEE Workshop on*, pages 85–88, October 2011.
- [131] J.I. Marin-Hurtado and D.V. Anderson. Robust non-vad implementation of multichannel wiener filter for binaural noise reduction. In *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2011 IEEE Workshop on*, pages 341–344, October 2011.
- [132] A.A. Kressner, D.V. Anderson, and C.J. Rozell. Robustness of the hearing aid speech quality index (hasqi). In *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2011 IEEE Workshop on*, pages 209–212, October 2011.
- [133] R. Curtin, N. Vasiloglou, and D.V. Anderson. Learning distances to improve phoneme classification. In *Machine Learning for Signal Processing (MLSP), 2011 IEEE International Workshop on*, September 2011.
- [134] A. Javadi, A. Gray, D. Anderson, and V. Berisha. Semi-supervised hierarchy learning using multiple-labeled data. In *Machine Learning for Signal Processing (MLSP), 2011 IEEE International Workshop on*, September 2011.
- [135] J.I. Marin-Hurtado and D.V. Anderson. Practical mwf-based noise-reduction methods for binaural hearing aids. In *Image, Signal Processing, and Artificial Vision (STSIVA), 2012 XVII Symposium of*, pages 84–91, 2012.
- [136] S Harbert, DF Britton, WDR Daley, DV Anderson, G Colon, M Giannelli, and E Hanson. Development of an audio and video observation and recording platform for data collection in a broiler growout environment. In *American Society of Agricultural and Biological Engineers Annual International Meeting 2012, ASABE 2012*, volume 4, pages 3235–3244, Jan 2012.

- [137] RR Curtin, WB March, P Ram, DV Anderson, AG Gray, and CLI Jr. Tree-independent dual-tree algorithms. In *2013 IEEE International Conference on Machine Learning*, Apr 2013.
- [138] AA Kressner, DV Anderson, and CJ Rozell. A novel binary mask estimator based on sparse approximation. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7497–7501, 2013.
- [139] NV Parrish, DV Anderson, WDR Daley, DF Britton, S Harbert, AB Webster, and CW Ritz. Analysis of poultry vocalizations for age estimation using cepstral coefficients and c4.5 classification. In *American Society of Agricultural and Biological Engineers Annual International Meeting 2013, ASABE 2013*, volume 4, pages 2949–2955, Jan 2013.
- [140] NH Sephus, TO Olubanjo, and DV Anderson. Enhancing online music lessons with applications in automating self-learning tutorials and performance assessment. In *Proceedings - 2013 12th International Conference on Machine Learning and Applications, ICMLA 2013*, volume 2, pages 568–571, 2013.
- [141] NH Sephus, AD Lanterman, and DV Anderson. Exploring frequency modulation features and resolution in the modulation spectrum. In *2013 IEEE Digital Signal Processing and Signal Processing Education Meeting, DSP/SPE 2013 - Proceedings*, pages 169–174, 2013.
- [142] V Duangudom and DV Anderson. Identifying salient sounds using dual-task experiments. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2013.
- [143] CM Chu, DV Anderson, and IEEE. Likelihood codebook reordering vector quantization. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5114–5117, 2013.
- [144] M Rizwan and DV Anderson. Using k-nearest neighbor and speaker ranking for phoneme prediction. In *Proceedings - 2014 13th International Conference on Machine Learning and Applications, ICMLA 2014*, pages 383–387, Dec 2014.
- [145] RR Curtin, W Daley, and DV Anderson. Classifying broiler chicken condition using audio data. In *2014 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2014*, pages 1141–1144, Jan 2014.
- [146] BT Carroll, DV Anderson, W Daley, S Harbert, DF Britton, and MW Jackwood. Detecting symptoms of diseases in poultry through audio signal processing. In *2014 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2014*, pages 1132–1135, Jan 2014.
- [147] BM Whitaker, BT Carroll, W Daley, and DV Anderson. Sparse decomposition of audio spectrograms for automated disease detection in chickens. In *2014 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2014*, pages 1122–1126, Jan 2014.

- [148] SH Raza, A Humayun, I Essa, M Grundmann, and D Anderson. Finding temporally consistent occlusion boundaries in videos using geometric context. In *Proceedings - 2015 IEEE Winter Conference on Applications of Computer Vision, WACV 2015*, pages 1022–1029, Feb 2015.
- [149] IA Khan, DV Anderson, and CF Barnes. Classification using residual vector quantization with markov-bayesian structure. In *Data Compression Conference*. Snowbird, Utah, Apr 2015.
- [150] Muhammad Rizwan and David V Anderson. Speaker adaptation using speaker similarity score on dnn features. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 877–882. IEEE, 2015.
- [151] Ayush Sharma and David V Anderson. Deep emotion recognition using prosodic and spectral feature extraction and classification based on cross validation and bootstrap. In *Signal Processing and Signal Processing Education Workshop (SP/SPE), 2015 IEEE*, pages 421–425. IEEE, 2015.
- [152] Jinwoo Kang, David V Anderson, and Monson H Hayes. Face recognition in vehicles with near infrared frame differencing. In *Signal Processing and Signal Processing Education Workshop (SP/SPE), 2015 IEEE*, pages 358–363. IEEE, 2015.
- [153] Bradley M Whitaker and David V Anderson. Learning anomalous features via sparse coding using matrix norms. In *Signal Processing and Signal Processing Education Workshop (SP/SPE), 2015 IEEE*, pages 196–201. IEEE, 2015.
- [154] Chu Meh Chu, Nathan V Parrish, and David V Anderson. Adaptive likelihood codebook reordering vector quantization for 1-d data sources. In *Signal Processing and Signal Processing Education Workshop (SP/SPE), 2015 IEEE*, pages 107–112. IEEE, 2015.
- [155] Jinwoo Kang, David V Anderson, and Monson H Hayes. Direct image alignment for active near infrared image differencing. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 334–344. Springer International Publishing, 2015.
- [156] Muhammad Rizwan, Babafemi O Odelowo, and David V Anderson. Word based dialect classification using extreme learning machines. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 2625–2629. IEEE, 2016.
- [157] Chieh-Feng Cheng, Abbas Rashidi, Mark A. Davenport, and David Anderson. Audio signal processing for activity recognition of construction heavy equipment. In *33rd International Symposium on Automation and Robotics in Construction (ISARC 2016)*, 2016.
- [158] K. L. Fair and D. V. Anderson. Sparse approximation on energy efficient hardware. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 4302–4307, July 2016.

- [159] B. M. Whitaker and D. V. Anderson. Heart sound classification via sparse coding. In *2016 Computing in Cardiology Conference (CinC)*, pages 805–808, September 2016.
- [160] C. F. Cheng, A. Rashidi, M. A. Davenport, D. V. Anderson, and C. A. Sabillon. Hardware and software requirements for acoustical monitoring of construction jobsites. In *Proc. Int. Workshop on Computing in Civil Engineering 2017*, 2017.
- [161] B. O. Odelowo and D. V. Anderson. Speech enhancement using extreme learning machines. In *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 200–204, October 2017.
- [162] B. O. Odelowo and D. V. Anderson. A mask-based post processing approach for improving the quality and intelligibility of deep neural network enhanced speech. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1134–1138, December 2017.
- [163] B. O. Odelowo and D. V. Anderson. A noise prediction and time-domain subtraction approach to deep neural network based speech enhancement. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 372–377, December 2017.
- [164] B. M. Whitaker, M. Rizwan, V. B. Aydemir, J. M. Rehg, and D. V. Anderson. AF classification from ECG recording using feature ensemble and sparse coding. In *2017 Computing in Cardiology (CinC)*, September 2017.
- [165] Muhammad Rizwan and David V. Anderson. Investigation on adaptive data condensation for exemplar based method in speech task. In *2017 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2017*, Nov 2017.
- [166] Babafemi Odelowo and David V. Anderson. A study of training targets for deep neural network-based speech enhancement using noise prediction. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.
- [167] Bradley M. Whitaker and David V. Anderson. Using block coordinate descent to learn sparse coding dictionaries with a matrix norm update. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2761–2765, April 2018.
- [168] C. A. Sabillon, A. Rashidi, B. Samanta, C. F. Cheng, M. A. Davenport, and D. V. Anderson. A productivity forecasting system for construction cyclic operations using audio signals and a Bayesian approach. In *Construction Research Congress (CRC)*, April 2018.
- [169] C. Cheng, D. V. Anderson, M. A. Davenport, and A. Rashidi. Audio classification based on weakly labeled data. In *2018 IEEE Statistical Signal Processing Workshop (SSP)*, pages 568–572, June 2018.
- [170] Babafemi O. Odelowo and David V. Anderson. Canonical ELM: Improving the performance of extreme learning machines on multivariate regression tasks

- using canonical correlations. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 734–740, December 2018.
- [171] Shoba S. Meera, Rahul Pawar, Kha vi Khuu, Steven Warren, Gizem Cetin, Vybhav Jagannath, Mark Clements, David Anderson, Heather Hazlett, Robert Schultz, Annette Estes, Stephen Dager, Kelly Botteron, Lonnie Zwaigenbaum, Julia Parrish-Morris, Juhi Pandey, Tanya St. John, Meghan Swanson, Linda Watson, and Joe Piven. Canonical babbling in 9-month-old infants later diagnosed with autism spectrum disorder: A naturalistic evaluation of all-day recordings. In *International Society for Autism Research (INSAR) 2019 Annual Meeting*, May 2019.
 - [172] Chuyao Feng, Eva van Leer, and David V. Anderson. Identification of voice quality variation using i-vectors. In *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 105–109, October 2019.
 - [173] Harish Haresamudram, David V Anderson, and Thomas Plötz. On the role of features in human activity recognition. In *Proceedings of the 23rd International Symposium on Wearable Computers*, pages 78–88, 2019.
 - [174] Devon Janke and David V. Anderson. Analyzing the effects of noise and variation on the accuracy of analog neural networks. In *2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 150–153, 2020.
 - [175] Bradley M. Whitaker and David V. Anderson. Exploring the effectiveness of linear matrix factorizations after nonlinear processing. In *2020 Intermountain Engineering, Technology and Computing (IETC)*, 2020.
 - [176] You Wang, Chuyao Feng, and David V. Anderson. A multi-channel temporal attention convolutional neural network model for environmental sound classification. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 930–934, 2021.
 - [177] Ranveer Uppal, Sanath Nagaraj, Eva van Leer, and David V. Anderson. Non-parametric online changepoint detection algorithm. In *2021 IEEE Statistical Signal Processing Workshop (SSP)*, pages 396–400, 2021.
 - [178] Devon Janke and David V. Anderson. Best practices for designing and training neural networks subjected to random variations. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021.
 - [179] Devon Janke and David V. Anderson. Improving robustness in analog neural networks by introducing sparse connectivity. In *Conference Record of the Asilomar Conference on Signals, Systems and Computers, 2021.*, October 2021.
 - [180] Dorsey Beckles, Desmond Caulley, Luca DeCicco, Chandler Mason, Zhaozhou Tang, and David V Anderson. Speaker diarization and automatic analysis methods of audio from individuals with autism spectrum disorder. In *2021*

IEEE International Conference on RFID (RFID) (IEEE RFID 2021), April 2021.

- [181] You Wang and David V. Anderson. Hybrid attention-based prototypical networks for few-shot sound classification. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 651–655, 2022.

C. Other Publications and Creative Products

C.1. Technical Reports

- [1] Heejong Yoo, Rich Ellis, David V. Anderson, Paul Hasler, David W. Graham, and Mat Hans. Continuous-time audio noise suppression and real-time implementation. Technical Report HPL-2002-311, HP Labs, November 2002. available at <http://www.hpl.hp.com/techreports>.
- [2] André van Schaik, David Anderson, Steven Greenberg, Malcom Slaney, et al. Final report—chapter on audio projects. Technical report, Telluride Workshop on Neuromorphic Engineering, Telluride, CO, July 2003.
- [3] Malcolm Slaney, David V. Anderson, André van Schaik, et al. Final report—chapter on audio projects. Technical report, Telluride Workshop on Neuromorphic Engineering, Telluride, CO, July 2004.
- [4] Nordine Sebkhi, Md Nazmus Sahadat, Erica Walling, Michelle Hoefnagel, Fulcher Chris, Fanpeng Kong, Geneva Tonuzi, Raine Osborne, Maysam Ghovanloo, and David Anderson. Evaluation of Multimodal Tongue Drive System by People with Tetraplegia for Computer Access. *IEEE TechRxiv*, April 2021.

D. Presentations

D.1. Conference Presentations without Proceedings

- [1] David V. Anderson and Mark A. Clements. Noise suppression in speech using multi-resolution sinusoidal modeling. presented at the Fall 1998 Meeting of the Acoustical Society of America, Norfolk, VA.
- [2] David V. Anderson, Douglas Chabries, and Richard W. Christiansen. Time constants in multi-band compressive gain hearing aids. In *International Hearing Aid Research Conference*, Lake Tahoe, CA, August 2000.
- [3] David V. Anderson, Phil Spencer Whitehead, and Mark A. Clements. Acoustic noise suppression for speech enhancement. presented at the Fall 2001 Meeting of the Acoustical Society of America, Ft. Lauderdale, FL, October 2001.
- [4] David V. Anderson. Fast dynamic range compression method for multichannel hearing aids. presented at the 143rd Meeting of the Acoustical Society of America, Pittsburgh, Pennsylvania, June 2002.
- [5] I. Faik Baskaya, Sasank Reddy, Tyson Hall, David V. Anderson, Paul Hasler, and Sung Kyu Lim. Analog circuit modeling and clustering for large scale FPAA. presented at the Design Automation and Test Conference, May 2004.

- [6] Abbas Rashidi, Mark A. Davenport, David V. Anderson, Chieh-Feng Cheng, and Chris A. Sabillon. Achievements and challenges in audio-based modeling of construction job sites. *The Meeting of the Acoustical Society of America*, 141:3917–3917, May 2017.

D.2. Seminar Presentations

- [1] David V. Anderson, Monson H. Hayes III, and Joel Jackson. Streaming multimedia course development. Invited seminar at Virginia Tech, December 2000.
- [2] David V. Anderson. Cooperative analog/digital signal processing. Georgia Tech Analog Consortium Industry Review, March 2001.
- [3] David V. Anderson. Analog signal processing in a digital world. Georgia Tech Analog Consortium Industry Review, March 2002.
- [4] David V. Anderson. Analog and signal processing. Georgia Tech Analog Consortium Industry Review, October 2003.
- [5] David V. Anderson. Cooperative analog/digital signal processing. Invited speaker: IEEE Georgia Tech Student Chapter meeting, March 2000.
- [6] Mat Hans and David V. Anderson. Sensor inputs for portable devices. Invited speaker: IEEE Georgia Tech Student Chapter meeting, February 2002.
- [7] David V. Anderson. Bucking the trend. Keynote speaker at HKN Honor Society induction, April 2002.
- [8] David V. Anderson. Prototyping cooperative analog-digital signal processing for auditory applications. Telluride Workshop on Neuromorphic Engineering, July 2003.
- [9] David V. Anderson. Neuro-inspired audio processing. Telluride Workshop on Neuromorphic Engineering, July 2004.
- [10] David V. Anderson. Signal processing trends. Invited speaker: Korea University, May 2005.
- [11] David V. Anderson. Low-power signal processing trends. Invited speaker: Johns Hopkins University, November 2005.
- [12] David V. Anderson. Signal processing with analog VLSI. Invited speaker: State University of New York – Stony Brook, January 2006.
- [13] David V. Anderson. Neuro-inspired signal processing. Telluride Workshop on Neuromorphic Engineering, July 2005.
- [14] David V. Anderson. Audio signal enhancement. Telluride Workshop on Neuromorphic Engineering, July 2006.
- [15] David V. Anderson. Human perception and signal processing. Telluride Workshop on Neuromorphic Engineering, July 2008.

- [16] David V. Anderson. Bayesian probability in everyday life. Presentation to Autry Mill Middle School Math Classes, May 2009.
- [17] David V. Anderson. Multimedia signal processing. Invited speaker: Korea University, July 2009.

E. Grants and Contracts

E.1. & E.2. As Principal and Co-Principal Investigator

	Sponsor	Title	Dates	\$ Funded	Contribution
1.	NSF	High Density Analog Computing Arrays	2000-04	\$443,000	Co-PI – 40%
2.	GTBI	Blind Source Separation for Audio	2001-02	\$26,000	PI – 100%
3.	GTBI	Audio Classification	2002-03	\$26,000	PI – 100%
4.	GTBI	Audio Classification	2003-04	\$20,000	PI – 100%
5.	GVU	Auditory Scene Analysis	2002-03	\$16,170	PI – 100%
6.	DARPA	Focal-Plane Image Enhancement	2002-04	\$375,000	Co-PI – 40%
7.	DARPA	Focal-Plane Image Processing	2004-06	\$900,000	Co-PI – 40%
8.	DARPA	Improved Speech Analysis, Coding, and Enhancement using Microradars	2002-06	\$2,275,955	Co-PI – 40%
9.	NSF	CAREER: Ultra-Low Power Programmable Analog Signal Processing Systems	2004-09	\$400,000	PI – 100%
10.	CIA	Low-power Array Processing	2004-05	\$383,200	PI – 100%
11.	NSF	Bringing Low Power Reconfigurable Analog Signal Processing to Embedded Systems	2004-07	\$240,000	Co-PI – 20%
12.	NSF	Probabilistic CMOS Computing and Applications	2007-10	\$762,000	PI – 100%

	Sponsor	Title	Dates	\$ Funded	Contribution
13.	Raytheon	Machine Learning of Visual Features	2009-10	\$85,000	Co-PI – 50%
14.	National Semiconductor	Hearing Aid Audio Processing	2009-12	\$150,000	PI – 100%
15.	National Semiconductor	Speaker Driver Sound Enhancement	2009-10	\$50,000	PI – 100%
16.	Army Research Office	Realizing Lattice-Reduction-Based Detectors for High-Rate Wireless Communications	2011-12	\$50,000	Co-PI – 50%
17.	NSF	I-CORPS	2012-13	\$50,000	PI – 100%
18.	Walmart Foundation	Craft with Pride: Development and Deployment of Disruptive Manufacturing	2014-16	\$2,959,656	CoI – 10%
19.	NSF	I-Corps: Automated Audio Monitoring	2018-19	\$50,000	PI – 100%
20.	NSF	I-Corps	2019-21	\$50,000	PI – 100%
21.	NSF	PFI-TT: Continuous Livestock Monitoring Using Audio and Machine Learning	2019-21	\$249,999	PI – 100%
22.	Georgia CTSA	GACTSA I-Vector Based Voice Quality	2018-19	\$33,892	PI – 100%
23.	GRA	Georgia Research Alliance Phase 1A (Ergense)	2020	\$30,000	PI – 100%
24.	GRA	AudioT	2020-2021	\$50,000	PI – 100%
25.	GRA	Phase IIA grant – AudioT	2022-2023	\$59,750	PI – 100%

27/42

	Sponsor	Title	Dates	\$ Funded	Contribution
26.	NSF	Collaborative Research: An Audio-based Spatio-temporal System for Automated Monitoring of Construction Operations	2015-18	\$208,909	Co-PI – 50%
27.	BMW Corp	Audio Analysis of Motor Sounds for Quality Control	2017-20	\$225,000	PI – 100%
28.	Brooks Foundation	Evaluation of a Multimodal Assistive System for Mobility by Combining Tongue	2019-20	\$30,064	PI – 100%
29.	NIH/GSU	Analysis of Voice Exercises Using Signal Processing	2017-18	\$11,700	PI – 100%
30.	GVU/IPAT	Acoustic Sensor Deployment in the EcoCommons	2019-20	\$20,000	Co-PI – 50%
31.	Blue Sky Research Grants	Parkinsons speech	2021-22	\$40,000	PI – 50%
32.	Cisco	Smart Homes for Effective and Safe Remote Work During a Pandemic and Beyond	2021-22	\$171,653	Co-PI – 50%
33.	GTRI	Facial Emotion Detection	2021-22	\$15,000	PI – 100%

Foundation Gifts

	Sponsor	Title	Dates	\$ Funded
1.	Google	Fast Analysis of High-dimensional Data	10/07	\$75,000
		With Alex Gray in CoC		
2.	Google	Audio Saliency	10/07	\$100,000
		With Alex Gray in CoC		
3.	National Semiconductor	Sound Classification Circuits Research	9/05–	\$125,000
4.	IDT Corporation	FFT Circuits Research	10/06	\$30,000

E.5. Proposals Submitted but Not Funded (Last Two Years)

	Sponsor	Title	Year	Budget	Contribution
1.	NSF	SCH: INT: Exploiting on-body measurements of physiological and contextual signatures for personalized anxiety trigger detection	2020	\$1,198,178	Co-PI – 33%
2.	NSF	SCH: INT: Improving the Patient Healing Process in the Hospital Setting via Transformational Sound and Light Sensing	2019	\$1,200,000	PI – 25%
3.	NSF	PFI-TT: Monitoring of Acoustic Scenes using Spatiotemporal Audio Circuits at near zero-power	2019	\$250,000	PI – 100%
4.	Imlay Foundation	Sound and Light Monitoring Device for Use in Improving Sleep in ICUs	2019	\$125,000	PI – 60%
5.	Simons Foundation	Integrating Novel Video and Acoustic Home-Based Outcome Measures in Autism	2019	\$207,826	PI – 100%

	Sponsor	Title	Year	Budget	Contribution
6.	NSF-NIFA	AI-Driven Risk Management in Agriculture and Food Systems: Resiliency from Cells to Watersheds	2020	\$3,771,681	Co-PI – 33%
7.	NSF	Next Generation Smart Microphone with Automated Monitoring	2020	\$250,000	PI – 100%
8.	NIH	Optimizing voice therapy efficacy and efficiency through conversation training and app support	2019	\$2,960,352	Co-PI – 15%
9.	NIH	Individualized Automated Voice Quality Assessment for Voice Therapy Adherence Feedback	2021	\$424,114	PI – 60%
10.	NSF-NIFA	AI Institute: Multiscale sustainable intensification through AI-enabled agriculture	2020	\$1,940,771	GT PI – 33%
11.	NBMC	Long-Term Wearable, Multimodal E-Tattoos for Real-Time Cognitive State Assessment of Pilots	2020	\$855,851	GT Co-PI – 16%

F. Other Scholarly and Creative Accomplishments

- 2011–2020 Co-founder of Ratrix Technologies with Xiaoli Ma (GT-ECE) and Drayt Avera. Ratrix Technologies develops high-throughput wireless data solutions. Received NSF SBIR Phase I and Phase II funding
- 2018–present Co-founder of AudioT with Brandon Carroll (GT PhD grad) and Tom Darbonne. AudioT develops audio surveillance systems for use in livestock operations. Received FFAR Phase I and Phase II funding

F.1. Patents

- [1] David V. Anderson, Kwan Truong, and Stephen McGrath. Adaptive filter featuring spectral gain smoothing and variable noise multiplier for noise

reduction, and methods therefor. U.S. Patent no. 6,351,731, July 1999. Issued 2002.

- [2] David V. Anderson, Kwan Truong, and Stephen McGrath. Speech activity detector for use in noise reduction system, and methods therefor. U.S. Patent no. 6,453,285, July 1999. Issued 2002.
- [3] Jeffery Dugger, Tyson S. Hall, Paul Hasler, David V. Anderson, Paul D. Smith, Matthew R. Kucic, and Abhishek Bandyopadhyay. Floating-gate analog circuit. U.S. Patent no. 6,898,097, May 2005.
- [4] Daniel J. Allred, David V. Anderson, Walter G. Huang, Venkatesh Krishnan, and Heejong Yoo. Distributed arithmetic adaptive filter and method. U.S. Patent application no. 20050201457, September 2005.
- [5] Philomena Cleopha Brady, Haw-Jing Lo, Guillermo Jose Serrano, Farhan Adil, Matthew R. Kucic, Paul Hasler, David V. Anderson, and Angelo Pereira. Floating-gate reference circuit. U.S. Patent no. 7,034,603, April 2006.
- [6] Erhan Ozalevli, Paul Hasler, David V. Anderson, and Walter G. Huang. Reconfigurable mixed-signal VLSI implementation of distributed arithmetic. U.S. Patent no. 7,348,909, March 2008.
- [7] Richard T. Ellis, Heejong Yoo, David W. Graham, Paul E. Hasler, and David V. Anderson. Analog audio signal enhancement system using a noise suppression algorithm. U.S. Patent no. 7,590,250, September 2009.
- [8] David Verl Anderson, Brian Gestner, and Xiaoli Ma. Incremental lattice reduction systems and methods. U.S. Patent no. 8,948,318 B2, February 2015.

G. Societal and Policy Impacts

- 2022–present Expert witness for *undisclosed*, represented by Alston & Bird in *pending*
- 2022–present Expert witness for Google LLC, represented by Quinn Emanuel Urquhart & Sullivan, LLP in Google LLC v. Sonos, Inc., Nos. 22-cv-04552 & -04553 (Northern District of California)
- 2022–present Expert witness for Google LLC, represented by Quinn Emanuel Urquhart & Sullivan, LLP in Certain Audio Players and Components Thereof; and Certain Audio Players and Components Thereof II, Nos. 337-TA-1329 & 337-TA-1330 (USITC)
- 2022–present Expert witness for Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc., represented by Kirkland & Ellis LLP in Staton Techiya, LLC and Synergy IP Corp. v. Samsung Elecs. Co. et al. , Nos. 2:21-cv-413-JRG, 2:22-cv-00053-JRG-RSP (Eastern District of Texas)
- 2022–present Expert witness for TVision, represented by Kecker, Van Nest & Peters LLP in The Nielsen Company (US), LLC v. TVision Insights, Inc., No. 1:21-cv-01592-CJB and No. 1:22-cv-00057-CJB (Delaware)

- 2022–present Expert witness for Promptu Systems Corporation, represented by Finnegan, Henderson, Farabow, Farrett & Dunner LLP in Promptu Systems Corporation v. Comcast Corporation and Comcast Cable Communications, LLC, No. 2:16-cv-06516-LDD (Eastern District of Pennsylvania)
- 2022–present Expert witness for Google, represented by Keker, Van Nest & Peters LLP in *Jawbone Innovations, LLC v. Google LLC*, No. 6:21-cv-00985 (Western District of Texas)
- 2022–present Expert witness for Google, represented by Finnegan, Henderson, Farabow, Farrett & Dunner LLP in Google LLC, v. Jawbone Innovations, LLC, Cases IPR2022-01027, IPR2022-00649, IPR2022-00623
- 2021–present Expert witness for Zentian, LTD, represented by Bartlit & Beck, LLP in *Zentian Ltd. v. Apple Inc.*; and *Zentian Ltd. v. Amazon.com Inc.* (Western District of Texas)
- 2020 Expert witness for St. Lawrence Communications LLC, represented by AZA Law LLC in *Saint Lawrence Communications LCC v. Amazon.com, Inc.*, No. 2:19-cv-00027-JRG (Eastern District of Texas) (deposed)
- 2020 Expert witness for Advanced Voice Recognition Systems, Inc., represented by Buether Joe & Carpenter, LLC in *IPR2019-01352* (deposed)
- 2020–2020 Expert witness for Motorola Solutions, Inc, represented by Kirkland and Ellis, LLC in *Motorola Solutions, Inc. v. Hytera Communications Corp. Ltd. et al.*, No. 1:17-CV-01972 (Northern District of Illinois, Eastern Division) (deposed)
- 2018–2020 Expert witness for Motorola Solutions, Inc, represented by Kirkland and Ellis, LLC in *Hytera Communications Corp. Ltd. v. Motorola Solutions, Inc.*, No. 1:17-CV-1794 (Northern District of Ohio, Eastern Division) (deposed twice)
- 2018–2021 Expert witness for MModal Services Ltd., represented by Latham & Watkins LLP in *MModal Services Ltd. v. Nuance Communications, Inc.*, No. 1:18-cv-00901-WMR (Northern District of Georgia Atlanta Division) (deposed)
- 2017–2018 Expert witness for Analytics for Life, Inc., represented by Meunier Carlin & Curfman LLC in IPR2017-01742 (deposed)
- 2017–2018 Expert witness for III Holdings, represented by Posinelli LLP in IPR2017-00930 (K/S HIMPP et al. v. III Holdings) (deposed)
- 2017–2018 Expert witness for Motorola Solutions, Inc, represented by Kirkland and Ellis, LLC in *In the Matter of Certain Two-Way Radio Equipment and Systems, Related Software and Components Thereof* before the International Trade Commission, Investigation No. 337-TA-1053 (deposed)
- 2017–2018 Expert witness for St. Lawrence Communications LLC, represented by AZA Law LLC in *Saint Lawrence Communications LCC v. Apple Inc., AT&T Mobility, and Cellco Partnership D/B/A Verizon Wireless Civil Action No. 2:16-cv-00082-JRG* (Eastern District of Texas) (deposed)
- 2016–2017 Expert witness for Samsung Corporation, represented by Covington Burlington, LLP in *In the Matter of Certain Audio Processing Hardware, Software, and Products Containing the Same* before the International Trade Commission, Investigation No. 337-TA-1026 (deposed)

- 2016–2017 Expert consultant for Alston and Bird, LLC on behalf of Nokia.
- 2015–2016 Expert witness for Acer, Inc.; Acer America Corp.; ASUS TeK Computer Inc.; ASUS Computer International; Dell Inc.; Hewlett Packard Co.; Lenovo Holding Co., Inc.; Lenovo (United States) Inc.; Toshiba Corp.; and Toshiba America Information Systems, Inc. in *In the matter of certain audio processing hardware and software products contain same* before the International Trade Commission, case 337-TA-949 (deposed)
- 2015 Expert witness for Realtek Semiconductor Corporation, represented by Steptoe & Johnson, LLP in *Realtek Semiconductor Corporation v. Andrea Electronics Corporation before the United States Patent and Trademark Office Patent Trial and Appeal Board*, Case Numbers IPR2015-01392, IPR2015-01393, IPR2015-01394, IPR2015-01395, IPR2015-01396
- 2015 Expert witness for Realtek Semiconductor Corporation, represented by Chen Malin, LLP in *In the matter of certain audio processing hardware and software products contain same* before the International Trade Commission, case 337-TA-949
- 2012–2013 Expert witness for MobileMedia Ideas, LLC, represented by Proskauer in *MobileMedia Ideas LLC, v. Research in Motion Limited and Research in Motion Corporation, Civil Action No. 3:11-cv-02353-N (Northern District of Texas)* (deposed)
- 2009–2013 Expert witness for Oticon, LLC, represented by Birch, Steward, Kolasch, & Birch, L.L.P. in *Oticon, Inc. v. Sebotek Hearing systems, LLC, et al., Civil Action No. 3:08-cv-05489-FLW (District of New Jersey)* and *Sound Design Technologies, Ltd. v Oticon, Inc., Civil Action No. 2:11-cv-01375-SRB (District of Arizona)* (deposed twice, testified in Markman hearing)
- 2007–2011 Expert Consultant, Alston and Bird, LLC performing internal analysis for various Nokia Patents
- 2011 Expert consultant for Alston and Bird, LLC on behalf of Nokia in *Nokia Corporation v. Apple Inc., Civil Action No. 1:09-cv-00791-GMS (District of Delaware)*
- 2011 Expert witness for Flightcom Corporation, represented by Alston and Bird, LLC in *TechnoFirst S.A. v. Flightcom Corp., Civil Action No. 2:05-cv-00411-HCM-FBS (Eastern District of Virginia)*
- 2011 Expert consultant for Alston and Bird, LLC on behalf of Nokia in *Nokia Corporation v. Apple Inc., Civil Action No. 1:09-cv-00791-GMS (District of Delaware)* (patent analysis)
- 2008 Technical Consultant, Sound Innovations, Inc., White River Junction, Vermont
- 2007–2008 Expert witness for Oticon, LLC and Bernafon, represented by Finnegan and Henderson in *Energy Transportation Group, Inc, v. Sonic Innovations, Inc., et al, Case 1:05-cv-00422-GMS (District of Delaware)* (non-infringement expert, deposed, testified before jury)
- 2007–2008 Technical Consultant, Personics Labs, Boca Raton, Florida

- 2005–2008 Technical Consultant and Technical Advisory Board member, GTronix, Inc., Fremont, California
- 1997–2000 Technical Consultant, ASPI, Inc., Atlanta, Georgia

H. Other Professional Activities

- 2008 Technical Consultant, Sound Innovations, Inc., White River Junction, Vermont
- 2007–2008 Technical Consultant, Personics Labs, Boca Raton, Florida
- 2005–2008 Technical Consultant and Technical Advisory Board member, GTronix, Inc., Fremont, California
- 1998–2000 Technical Consultant, ASPI, Inc., Atlanta, Georgia

V. Teaching

A. Courses Taught

Semester/year	Course	enrollment
Fall 2022	ECE 4872- Create-X Capstone Design	15
Summer 2022	ECE 6250- Adv Digital Signal Proc	35
Spring 2022	ECE 6250- Adv Digital Signal Proc	60
Spring 2022	ECE 2026- Intro to Signal Proc (advisor)	—
Spring 2022	ECE 4872- Create-X Capstone Design	15
Summer 2021	ECE 6250 - Adv Digital Signal Proc	52
Spring 2021	ECE 4270 - Fund-Digital Signal Proc	30
Spring 2021	ECE 4872 - Special Topics: Capstone Design 2	6
Spring 2021	ECE 6254 - Stat Machine Learning	86
Summer 2020	ECE 6250 - Adv Digital Signal Proc: Advanced Digital Signal P	9
Spring 2020	ECE 4271- DSP Applications	36
Spring 2020	ECE 4270- Fund-Digital Signal Proc	38
Summer 2019	ECE 3057 - Arch, Sys, Conc & Engy Comp	20
Spring 2019	ECE 2026 - Intro Signal Processing	95
Spring 2019	ECE 4012 - ECE Culminating Design 2	9
Fall 2018	ECE 4270 - Fund-Digital Signal Proc	44
Fall 2018	ECE 6250 - Adv Digital Signal Proc	37
Summer 2018	ECE 3077 - Prob/Stats for ECE	26
Summer 2018	ECE 6271 - Adaptive Filtering	33
Spring 2018	ECE 2026 - Intro Signal Processing	98
Spring 2018	ECE 6254 - Stat Machine Learning	132
Fall 2017	ECE 4012 - ECE Culminating Design 2	16
Spring 2017	ECE 2036 - Engr Software Design	13
Spring 2017	ECE 6250 - Adv Digital Signal Proc	13
Spring 2016	ECE 2036 - Engr Software Design	10
Spring 2016	ECE 6271 - Adaptive Filtering	11
Fall 2016	ECE 4270 - Fund-Digital Signal Proc	63
Fall 2016	ECE 6273 - Pattern Recognit-Speech	59

B. Individual Student Guidance

B.1.a. Ph.D. Students Graduated

2004 **Tyson S. Hall, Ph.D.**

Thesis: Field-Programmable Analog Arrays: A Floating-Gate Approach

Current: Professor at Southern Adventist University

2005 **Heejong Yoo, Ph.D.**

Thesis: Low-Power Audio Input Enhancements for Portable Devices

Current: IP department at Qualcomm, Inc.

- 2005 **Venkatesh Krishnan, Ph.D.**
 Thesis: A framework for low bit-rate speech coding in noisy environments
 Current: Multimedia research group at Qualcomm, Inc.
- 2006 **Cenk Demiroglu, Ph.D.**
 Thesis: Multisensor Segmentation-based Noise Suppression for Intelligibility Improvement in MELP Coders
 Current: Assistant Professor at Özyeğin University (Turkey)
- 2006 **Rongqiang (James) Hu, Ph.D.**
 Thesis: Multi-Sensor Noise Suppression and Bandwidth Extension for Enhancement of Speech
 Current: Nintendo
- 2007 **Sourabh Ravindran, Ph.D.**
 Thesis: Physiologically Motivated Methods for Audio Pattern Classification
 Current: Director, Mobile Processor Innovation Lab at Samsung
- 2007 **Shyam Subramanian, Ph.D.**
 Thesis: Methods for Synthesis of Multiple-Input Translinear Element Networks
 Current: Senior Principal Design Engineer at Cadence Design Systems
- 2007 **Teahyung Lee, Ph.D.**
 Thesis: Algorithm-based Efficient Approaches for Motion Estimation Systems
 Current: Intel Research
- 2009 **Nikolaos Vasiloglou, Ph.D.**
 Thesis: Isometry and Convexity in Dimensionality Reduction
 Current: Entrepreneur and Consultant in Machine Learning
- 2009 **Haw Jing (Michael) Lo, Ph.D.**
 Thesis: Design of a Reusable Distributed Arithmetic Filter and its Application to the Affine Projection Algorithm
 Current: Qualcomm, Inc.
- 2009 **Ismail Faik Baskaya, Ph.D.**
 Thesis: Physical Design Automation for Large Scale Field Programmable Analog Arrays
 Current: Assistant Professor at Boğaziçi University, Istanbul (Turkey)
- 2009 **Walter Huang, Ph.D.**
 Thesis: Implementation of Adaptive Digital FIR and Reprogrammable Mixed-Signal Filters using Distributed Arithmetic

Current: Qualcomm, Inc.

2009 **Jungwon Lee, Ph.D.**

Thesis: Efficient Image Compression System with a CMOS Transform Imager

Current: Senior Engineer, Samsung Electronics, Korea

2009 **Harry (Bo) Marr, Ph.D.**

Co-advisor: Jennifer Hasler

Thesis: Learning, Probabilistic, and Asynchronous Technologies for an Ultra Efficient Datapath

Current: Director of Low Power Computing Group Raytheon at Raytheon, Inc.

2011 **Brian Gestner, Ph.D.**

Thesis: Lattice Reduction for MIMO Detection: From Theoretical Analysis to Hardware Realization

Current: Chief Technology Officer at Soneter

2012 **Jason George, Ph.D.**

Thesis: Harnessing Resilience: Biased Voltage Overscaling for Probabilistic Signal Processing

Current: Entrepreneur

2012 **Leung Kin Chui, Ph.D.**

Thesis: Efficient audio signal processing for embedded systems

Current: Design Engineer at Texas Instruments

2012 **Jorge Marin, Ph.D.**

Thesis: Robust binaural noise-reduction strategies with binaural-hearing-aid constraints: design, analysis and practical considerations

Current: Professor at Universidad del Quindío

2012 **Devangi Parikh, Ph.D.**

Thesis: Improving the quality of speech in noisy environments

Current: Biomedical Research Lab at Texas Instruments

2012 **Varinthira Duangudom, Ph.D.**

Thesis: Computational auditory saliency

Current: Homemaker

2013 **Irteza Syed, Ph.D.**

Thesis: Classification using Residual Vector Quantization

Current: Aero, Inc, Pakistan

- 2014 **Syed Hussain Raza, Ph.D.**
 Thesis: Temporally Consistent Semantic Segmentation in Videos
 Current: NVIDIA
- 2014 **Nashie Sephus, Ph.D.**
 Thesis: A Framework for Exploiting Modulation Spectral Features in Music Data Mining and Other Applicaitons
 Current: CTO at Partpic, Inc.
- 2015 **Ryan Curtin, Ph.D.**
 Thesis: Improving Dual-Tree Algorithms
 Current: relational.ai
- 2015 **Chu Meh Chu, Ph.D.**
 Thesis: Exploiting Temporal and Spatial Redundancies for Vector Quantization of Speech and Images
 Current: Luxoft
- 2016 **Jinwoo Kang, Ph.D.**
 Thesis: Face Recognition for Vehicle Personalization
- 2017 **Kaitlin Fair, Ph.D.**
 Thesis: A Biologically Plausible Sparse Approximation Solver on Neuromorphic Hardware
 Current: Air Force Research Lab
- 2017 **Muhammad Rizwan, Ph.D.**
 Thesis: Adaptation of Hybrid Deep Neural Network–Hidden Markov Model Speech Recognition System using a Sub-space Approach
 Current: Assistant Professor in Pakistan
- 2018 **Nathan Parrish, Ph.D.**
 Thesis: System Configuration for Proportional Control of an Assistive Technology for Patients with Cervical Spinal Cord Injuries
- 2018 **Brandon Carroll**
 Thesis: Characterizing Acoustic Environments with OLAF and ELSA
 Current: Georgia Tech (post-doc)
- 2018 **Bradley Whitaker**
 Thesis: Modifying Sparse Coding for Imbalanced Classification
 Current: Assistant Professor at Montana State University

- 2018 **Babafemi Odelowo**
 Thesis: Development of a Neural Network-Based Speech Enhancement System
 Current: Georgia Tech Research Institute
- 2019 **Md Nazmus Sahadat**
 Thesis: Design and Evaluation of a Multimodal Assistive Technology using Tongue Commands, Head Movements, and Speech Recognition for People with Tetraplegia
 Current: Starkey Corporation
- 2019 **Chieh-Feng Cheng**
 Thesis: Audio Classification for Small-sized and Weakly Labelled Data
- 2021 **Devon Janke**
 Thesis: Overcoming Process Variations and Noise in Analog Neural Networks
 Current: IBM Corporation
- 2021 **Lee Richert**
 Thesis: Multi-Layer Dictionary Learning Using Low-Rank Updates
 Current: Army Research Lab
- 2022 **You Wang**
 Thesis: Attention-based Convolutional Neural Network Model and Its Combination with Few-Shot Learning for Audio Classification
 Current: Qualcomm
- 2022 **Chuyao Feng**
 Thesis: Intra-Speaker Voice Quality Recognition for Voice Therapy
 Current: Pearson
- 2022 **Desmond Caulley**
 Thesis: Improved Automatic Analysis Methods for Lena-Obtained Audio Recordings of Children with Autism Spectrum Disorder
 Current: TQ Intelligence
- [B.1.b. Ph.D. Students in Progress](#)
- 2019–present **Mohamad Salut**—Passed Proposal 2021. Thesis title: Randomized Online Tensor Robust Principal Component Analysis
- 2019–present **Nicolas Shu**—Passed Prelim Exam 2021. (co-advised) Thesis title: Generalized Real-Time Indoor Monitoring System via Audio applied to Interaction Analysis
- 2020–present **Daniel Dicheck**. Thesis area: System Performance Analysis based on Anomaly Detection
- 2021–present **Shuvo Newaz**. Thesis area: Emotion recognition in video

B.2.a. M.S. Students Graduated (with Thesis)

2002 **Hyung K. Choi**

Thesis: Blind Source Separation of the Audio Signals in a Real World

2003 **Paul Hultz**

Thesis: Backward Masking in the Human Auditory System

2006 **Daniel Allred**

Thesis: Evaluation and Comparison of Beamforming Algorithms for Microphone Array Speech Processing

2012 **Ailar Javadi**

Thesis: Bio-inspired noise robust auditory features

2012 **Guillermo Colón**

Thesis: Avian musing feature space analysis

2016 **Tushar Supe**

Thesis: Super-CORDIC: Low Delay CORDIC Architectures for Computing Complex Functions

2019 **Harish K Haresamudram**

Thesis: The Role of Representations in Human Activity Recognition

2020 **Maham Tanveer**

Thesis: Learning Distance Metrics for High-Dimensional Data

2020 **Yufeng Yeng**

Thesis: Speaker Recognition and Diarization

2020 **Hua Wang**

Thesis: Noise Suppression in Speech

B.3. Undergraduate Research Students Supervised

- Supervised over 100 undergraduate research projects.

B.5. Mentorship of Postdoctoral Fellows or Visiting Scholars

2018–2021 Brandon Carroll: Audio machine learning for livestock monitoring

C. Other Teaching Activities

Graduate Course Development

- Human Perception and Signal Processing: Advanced Topics in Signal Processing, ECE 7252. The objective of this course is to explore human-centric applications of signal processing. Students learn about compression standards such as MPEG audio and video, perceptually relevant error measures, subjective and objective quality measures, hearing impairments and compensation, signal enhancement, and current research topics in signal processing and human perception.

Short Course Development

- “Digital Signal Processing for Practicing Engineers” - a 12-week on-line course delivered: February 2000, ongoing approximately twice per year.
- “Streaming Multi-Media Production” One-day course delivered: May 2000
- “Finite-Precision Signal Processing” Four-day course delivered: Spring 2007, Summer 2007, Spring 2009.
- “Applications of Psychoacoustics to Signal Processing” One-day tutorial delivered: Spring 2009

VI. Service

A. Professional Contributions

A.1. Society Offices, Activities, and Membership

1991–present	Senior Member IEEE (Institute of Electrical and Electronics Engineers) Member IEEE Signal Processing Society Member IEEE Education Society Member IEEE Communications Society
1994–2016	Member ASA (Acoustical Society of America)

A.2. Organization and Chairmanship of Technical Sessions, Workshops, and Conferences

2014	Treasurer, IEEE Global Conference on Signal and Information Processing (GlobalSIP)
2010	Technical Area Chair for the Asilomar Conference on Signals, Systems, and Computers
2003	Co-Chair for the NSF Symposium on Next Generation Automatic Speech Recognition
2002	Local Arrangements Chair for the DSP Workshop
2002	Local Arrangements Chair for the Signal Processing Education Workshop
2002	Special Session Co-Chair at ICASSP 2002, Orlando, FL, May 2002. Session Title: “Cooperative Analog-Digital Signal Processing”
2000	Technical Co-Chair for Signal Processing Education Workshop

A.3 Other

2011–2019	ECE Advisory Board, Rose-Hulman Institute of Technology
1999–present	Reviewer for numerous journals and conference publications

B. Public Community Service

2006–2015 Judge for the Georgia FIRST Lego League State Championships robotics competition for middle school students

C. Institute Contributions

C.1. Institute Service

2018–2021 Member, Georgia Tech Institute Research Faculty Promotions Committee
2018–present Member, Georgia Tech Faculty Senate
2016–present Faculty Advisor for Eagles @ GT student club
2011–2014 Chair, Institute Faculty Status Grievance Committee, Georgia Institute of Technology
2010–2016 Member, Institute Faculty Status Grievance Committee, Georgia Institute of Technology
2005–2008 Member, Institute Graduate Committee, Georgia Institute of Technology
2005–2008 Member, Georgia Tech Faculty Senate, Georgia Institute of Technology

C.2. School Service

2022–present Member, ECE Statutory Advisory Committee
2020–present Chair, ECE Graduate Committee
2017–2020 Member, ECE Graduate Student Recruiting Committee, Georgia Institute of Technology
2015–present Member, ECE Graduate Committee
2020–2022 Associate Member, ECE Statutory Advisory Committee
2015–2019 Member, ECE Statutory Advisory Committee
2014–2015 Chair, Computer Systems and Software area in Electrical and Computer Engineering, Georgia Institute of Technology
2009–2014 Member, ECE Graduate Student Recruiting Committee, Georgia Institute of Technology
2006–2007 Associate Director, Center for Research in Embedded Systems Technology
2004–2007 Associate Director over Education Outreach for the Center for Research on Embedded Systems (CREST)
2001–2009 Member, ECE Graduate Committee, Georgia Institute of Technology
1999–2001 Member, ECE Computing Committee, Georgia Institute of Technology

Exhibit 7

An Application of Recurrent Neural Networks to Discriminative Keyword Spotting

Santiago Fernández¹, Alex Graves¹, and Jürgen Schmidhuber^{1,2}

¹ IDSIA, Galleria 2, 6928 Manno-Lugano, Switzerland

² TU Munich, Boltzmannstr. 3, 85748 Garching, Munich, Germany

Abstract. The goal of keyword spotting is to detect the presence of specific spoken words in unconstrained speech. The majority of keyword spotting systems are based on generative hidden Markov models and lack discriminative capabilities. However, discriminative keyword spotting systems are currently based on frame-level posterior probabilities of sub-word units. This paper presents a discriminative keyword spotting system based on recurrent neural networks only, that uses information from long time spans to estimate word-level posterior probabilities. In a keyword spotting task on a large database of unconstrained speech the system achieved a keyword spotting accuracy of 84.5 %.

1 Introduction

The goal of keyword spotting is to detect the presence of specific spoken words in (typically) unconstrained speech. Applications of keyword spotting include audio indexing, detection of command words in interactive environments and spoken password verification. In general, it is most useful in domains where a full speech recogniser is cumbersome and unnecessary, partly because less than perfect detection rates are still very satisfactory.

Nonetheless, the same mathematical framework used in the majority of speech recognisers also forms the basis for keyword spotting systems. The typical keyword spotting system consists of a set of hidden Markov models (HMM), one for each keyword plus one or more filler models [1]. The filler models characterize non-keyword events in the speech signal, such as other words, background noises and silence. The approach is generative and therefore finds the sequence of models most likely to have produced the observations. The output of the system is post-processed before deciding on the presence or absence of keywords in the utterance [2]. First, the confidence of the predictions is estimated. This is typically done by computing the ratio of likelihoods between keyword model hypotheses and filler model hypotheses. Finally, a threshold level is applied to the confidence measure in order to achieve a compromise between the number of true and false positives predicted by the system.

In general, a discriminative approach to keyword spotting is more suitable, as it allows discrimination between keyword and non-keyword events, and also among similar keywords. In addition, posterior probabilities can directly be

used as an effective confidence measure. Recently, discriminative keyword spotting systems have been explored in various papers. The most common approach [3,4,5] uses artificial neural networks (ANN) in the framework of hybrid ANN/HMM systems [6,7]. In these systems, keywords are modelled by concatenating phoneme models. A phoneme classifier (typically a multi-layer perceptron) estimates frame-level posteriors. These local posterior estimates are accumulated over the duration of a segment and, typically, the result is normalised by the length of the segment. Another approach to discriminative keyword spotting uses kernel machines and large margin classifiers with a set of models and feature functions consistent with hybrid ANN/HMM systems [8].

The approach presented in this paper attempts to model full keywords in the sequence data stream, while previous discriminative keyword spotting systems are based on sub-word units. It is based on recurrent neural networks (RNNs) trained with the connectionist temporal classification (CTC) objective function. This objective function allows artificial neural networks to map unsegmented sequential data onto a sequence of labels [9,10]. The proposed system uses information over long time spans, thereby providing non-local estimates of the a posteriori probability of the presence of either keyword or non-keyword events in the speech signal.

There is a plethora of different keyword spotting systems, often tailored to meet the requirements of particular tasks. However, there is little consensus on how to define benchmark tasks [11]. We have opted to tackle a realistic keyword spotting task in a large database of spontaneous and unconstrained speech where an HMM-based speech recogniser achieves a word accuracy of only 65%.

The remainder of the paper is structured as follows. Section 2 provides a description of the system. Keyword spotting experiments are presented in section 3 and the results are shown in section 3.4. Section 4 offers a discussion on differences between previous approaches and the one presented in this paper, and gives directions for future work. Final conclusions are given in section 5.

2 Method

2.1 Outline

The network architecture selected for the keyword spotting task is the bi-directional long short-term memory recurrent neural network (BLSTM), which has shown good performance in a series of speech tasks [12,13]. The network is trained with the connectionist temporal classification (CTC) objective function [9,10].

The input to the recurrent network is the data sequence of a speech utterance. The network has a soft-max output layer with as many output units as keywords to be detected, plus one output unit associated with non-keyword events. The target for the training algorithm is a list (which may be empty) of keywords in the order in which they appear in the input speech utterance. No further constraints are applied to the system. In particular, the segmentation of the speech signal is not required and various pronunciation variants of the same keyword can appear in the data set.

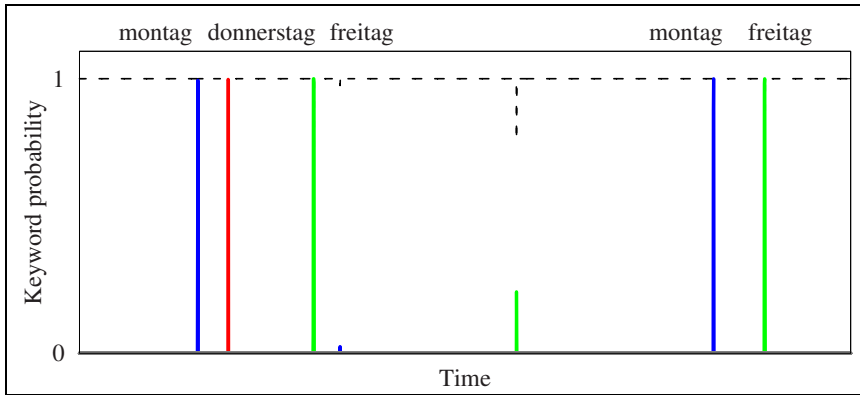


Fig. 1. Discriminant keyword spotting with BLSTM and CTC. When a keyword is detected the ANN produces a spike-like output with higher probability than the non-keyword output (dashed line).

Once the network has been trained, its output typically consists of a series of spikes corresponding to keyword events that have been detected in the input, separated by long periods of activation of the non-keyword output unit (see figure 1). The activation of every output unit at every time step is an estimate of the probability of detecting a particular keyword (or non-keyword event) at that time step.

2.2 BLSTM

The long short-term memory (LSTM) [14,15] is an RNN architecture designed to deal with long time-dependencies. It addresses the problem of the back-propagated error either blowing up or decaying exponentially for long time lags in conventional RNNs. The hidden layer of an LSTM network consists of a set of recurrently connected blocks containing one or more memory cells and three multiplicative units (the input, output and forget gates), which allow writing, reading or resetting of the information in the memory cell.

Bi-directional RNNs [16] address in an elegant way the need for delayed decisions in some sequential tasks such as speech processing. Data sequences are presented forwards and backwards to two separate recurrent networks, which are connected to the same output layer. Therefore, for every point in a given sequence, the network has complete sequential information about the points before and after it. An implementation of bi-directional LSTM (BLSTM) can be found in [12].

2.3 CTC

Connectionist temporal classification (CTC) [9,10] is an objective function to label unsegmented sequential data with RNNs. The basic idea behind CTC is to interpret the network outputs as a probability distribution over all possible label

sequences, conditioned on the input data sequence. Given this distribution, an objective function can be derived that directly maximises the probability of the correct labelling. Since the objective function is differentiable, the network can be trained with standard back-propagation through time.

3 Experiments

3.1 Material

The experiments were carried out on a set of dialogues in German from the Verbmobil database [17]. The dialogues deal with the scheduling of date appointments for meetings. The database consist in spontaneous and unconstrained speech, with long silences and noises. The results provided along with the database for a baseline automatic speech recogniser give an approximate idea of the difficulty of processing this database: the HMM-based speech recogniser achieves a word accuracy of 65 % in an automatic, full transcription, of the test set [17].

The material used in the experiments corresponds to version 2.3 (March 2004) of the Verbmobil database (except for CD-ROM number 53.1 of the training set which was not available) [17]. It includes separate training, validation and test sets. The database is speaker independent. Speakers were distributed equally across sexes in all sets and every speaker appears in only one of the sets. The training set includes 748 speakers and 23975 dialogue turns for a total of 45.6 hours of speech. The validation set includes 48 speakers and 1222 dialogue turns for a total of 2.9 hours of speech. The test set includes 46 speakers and 1223 dialogue turns for a total of 2.5 hours of speech.

Due to the nature of the dialogues included in the database, we decided to use dates and places as keywords for the detection task. This ensures a relatively good coverage of keywords in the training, validation and test data sets. The twelve keywords chosen were:

april, august, donnerstag, februar, frankfurt, freitag, hannover, januar,
juli, juni, mittwoch, montag

Note that the database includes various pronunciation variants of some of these keywords (e.g. “montag” can end either with a /g/ or with a /k/). In addition, several keywords appear as sub-words, e.g. in plural form such as “montags” or as part of another word such as “ostermontag” (Easter Monday).

The orthographic transcription provided along with the database was examined for the presence in every utterance of one or more of the twelve keywords selected. Once a keyword was found, the begin and end times for the keyword were saved for evaluating the performance of the keyword spotting system at a later stage. These times were provided along with the database and correspond to the segmentation given by an automatic speech recognition system.

This procedure gave a total of 10469 keywords on the training set with an average of 1.7 % keywords per non-empty utterance (73.6 % of the utterances did not have any keyword); 663 keywords on the validation set with an average

of 1.7% keywords per non-empty utterance (68.7% of the utterances did not have any keyword); and 620 keywords on the test set with an average of 1.8% keywords per non-empty utterance (71.1% of the utterances did not have any keyword).

The list of keywords (which may be empty) in the order in which they appear in every utterance, forms the target sequence for the network's training algorithm.

Finally, the speech data was characterised as a sequence of vectors of thirty nine coefficients, consisting of twelve Mel-frequency cepstral coefficients (MFCC) plus energy and first and second order derivatives of these magnitudes. The coefficients were computed every 10 ms over 25 ms-long windows. First, a Hamming window was applied; secondly, a mel-frequency filter bank of 26 channels was computed; and finally, MFCC coefficients were calculated with a 0.97 preemphasis coefficient. The input data was normalised to have zero mean and standard deviation one on the training set.

3.2 Difficulty with HMMs

We briefly illustrate the difficulty of performing keyword spotting with a generative HMM with as few constraints as those required by our system. In particular, no post-processing is applied. For these experiments every keyword was modelled as a left-to-right HMM with sixteen states and observation probabilities given by a mixture of Gaussians with diagonal covariance matrices. For the filler model we used an HMM of the same type with three states. The grammar allows one or more repetitions of any keyword or filler per utterance.

We experimented with: a) doubling the number of mixtures from two until a maximum of 128 and re-estimating the parameters twice after every step; b) the parameters of the HMMs with eight Gaussians were re-estimated for more than five hundred iterations, with an evaluation of the results carried out every ten iterations (henceforth, one step); and c) the first experiment was repeated with HMMs initialized using the reference segmentation and increasing the number of mixtures up to 256. The performance was evaluated on the validation set after every step with insertion penalties from zero to minus one hundred in steps of ten. For experiment b), the performance did not improve (it slightly decreased) after 350 iterations.

The results on the test set for all experiments, optimised on the validation set, showed negative accuracy due to the many false positives generated by the system: of the order of three thousand in all cases. Hence, the importance of a post-processing stage for HMMs that estimates the confidence of the predictions and a threshold level for balancing true and false positives.

3.3 Setup

The CTC-BLSTM network was trained with the input sequences and keyword targets from section 3.1. The network has, therefore, 39 input units and 13 units in the output soft-max layer. Both the forward and backward hidden recurrent

layers consist of 128 LSTM memory blocks with one memory cell per block, peephole connections and forget gates. The input and output cell activation functions are a hyperbolic tangent. The gates use a logistic sigmoid function in the range $[0,1]$. The input layer is fully connected to the hidden layer and the hidden layer is fully connected to itself and to the output layer. The total number of weights in the network is 176,141.

Training was carried out with a learning rate of 10^{-4} and a momentum coefficient of 0.9. Weights were initialised with a Gaussian random distribution with mean zero and standard deviation 0.1. In addition, Gaussian noise with a standard deviation of 0.5 was added to the inputs during training to improve generalization. The network was tested every five epochs on the validation set and training finished once the error on the validation set stopped decreasing.

3.4 Results

The typical sequence of outputs for a trained network can be seen in figure 1. It consists in a sequence of spikes associated with detected keywords and separated by long periods during which the non-keyword output unit is the most active. The CTC algorithm allows the network to keep any output active for as long as necessary or as little as one time step only. To evaluate the results, at every time step the output with the highest activation is chosen. If an output is the most active for a period of time, we choose the highest activation during that period as the probability of detecting the keyword, and the time at which the highest activation occurs as the location of the detected keyword.

Spikes whose identity match a keyword in the speech signal and that appear within the boundaries of the keyword in the speech signal are counted as true positives, unless more than one spike is output in that period, in which case only one of them counts as a true positive and the rest count as false positives. Those spikes that appear out of the boundaries of the keyword in the speech signal are considered false positives. The accuracy of the system is given by the number of true positives minus the number of false positives divided by the number of keywords in the data set.

Four networks were trained on the same task with different initial random weights. Average accuracy over four runs was 84.5% with a standard error of 1.2%. The average probability of true positives was 0.98 with a standard error of 0.004. The average probability of false positives was 0.80 with an standard error of 0.01. As expected, given the discriminative nature of the algorithm, setting a posteriori a threshold level in between this probability levels did not increase performance significantly (84.8% accuracy, with an standard error of 1.2%). Table 1 shows the number of true and false positives by keyword for the network with the best performance. As shown in the table, the system discriminates almost perfectly between similar keywords such as “juni” and “juli”.

As of now, the CTC training algorithm does not constrain the spikes to appear within the begin and end times of the speech pattern (it is assumed that a reference segmentation is not available). The algorithm trains the network to detect the correct sequence of keywords and in the right order for any utterance

Table 1. Results by keyword for the ANN with the best performance. Accuracy is the number of hits minus false positives divided by the actual number of keywords in the test set. Average accuracy over four runs is 84.5 % with a standard error of 1.2 %.

Keyword	# Hits	# FPs	# Actual	% Accuracy
april	27	2	32	78.12
august	29	1	34	82.35
donnerstag	55	6	56	87.50
februar	55	1	60	90.00
frankfurt	18	0	25	72.00
freitag	40	4	45	80.00
hannover	76	5	86	82.56
januar	35	4	38	81.58
juli	53	1	56	92.86
juni	63	2	66	92.42
mittwoch	36	1	39	89.74
montag	79	3	83	91.57
Overall	566	30	620	86.45

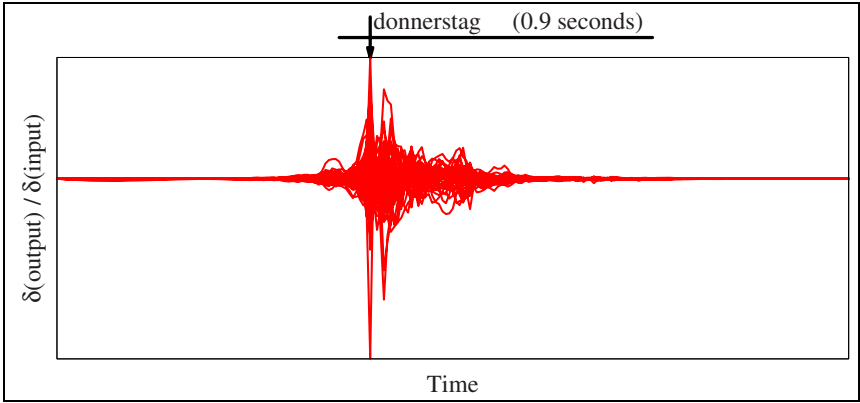


Fig. 2. The figure shows the dependency of the output unit associated with keyword “donnerstag” at the time step when this keyword is detected in figure 1, with respect to the network inputs at various time steps around the detection point (indicated by an arrow at the top of the figure). In addition, the extent (0.9s) and location of the keyword in the speech signal is shown at the top of the figure. As can be seen, the probability of detecting the keyword depends on the inputs over a long time span, and decreases towards the end of the keyword which is the least discriminative part of it: “tag” (day).

in the training set. Probably this makes training slower, but it seems that it is not necessary to add this constraint to the training algorithm in order to achieve good performance: when the results are evaluated without using the segmentation, the average accuracy over four runs is 86.1 % with a standard

error of 0.7 %. Of those keyword candidates appearing outside the boundaries in the speech signal, only one was more than 0.5 s outside the reference segment. This level of precision should be enough for tasks where high detection accuracy is more important.

Finally, figure 2 shows, for a successful detection, the dependency of the network output at the time when the keyword is detected with respect to the inputs at some previous and following time steps. As can be seen, the probability of detecting the keyword depends on the inputs over a long time span (of about one second, or a hundred network time steps) and, for the example shown, the dependency decreases towards the end of the keyword, which is the least discriminative part of it.

4 Discussion and Future Work

The main difference between our approach to discriminative keyword spotting and other discriminative systems is the capability of computing non-local posteriors for the keywords, i.e. the system uses information over long time spans to compute the probability of a keyword appearing in the speech signal. In addition to this, the a posteriori estimation of a threshold level to balance true and false positives is not required.

The algorithm makes very few assumptions about the domain, which facilitates the development of systems for detection tasks. Nonetheless, if very high precision in the location of spots is required, the system might benefit from adding to the algorithm the constraint that keywords must be detected within the boundaries established by the reference segmentation. However, not having this constraint greatly simplifies the preparation of training data.

To make the system scale, one solution consists in implementing smaller networks. For example, one for each keyword, or one for each set of similar keywords among which the system must discriminate. New networks can be added to the system at any time. Under the assumptions that, for different networks, the training data has similar characteristics and that the non-keyword output unit is associated to a model of similar characteristics, which is reasonable, the estimates of the a posteriori probabilities given by different networks can be compared and used to determine the presence or absence of keywords in the input data stream.

Another option to improve scalability consists in detecting sub-word units in the signal and use these outputs as inputs for one of the existing systems that builds keywords from sub-word units. The detection of sub-word units with the system proposed in this paper will benefit from the same features and ease of use described in this paper (see [9] for results on phoneme recognition with CTC-BLSTM). Besides this, it is possible to feed the sub-word level predictions to another CTC-BLSTM network with keyword-level outputs [10].

The choice of sub-word or word models depends on the task. In general, sub-word units are more practical for audio indexing tasks because the system must respond to searches for unknown keywords. However, in tasks such as detection

of command words and spoken password verification the vocabulary is unlikely to change. Also, the benefits of word-level discrimination (e.g. decreased confusion among similar keywords) are most desirable in these cases, where security is paramount.

5 Conclusion

We have presented a word-level discriminative keyword spotting system that is fast, accurate and easy to use. The probability of detecting a keyword is computed using information from long time spans. These probabilities can be read directly from the outputs of a recurrent neural network. An a posteriori estimate of an acceptance threshold level is not required. Finally, the system makes few assumptions about the domain: only the input speech signal and a list of keywords (which may be empty) in the order in which they occur in the signal are necessary to train the system. The algorithm is general and can be used for any task requiring the detection of patterns in sequence data. In a keyword spotting task in a large database of unconstrained speech the system achieved a keyword spotting accuracy of 84.5 %.

Acknowledgments

This research was funded by SNF grant 200021-111968/1.

References

1. Wilpon, J.G., Rabiner, L.R., Lee, C., Goldman, E.R.: Automatic recognition of keywords in unconstrained speech using hidden Markov models. *IEEE Transactions on Acoustics, Speech and Audio Processing* 38(11), 1870–1878 (1990)
2. Benítez, M.C., Rubio, A., García, P., de la Torre, A.: Different confidence measures for word verification in speech recognition. *Speech Communication* 32, 79–94 (2000)
3. Silaghi, M.C., Bourlard, H.: Iterative posterior-based keyword spotting without filler models. In: *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop, Keystone - Colorado (U.S.A.)* (1999)
4. Silaghi, M.: Spotting subsequences matching an HMM using the average observation probability criteria with application to keyword spotting. In: *Proceedings of the 20th National Conference on Artificial Intelligence and the 17th Conference on Innovative Applications of Artificial Intelligence*, pp. 1118–1123. Pittsburgh - Pennsylvania (U.S.A.) (2005)
5. Ketabdar, H., Vepa, J., Bengio, S., Bourlard, H.: Posterior based keyword spotting with a priori thresholds. In: *Proceedings of the 9th International Conference on Spoken Language Processing, Pittsburgh - Pennsylvania (U.S.A.)* (2006)
6. Robinson, A.J.: An application of recurrent nets to phone probability estimation. *IEEE Transactions on Neural Networks* 5(2), 298–305 (1994)
7. Bourlard, H.A., Morgan, N.: *Connectionist speech recognition: a hybrid approach*. Kluwer Academic Publishers, Dordrecht (1994)

8. Keshet, J., Grangier, D., Bengio, S.: Discriminative keyword spotting. In: Workshop on Non-Linear Speech Processing NOLISP, Paris (France) (2007)
9. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural nets. In: Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh - Pennsylvania (U.S.A.) (2006)
10. Fernández, S., Graves, A., Schmidhuber, J.: Sequence labelling in structured domains with hierarchical recurrent neural networks. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad (India) (2007)
11. Silaghi, M.C., Vargiya, R.: A new evaluation criteria for keyword spotting techniques and a new algorithm. In: Proceedings of InterSpeech (2005)
12. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks* 18(5–6), 602–610 (2005)
13. Graves, A., Fernández, S., Schmidhuber, J.: Bidirectional LSTM networks for improved phoneme classification and recognition. In: Duch, W., Kacprzyk, J., Oja, E., Zadrozny, S. (eds.) ICANN 2005. LNCS, vol. 3697, pp. 799–804. Springer, Heidelberg (2005)
14. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Computation* 9(8), 1735–1780 (1997)
15. Gers, F., Schraudolph, N., Schmidhuber, J.: Learning precise timing with LSTM recurrent networks. *Journal of Machine Learning Research* 3, 115–143 (2002)
16. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* 45, 2673–2681 (1997)
17. Verbmobil: Database version 2.3 (2004), <http://www.phonetik.uni-muenchen.de/Forschung/Verbmobil/Verbmobil.html>

Exhibit 8

(56)

References Cited

U.S. PATENT DOCUMENTS

6,973,426 B1 12/2005 Schier
 7,016,833 B2 3/2006 Gable
 7,222,072 B2 * 5/2007 Chang G10L 17/22
 704/249
 7,720,012 B1 5/2010 Borah
 7,904,297 B2 * 3/2011 Mirkovic G06F 17/28
 704/235
 8,099,288 B2 1/2012 Zhang
 8,209,174 B2 6/2012 Al-Telmissani
 8,340,975 B1 * 12/2012 Rosenberger G10L 15/22
 704/270
 8,717,949 B2 5/2014 Crinon
 8,719,018 B2 5/2014 Dinerstein
 8,768,687 B1 * 7/2014 Quasthoff G10L 15/26
 379/142.14
 8,775,191 B1 7/2014 Sharifi et al.
 8,996,372 B1 * 3/2015 Secker-Walker G10L 15/34
 704/231
 9,142,218 B2 * 9/2015 Schroeter G10L 17/20
 2002/0049596 A1 * 4/2002 Burchard G10L 15/26
 704/270
 2002/0072905 A1 * 6/2002 White G10L 15/30
 704/231
 2002/0123890 A1 * 9/2002 Kopp H04L 12/6418
 704/233
 2002/0193991 A1 * 12/2002 Bennett G10L 15/32
 704/247
 2003/0200090 A1 * 10/2003 Kawazoe G10L 15/00
 704/251
 2003/0231746 A1 12/2003 Hunter
 2004/0101112 A1 * 5/2004 Kuo H04M 1/67
 379/88.01
 2005/0165607 A1 * 7/2005 Di Fabbri G06F 17/2785
 704/256
 2006/0074656 A1 * 4/2006 Mathias G10L 15/063
 704/243
 2006/0184370 A1 * 8/2006 Kwak G10L 15/1822
 704/275
 2007/0100620 A1 * 5/2007 Tavares G10L 17/06
 704/246
 2007/0198262 A1 8/2007 Mindlin

2009/0258333 A1 * 10/2009 Yu G09B 5/04
 434/157
 2009/0292541 A1 * 11/2009 Daya G10L 15/063
 704/251
 2010/0070276 A1 * 3/2010 Wasserblat H04M 3/5232
 704/243
 2010/0110834 A1 * 5/2010 Kim G01S 3/8083
 367/124
 2011/0026722 A1 * 2/2011 Jing G10L 21/0208
 381/71.1
 2011/0054892 A1 * 3/2011 Jung G10L 15/08
 704/233
 2011/0060587 A1 * 3/2011 Phillips G10L 15/30
 704/235
 2011/0066429 A1 * 3/2011 Shperling G10L 25/78
 704/228
 2012/0084087 A1 4/2012 Yang
 2012/0232896 A1 * 9/2012 Taleb G10L 25/78
 704/233
 2012/0265528 A1 * 10/2012 Gruber G10L 15/18
 704/235
 2013/0060571 A1 * 3/2013 Soemo G10L 15/30
 704/251
 2013/0132086 A1 * 5/2013 Xu G10L 15/01
 704/257
 2013/0183944 A1 * 7/2013 Mozer H04L 12/282
 455/414.1
 2014/0012573 A1 * 1/2014 Hung G06F 1/3215
 704/233
 2014/0088961 A1 * 3/2014 Woodward G10L 15/22
 704/235
 2014/0222430 A1 * 8/2014 Rao G10L 15/04
 704/254
 2014/0257821 A1 * 9/2014 Adams G10L 25/93
 704/275
 2014/0278383 A1 * 9/2014 Fan G10L 25/84
 704/224
 2014/0278435 A1 * 9/2014 Ganong, III G10L 15/22
 704/275
 2015/0154953 A1 * 6/2015 Bapat G10L 15/06
 704/251
 2015/0262577 A1 * 9/2015 Nomura G10L 15/22
 704/231

* cited by examiner

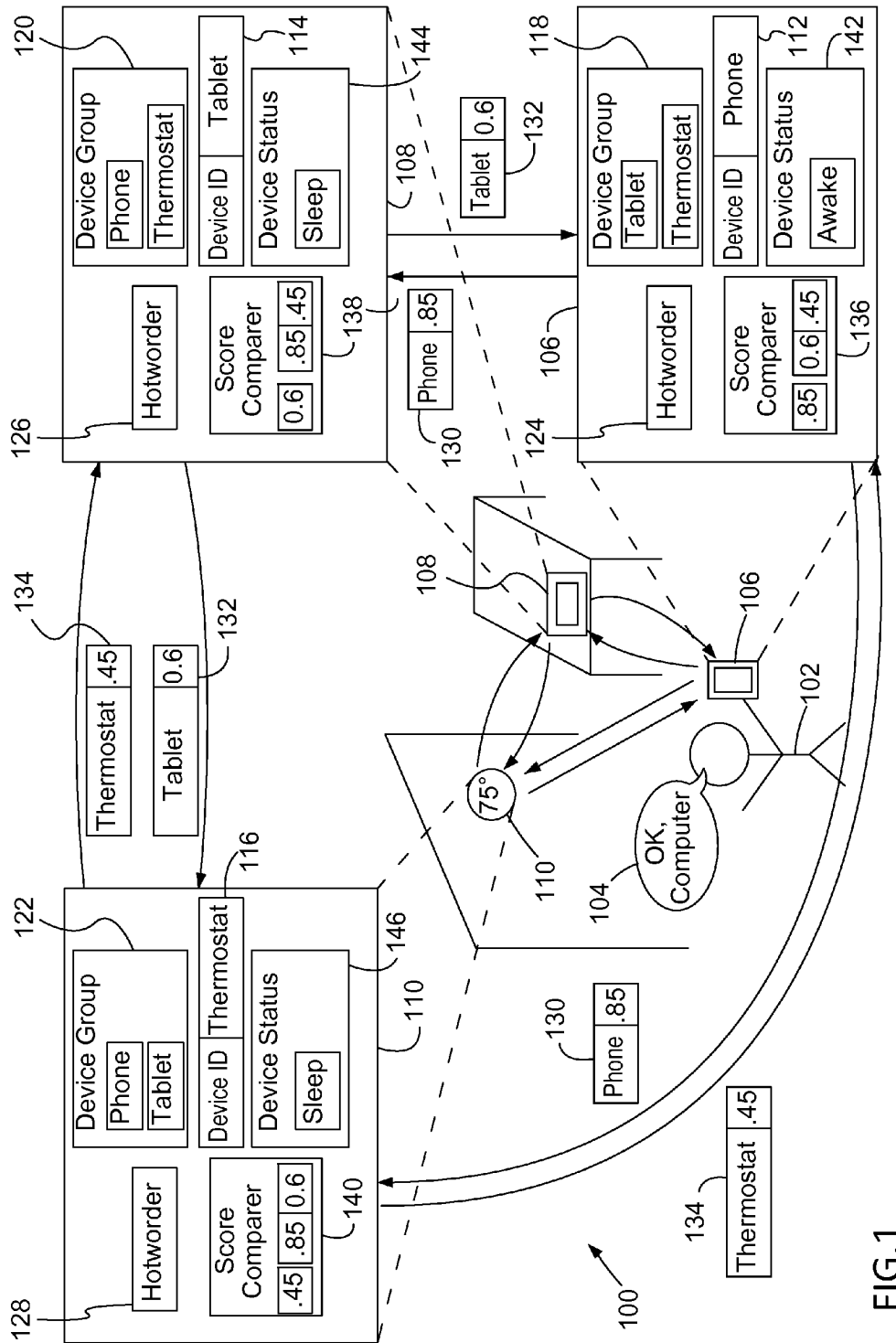


FIG.1

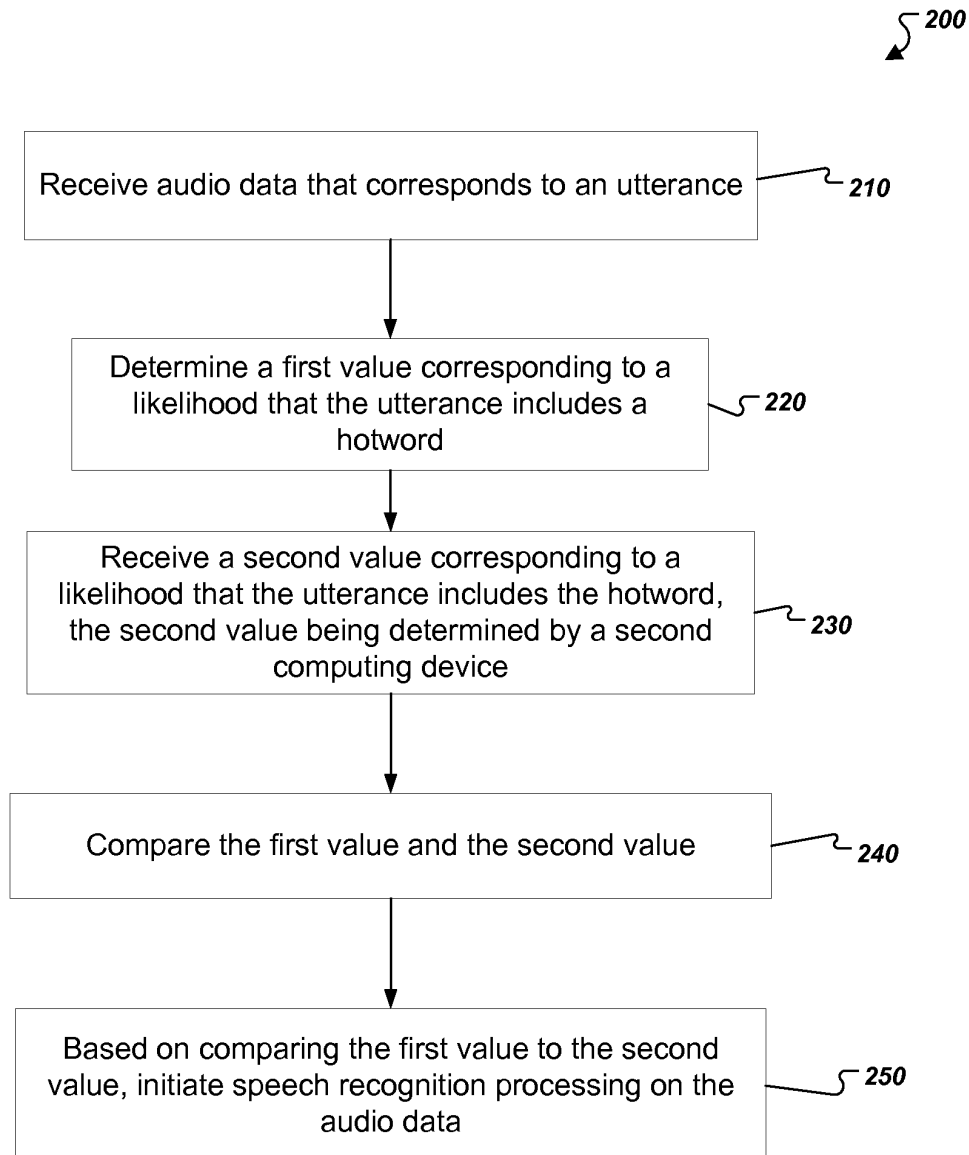


FIG. 2

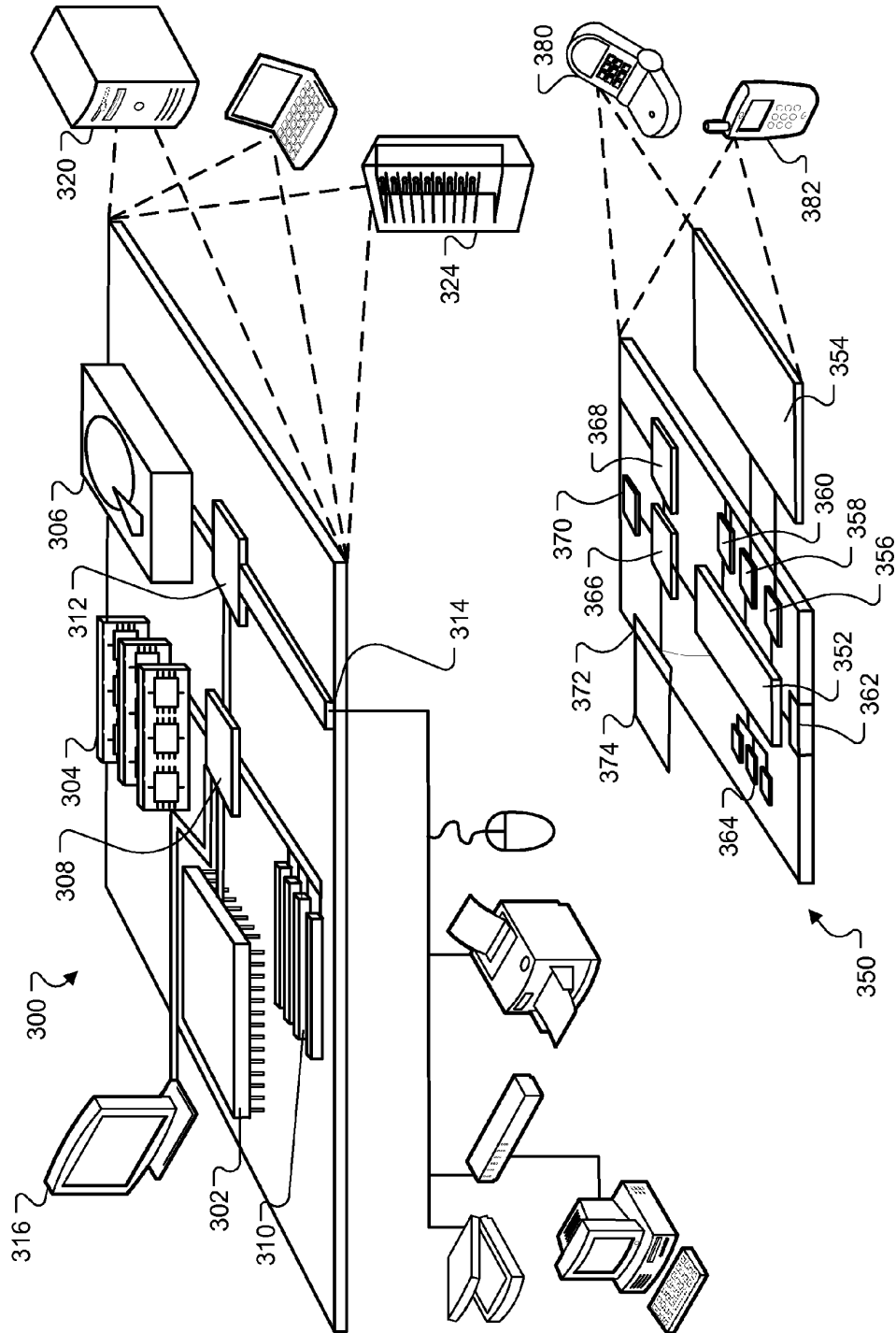


FIG. 3

1

HOTWORD DETECTION ON MULTIPLE DEVICES**CROSS-REFERENCE TO RELATED APPLICATION**

This application claims the benefit of U.S. Provisional Application Ser. No. 62/061,830, filed on Oct. 9, 2014, which is incorporated by reference

TECHNICAL FIELD

This specification generally relates to systems and techniques for recognizing the words that a person is speaking, otherwise referred to as speech recognition.

BACKGROUND

The reality of a speech-enabled home or other environment—that is, one in which a user need only speak a query or command out loud and a computer-based system will field and answer the query and/or cause the command to be performed—is upon us. A speech-enabled environment (e.g., home, workplace, school, etc.) can be implemented using a network of connected microphone devices distributed throughout the various rooms or areas of the environment. Through such a network of microphones, a user has the power to orally query the system from essentially anywhere in the environment without the need to have a computer or other device in front of him/her or even nearby. For example, while cooking in the kitchen, a user might ask the system “how many milliliters in three cups?” and, in response, receive an answer from the system, e.g., in the form of synthesized voice output. Alternatively, a user might ask the system questions such as “when does my nearest gas station close,” or, upon preparing to leave the house, “should I wear a coat today?”

Further, a user may ask a query of the system, and/or issue a command, that relates to the user’s personal information. For example, a user might ask the system “when is my meeting with John?” or command the system “remind me to call John when I get back home.”

SUMMARY

For a speech-enabled system, the users’ manner of interacting with the system is designed to be primarily, if not exclusively, by means of voice input. Consequently, the system, which potentially picks up all utterances made in the surrounding environment including those not directed to the system, must have some way of discerning when any given utterance is directed at the system as opposed, e.g., to being directed at an individual present in the environment. One way to accomplish this is to use a hotword, which by agreement among the users in the environment, is reserved as a predetermined word that is spoken to invoke the attention of the system. In an example environment, the hotword used to invoke the system’s attention are the words “OK computer.” Consequently, each time the words “OK computer” are spoken, it is picked up by a microphone, conveyed to the system, which performs speech recognition techniques to determine whether the hotword was spoken and, if so, awaits an ensuing command or query. Accordingly, utterances directed at the system take the general form [HOTWORD] [QUERY], where “HOTWORD” in this example is “OK computer” and “QUERY” can be any question, command, declaration, or

2

other request that can be speech recognized, parsed and acted on by the system, either alone or in conjunction with the server via the network.

According to one innovative aspect of the subject matter described in this specification, a user device receives an utterance that is spoken by a user. The user device determines whether the utterance includes a hotword and computes a hotword confidence score that indicates a likelihood that the utterance includes the hotword. The user device transmits this score to other user devices in the near vicinity. The other user devices likely received the same utterance. The other user devices compute a hotword confidence score and transmit their scores to the user device. The user device compares the hotword confidence scores. If the user device has the highest hotword confidence score, then the user device remains active and prepares to process additional audio. If the user device does not have the highest hotword confidence score, then the user device does not process the additional audio.

In general, another innovative aspect of the subject matter described in this specification may be embodied in methods that include the actions of receiving, by a first computing device, audio data that corresponds to an utterance; determining a first value corresponding to a likelihood that the utterance includes a hotword; receiving a second value corresponding to a likelihood that the utterance includes the hotword, the second value being determined by a second computing device; comparing the first value and the second value; and based on comparing the first value to the second value, initiating speech recognition processing on the audio data.

These and other embodiments can each optionally include one or more of the following features. The actions further include determining that the first value satisfies a hotword score threshold. The actions further include transmitting the first value to the second computing device. The actions further include determining an activation state of the first computing device based on comparing the first value and the second value. The action of determining an activation state of the first computing device based on comparing the first value and the second value further includes determining that the activation state is an active state. The actions further include receiving, by the first computing device, additional audio data that corresponds to an additional utterance; determining a third value corresponding to a likelihood that the additional utterance includes the hotword; receiving a fourth value corresponding to a likelihood that the utterance includes the hotword, the fourth value being determined by a third computing device; comparing the first value and the second value; and based on comparing the first value and the second value, determining that the activation state of the first computing device is an inactive state.

The action of transmitting the first value to the second computing device further includes transmitting, to a server, through a local network, or through a short range radio, the first value. The action of receiving a second value corresponding to a likelihood that the utterance includes the hotword, the second value being determined by a second computing device further includes receiving, from the server, through the local network, or through the short range radio, a second value that was determined by a second computing device. The actions further include identifying the second computing device; and determining that the second computing device is configured to respond to utterances that include the hotword. The action of transmitting the first value to the second computing device further includes transmitting a first identifier for the first computing device. The action of receiving a second value corresponding to a likelihood that the utterance includes the

3

hotword, the second value being determined by a second computing device further includes receiving a second identifier for the second computing device. The action of determining that the activation state is an active state further includes determining that a particular amount of time has elapsed since receiving the audio data that corresponds to the utterance. The actions further include continuing, for a particular amount of time, to transmit the first value based on determining that the activation state is an active state.

Other embodiments of this aspect include corresponding systems, apparatus, and computer programs recorded on computer storage devices, each configured to perform the operations of the methods.

Particular embodiments of the subject matter described in this specification can be implemented so as to realize one or more of the following advantages. Multiple devices can detect a hotword and only one device will respond to the hotword.

The details of one or more embodiments of the subject matter described in this specification are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages of the subject matter will become apparent from the description, the drawings, and the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a diagram of an example system for hotword detection.

FIG. 2 is a diagram of an example process for hotword detection.

FIG. 3 shows an example of a computing device and a mobile computing device.

Like reference numbers and designations in the various drawings indicate like elements.

DETAILED DESCRIPTION

In the not too distant future, it is possible that many devices may be continuously listening for hotwords. When a single user has multiple devices trained to respond to their voice (e.g., a phone, tablet, TV, etc), it may be desirable to suppress responding to hotwords on devices that are not likely to be the ones a user intends to address. For example, when a user speaks the hotword toward one device, if any of their other devices are nearby, it is likely that they will also trigger a voice search. In many cases, this is not the user's intention. Thus, it may be advantageous if only a single device should trigger, specifically the device the user is speaking to. The present specification addresses the problem of selecting the correct device for reacting to a hotword, and suppressing reaction to the hotword on other devices.

FIG. 1 is a diagram of an example system 100 for hotword detection. In general, system 100 illustrates a user 102 speaking an utterance 104 that is detected by microphones of computing devices 106, 108, and 110. The computing devices 106, 108, and 110 process the utterance 104 to determine a likelihood that the utterance 104 includes a hotword. The computing devices 106, 108, and 110 each transmit data to each other that indicates the likelihood that the utterance 104 includes a hotword. The computing devices 106, 108, and 110 each compare the data, and the computing device that computed the highest likelihood that the utterance 104 included a hotword initiates speech recognition on the utterance 104. The computing devices that did not compute the highest

4

likelihood that the utterance 104 includes a hotword do not initiate speech recognition on the speech following the utterance 104.

Before transmitting, to another computing device, data that indicates a likelihood that the utterance 104 corresponds to a hotword, the computing devices that are located near each other identify each other. In some implementations, the computing devices identify each other by searching the local network for other devices that are configured to respond to the hotword. For example, computing device 106 may search the local area network for other devices that are configured to respond to the hotword and identify computing device 108 and computing device 110.

In some implementations, the computing devices identify other nearby computing devices that are configured to respond to the hotword by identifying the user who is logged into each device. For example, user 102 is logged into computing devices 106, 108, and 110. The user 102 has the computing device 106 is the user's hand. The computing device 108 is sitting on the table, and the computing device 110 is located on a nearby wall. Computing device 106 detects computing devices 108 and 110 and each computing device shares information that is related to the user who is logged into the computing device, such as a user identifier. In some implementations, the computing devices may identify other near computing devices that are configured to respond to the hotword by identifying computing devices that are configured to respond when the hotword is spoken by the same user through speaker identification. For example, the user 102 configured the computing devices 106, 108, and 110 each to respond to the voice of user 102 when user 102 speaks the hotword. The computing devices share the speaker identification information by providing a user identifier for user 102 to each other computing device. In some implementations, the computing devices may identify other computing devices that are configured to respond to the hotword through short range radio. For example, the computing device 106 may transmit a signal through short range radio searching for other computing devices that are configured to respond to the hotword. The computing devices may employ one of these techniques or a combination of them to identify other computing devices that are configured to respond to the hotword.

Once the computing devices 106, 108, and 110 have identified other computing devices that are configured to respond to the hotword, the computing devices 106, 108, and 110 share and store device identifiers for the identified computing devices. The identifiers may be based on a type of device, an IP address of the device, a MAC address, a name given to the device by a user, or any similar unique identifier. For example, the device identifier 112 for computing device 106 may be "phone." The device identifier 114 for computing device 108 may be "tablet." The device identifier 116 for computing device 110 may be "thermostat." The computing devices 106, 108, and 110 store the device identifier for the other computing devices that are configured to respond to the hotword. Each computing device has a device group where the computing device stores the device identifiers. For example, computing device 106 has device group 118 that lists "tablet" and "thermostat" as the two devices that will receive the likelihood that the audio data includes the hotword as computed by the computing device 106. The computing device 108 has device group 120 that lists "phone" and "thermostat" as the two devices that will receive the likelihood that the audio data includes the hotword as computed by the computing device 108. The computing device 110 has device group 122 that lists "phone" and "tablet" as the two devices that will receive

5

the likelihood that the audio data includes the hotword as computed by the computing device 110.

When the user 102 speaks the utterance 104, "OK computer," each computing device that has a microphone in the vicinity of the user 102 detects and processes the utterance 104. Each computing device detects the utterance 104 through an audio input device such as a microphone. Each microphone provides audio data to a respective audio subsystem. The respective audio subsystem buffers, filters, and digitizes the audio data. In some implementations, each computing device may also perform endpointing and speaker identification on the audio data. The audio subsystem provides the processed audio data to a hotworder. The hotworder compares the processed audio data to known hotword data and computes a confidence score that indicates the likelihood that the utterance 104 corresponds to a hotword. The hotworder may extract audio features from the processed audio data such as filterbank energies or mel-frequency cepstral coefficients. The hotworder may use classifying windows to process these audio features such as by using a support vector machine or a neural network. Based on the processing of the audio features, the hotworder 124 computes a confidence score of 0.85, hotworder 126 computes a confidence score of 0.6, and hotworder 128 computes a confidence score of 0.45. In some implementations, the confidence score may be normalized to a scale of zero to one, with a higher number indicating a greater confidence that the utterance 104 includes a hotword.

Each computing device transmits a respective confidence score data packet to the other computing devices in the device group. Each confidence score data packet includes a respective confidence score and the respective device identifier for the computing device. For example, the computing device 106 transmits the confidence score data packet 130 that includes the confidence score of 0.85 and the device identifier "phone" to computing devices in device group 118, computing devices 108 and 110. The computing device 108 transmits the confidence score data packet 132 that includes the confidence score of 0.6 and the device identifier "tablet" to computing devices in device group 120, computing devices 106 and 110. The computing device 110 transmits the confidence score data packet 134 that includes the confidence score of 0.45 and the device identifier "thermostat" to computing devices in device group 118, computing device 106 and 108.

In some implementations, a computing device may transmit the confidence score data packet if the confidence score satisfies a hotword score threshold. For example, if the hotword score threshold is 0.5, then the computing device 110 would not transmit the confidence score data packet 134 to the other computing devices in device group 122. The computing devices 106 and 108 would still transmit the confidence score data packets 130 and 132 to computing devices in device groups 118 and 120, respectively.

In some implementations, the computing device that transmits a confidence score data packet may transmit the confidence score data packet to other computing devices directly. For example, computing device 106 may transmit the confidence score data packet 130 to computing devices 108 and 110 through a short range radio. The communication protocol used between two computing devices may be universal plug and play. In some implementations, a computing device that transmits a confidence score data packet may broadcast the confidence score data packet. In this instance, the confidence score data packet may be received by the computing devices in the device group and by other computing devices. In some implementations, a computing device that transmits a confidence score data packet may transmit the confidence score

6

data packet to a server and then the server transmits the confidence score data packet to the computing devices in the data group. The server may be located within the local area network of the computing devices or accessible through the Internet. For example, the computing device 108 sends the confidence score data packet 132 and the list of computing devices in device group 120 to a server. The server transmits the confidence score data packet 132 to computing device 106 and 110. In instances where a computing device that is transmitting the confidence score data packet to another computing device, the receiving computing device may send back a confirmation that the receiving computing device received the confidence score data packet.

Each computing device uses a score comparer to compare the hotword confidence scores that the computing device has received. For example, the computing device 106 computed a hotword confidence score of 0.85 and received hotword confidence scores of 0.6 and 0.45. In this instance, the score comparer 136 compares the three scores and identifies the score of 0.85 as the highest. For computing devices 108 and 110, the score comparers 138 and 140 reach similar conclusions, identifying the score of 0.85, which corresponds to computing device 106, as the highest.

The computing device that determines that its own hotword confidence score is the highest initiates speech recognition on speech data that follows the hotword utterance. For example, the user may speak "OK computer" and computing device 106 may determine that it has the highest hotword confidence score. The computing device 106 will initiate speech recognition on audio data received after the hotword. If the user speaks "call Alice," then the computing device 106 will process the utterance and execute the appropriate command. In some implementations, receiving a hotword may cause the computing devices that receive the hotword to activate from a sleep state. In this instance, the computing device with the highest hotword confidence score remains in an awake state while the other computing devices that do not have the highest hotword confidence score do not process speech data that follows the hotword utterance and enter a sleep state.

As illustrated in FIG. 1, the score comparer 136 identified the hotword confidence score corresponding to computing device 106 to be the highest. Therefore, the device status 142 is "awake." The score comparers 138 and 140 also identified the hotword confidence score corresponding to computing device 106 to be the highest. Therefore, the device statuses 138 and 140 are "asleep." In some implementations, the activation state of the computing device may be unaffected. For example, the user 102 may be watching a movie on the computing device 108 and have the computing device 106 in the user's hand. When the user 102 speaks "OK computer," the computing device 106, by virtue of having the highest hotword confidence score, initiates speech recognition on the audio data following the hotword. The computing device 108 does not initiate speech recognition on the audio data following the hotword, and continues to play the movie.

In some implementations, the computing device that determines that it has the highest hotword confidence score waits for a particular amount of time before beginning to perform speech recognition on speech following the hotword. Doing so allows a computing device that computed the highest hotword confidence score to begin performing speech recognition on speech that follows the hotword without waiting for a higher hotword confidence score. To illustrate, the score comparer 136 of computing device 106 received hotword confidence scores of 0.6 and 0.45 from computing device 108 and 110, respectively, as well as the hotword confidence score of 0.85 from the hotworder 124. From the time that the hot-

worder 124 computes a hotword confidence score of the “Ok computer” audio data, the computing device 106 waits five hundred milliseconds before performing speech recognition on speech that follows the hotword. In instances where the score comparer receives a higher score, the computing device may not wait for a particular amount of time before setting the device status to “sleep.” For example, the hotworder 126 of computing device 108 computes a hotword confidence score of 0.6 and receives hotword confidence scores of 0.85 and 0.45. Once the computing device 108 receives the hotword confidence score of 0.85, then the computing device 108 can set the device status 144 to “sleep.” This assumes that the computing device 108 receives the hotword confidence score of 0.85 within the particular amount of time after the hotworder 126 computes the hotword confidence score of 0.6.

In some implementations, when a computing device has the highest hotword confidence score, the computing device may continue to broadcast the confidence score data packet for a particular amount of time to ensure that other computing devices receive the confidence score data packet. This strategy would be most applicable in instances where a computing device does send back a confirmation once it receives a confidence score data packet from another computing device. Therefore, if the computing device 106 transmits the confidence score data packet 130 to computing devices in data group 118 and receives a confirmation before a particular amount of time such as five hundred milliseconds, then the computing device 106 may begin to perform speech recognition on speech following the hotword. In instances where computing devices broadcast their confidence score data packets and do not expect confirmation, the computing device may continue to broadcast their hotword confidence scores for a particular amount of time, such as five hundred milliseconds, or until the computing device receives a higher hotword confidence score, whichever comes first. For example, computing device 110 computes a hotword confidence score of 0.45 and begins to broadcast the confidence score data packet 134. After three hundred milliseconds, the computing device 110 receives confidence score data packet 130 and stops broadcasting the confidence score data packet 134 because the hotword confidence score of 0.85 from confidence score data packet 130 is higher than the hotword confidence score of forty five. As another broadcast example, computing device 106 computes a hotword confidence score of 0.45 and begins to broadcast the confidence score data packet 130. After five hundred milliseconds, the computing device 106 stops broadcasting confidence score data packet 130 and begins to perform speech recognition on speech following the hotword. The computing device 106 may receive the confidence score data packets 132 and 134 before five hundred milliseconds has elapsed, but because the hotword confidence scores in the confidence score data packets 132 and 134 are lower than 0.85, the computing device continues to wait until after the five hundred milliseconds has elapsed.

In some implementations, the computing device may begin to perform speech recognition on speech following the hotword until the computing device receives a higher hotword confidence score. The hotworder computes a hotword confidence score and if the hotword confidence score satisfies a threshold, then the computing device performs speech recognition on speech following the hotword. The computing device may perform the speech recognition without displaying any indication of the speech recognition to the user. This may be desirable because doing so gives the user the impression that the computing device is not active while also allowing the computing device to display results based on the

speech recognition to the user quicker than if the computing device had waited until the computing device confirmed that it computed the highest hotword score. As an example, the computing device 106 computes a hotword confidence score of 0.85 and begins to perform speech recognition on speech following the hotword. The computing device 106 receives confidence score data packets 132 and 134 and determines that the hotword confidence score of 0.85 is the highest. The computing device 106 continues to perform speech recognition on speech following the hotword and presents the results to the user. For computing device 108, the hotworder 126 computes a hotword confidence score of 0.6 and the computing device 108 begins to perform speech recognition on speech following the hotword without displaying data to the user. Once the computing device 108 receives the confidence score data packet 130 that includes the hotword confidence of 0.85, the computing device stops performing speech recognition. No data is displayed to the user, and the user is likely under the impression that the computing device 108 has remained in a “sleep” state.

In some implementations, to avoid any latency after a hotword is spoken, scores could be reported from the hotworder before the end of the hotword, e.g. for a partial hotword. For example, as a user is speaking “Ok computer,” a computing device could compute a partial hotword confidence score once the user has finished speaking “OK comp.” The computing device can then share the partial hotword confidence score with other computing devices. The computing device with the highest partial hotword confidence score can continue to process the user’s speech.

In some implementations, a computing device may emit an audible or inaudible sound, e.g., of a particular frequency or frequency pattern, when the computing device determines that a hotword confidence score satisfies a threshold. The sound would signal to other computing devices that the computing device will continue to process the audio data following the hotword. Other computing devices would receive this sound and cease processing of the audio data. For example, a user speaks “Ok computer.” One of the computing devices computes a hotword confidence score that is greater than or equal to a threshold. Once the computing device determines that the hotword confidence score is greater than or equal to a threshold, the computing device emits a sound of eighteen kilohertz. The other computing devices in the vicinity of the user may also be computing a hotword confidence score and may be in the middle of computing a hotword confidence score when the other computing devices receive the sound. When the other computing devices receive the sound, the other computing devices cease processing of the user’s speech. In some implementations, the computing device may encode the hotword confidence score in the audible or inaudible sound. For example, if the hotword confidence score is 0.5, then the computing device may generate an audible or inaudible sound that includes a frequency pattern that encodes the score of 0.5.

In some implementations, the computing devices may use different audio metrics to select a computing device to continue processing the user’s speech. For example, the computing devices may use loudness to determine which computing device will continue to process the user’s speech. The computing device that detects the loudest speech may continue to process the user’s speech. As another example, the computing device that is currently in use or has an active display may notify the other computing devices that it will continue to process the user’s speech upon detecting a hotword.

In some implementations, each computing device that is in the vicinity of the user while the user is speaking receives the

audio data and sends the audio data to a server to improve speech recognition. Each computing device can receive the audio data that corresponds to the user's speech. While only one computing device will appear to the user to be processing the user's speech, each computing device can transmit the audio data to a server. The server can then use the audio data that is received from each computing device to improve speech recognition because the server can compare different audio samples that correspond to the same utterance. For example, a user says "Ok computer, remind me to buy milk." Once the user finishes speaking "Ok computer," the nearby computing devices will have likely determined which computing device has the highest hotword confidence score and that computing device will process and respond "remind me to buy milk" as the user speaks those words. The other computing devices will also receive "remind me to buy milk." While the other computing device will not respond to the "remind me to buy milk" utterance, the other computing devices can send audio data corresponding to "remind me to buy milk" to a server. The computing devices responding to "remind me to buy milk" can also send its audio data to the server. The server can process the audio data to improve speech recognition because the server has different audio samples from different computing devices that correspond to the same "remind me to buy milk" utterance.

FIG. 2 is a diagram of an example process 200 for hotword detection. The process 200 may be performed by a computing device such as the computing device 108 from FIG. 1. The process 200 computes a value that corresponds a likelihood that an utterance includes a hotword and compares the value to other values computed by other computing devices to determine whether or not to perform speech recognition on the portion of the utterance after the hotword.

The computing device receives audio data that corresponds to an utterance (210). A user speaks the utterance and a microphone of the computing device receives the audio data of the utterance. The computing device processes the audio data by buffering, filtering, endpointing, and digitizing the audio data. As an example, the user may utter "Ok, computer" and the microphone of the computing device will receive the audio data that corresponds to "Ok, computer." An audio subsystem of the computing device will sample, buffer, filter, and endpoint the audio data for further processing by the computing device.

The computing device determines a first value corresponding to a likelihood that the utterance includes a hotword (220). The computing device determines the first value, which may be referred to as a hotword confidence score, by comparing the audio data of the utterance to a group of audio samples that include the hotword or by analyzing the audio characteristics of the audio data of the utterance. The first value may be normalized to a scale from zero to one where one indicates the highest likelihood that the utterance includes a hotword. In some implementations, the computing device identifies a second computing device and determines that the second computing device is configured to respond to utterances that include the hotword and is configured by the user to be responsive to the hotword. The user may be logged into both the computing device and the second computing device. Both the computing device and the second computing device may be configured to respond to the user's voice. The computing device and the second computing device may be connected to the same local area network. The computing device and the second computing device may both be located within a particular distance of each other, such as ten meters, as determined by GPS or signal strength. For example, the computing devices may communicate by a short range radio. The com-

puting device may detect a strength of a signal being transmitted by the second computing device as five dBm and translate that a corresponding distance such as five meters.

The computing device receives a second value corresponding to a likelihood that the utterance includes the hotword, the second value being determined by a second computing device (230). The second computing device receives the utterance through a microphone of the second computing device. The second computing device processes the received audio data that corresponds to the utterance and determines a second value or a second hotword confidence score. The second hotword confidence score reflect the likelihood, as calculated by the second computing device, that the utterance includes a hotword. In some implementations, the computing device transmits the first value to the second computing device using one or more of the following techniques. The computing device may transmit the first value to the second computing device through a server accessible through the Internet, through a server that is located on the local area network, or directly through the local area network or a short range radio. The computing device may transmit the first value only to the second computing device or the computing device may broadcast the first value so that other computing devices may also receive the first value. The computing device may receive the second value from the second computing device using the same or different technique as the computing device transmitted the first value.

In some implementations, the computing device may compute a loudness score for the utterance or a signal to noise ratio for the utterance. The computing device may combine the loudness score, the signal to noise ratio, and the hotword confidence score to determine a new value for comparing to similar values from other computing devices. For example, the computing device may compute a hotword confidence score and a signal to noise ratio. The computing device may then combine those two scores and compare to similarly computed scores from other computing devices. In some implementations, the computing device may compute different scores and transmit each score to other computing devices for comparison. For example, the computing device may compute a loudness score for the utterance and a hotword confidence score. The computing device may then transmit those scores to other computing devices for comparison.

In some implementations, the computing device may transmit a first identifier with the first value. The identifier may be based on one or more of an address of the computing device, a name of the computing device given by the user, or a location of the computing device. For example, an identifier may be "69.123.132.43" or "phone." Similarly, the second computing device may transmit a second identifier with the second value. In some implementations, the computing device may transmit the first identifier to particular computing devices that the computing device previously identified as configured to respond to the hotword. For example, the computing device may have previously identified the second computing device as configured to respond to the hotword because, in addition to being able to respond to a hotword, the same user was logged into the second computing device as the computing device.

The computing device compares the first value and the second value (240). The computing device then initiates, based on the result of the comparison, speech recognition processing on the audio data (250). In some implementations, for example, the computing device initiates speech recognition when the first value is greater than or equal to the second value. If the user spoke "ok computer, call Carol," then the computing device would begin to process "call Carol" by

11

performing speech recognition on “call Carol” because the first value is greater than or equal to second value. In some implementations, the computing device sets an activation state. In instances where the first value is greater than or equal to the second value, then the computing device sets the activation state as active or “awake.” In the “awake” state, the computing device displays results from the speech recognition.

In some implementations, the computing device compares the first value and the second value and determines that the first value is less than the second value. The computing device, based on determining that the first value is less than the second value, sets the activation state as inactive or “sleep.” In the “sleep” state, the computing device does not appear, to the user, to be active or processing the audio data.

In some implementations, when the computing device determines that the first value is greater than or equal to the second value, the computing device may wait for a particular amount of time before setting the activation state to active. The computing device may wait for the particular amount of time to increase the probability that the computing device will not receive a higher value from another computing device. The particular amount of time may be fixed or may vary depending on the technique that the computing devices transmit and receive values. In some implementations, when the computing device determines that the first value is greater than or equal to the second value, the computing device may continue to transmit the first value for a particular amount of time. By continuing to transmit the first value for a particular amount of time, the computing device increases the probability that the first value is received by the other computing devices. In instances where the computing device determines that the first value is less than the second value, the computing device may stop transmitting the first value.

In some implementations, the computing device may consider additional information in determining whether to execute the command following the hotword. One example of the additional information may be the portion of the utterance that follows the hotword. Typically, the audio data that follows the hotword corresponds to a command for the computing device such as “call Sally,” “play Halloween Movie,” or “set heat to 70 degrees.” The computing device may identify a typical device that handles the type of request or that is capable of handling the request. A request to call a person would be typically handled by a phone based on pre-programmed typical usages or based on usage patterns of a user of the device. If the user typically watches movies on a tablet, then the tablet may handle a request to play a movie. If the thermostat is capable of adjusting the temperature, then the thermostat may handle temperature adjustments.

For the computing device to consider the portion of the utterance that follows the hotword, the computing device would have to initiate speech recognition on the audio data once it likely identifies a hotword. The computing device may categorize the command portion of the utterance and compute a frequency of commands in that category. The computing device may transmit the frequency along with the hotword confidence score to other computing devices. Each computing device may use the frequencies and the hotword confidence scores to determine whether to execute the command following the hotword.

For example, if the user utters “OK computer, play Michael Jackson,” then if the computing device is a phone that the user use’s twenty percent of the time to listen to music, then the computing device may transmit that information along with the hotword confidence score. A computing device such as a tablet that the user uses five percent of the time to listen to

12

music may transmit that information along with the hotword confidence score to other computing devices. The computing devices may use a combination of the hotword confidence score and the percentage of time playing music to determine whether to execute the command.

FIG. 3 shows an example of a computing device 300 and a mobile computing device 350 that can be used to implement the techniques described here. The computing device 300 is intended to represent various forms of digital computers, such as laptops, desktops, workstations, personal digital assistants, servers, blade servers, mainframes, and other appropriate computers. The mobile computing device 350 is intended to represent various forms of mobile devices, such as personal digital assistants, cellular telephones, smart-phones, and other similar computing devices. The components shown here, their connections and relationships, and their functions, are meant to be examples only, and are not meant to be limiting.

The computing device 300 includes a processor 302, a memory 304, a storage device 306, a high-speed interface 308 connecting to the memory 304 and multiple high-speed expansion ports 310, and a low-speed interface 312 connecting to a low-speed expansion port 314 and the storage device 306. Each of the processor 302, the memory 304, the storage device 306, the high-speed interface 308, the high-speed expansion ports 310, and the low-speed interface 312, are interconnected using various busses, and may be mounted on a common motherboard or in other manners as appropriate. The processor 302 can process instructions for execution within the computing device 300, including instructions stored in the memory 304 or on the storage device 306 to display graphical information for a GUI on an external input/output device, such as a display 316 coupled to the high-speed interface 308. In other implementations, multiple processors and/or multiple buses may be used, as appropriate, along with multiple memories and types of memory. Also, multiple computing devices may be connected, with each device providing portions of the necessary operations (e.g., as a server bank, a group of blade servers, or a multi-processor system).

The memory 304 stores information within the computing device 300. In some implementations, the memory 304 is a volatile memory unit or units. In some implementations, the memory 304 is a non-volatile memory unit or units. The memory 304 may also be another form of computer-readable medium, such as a magnetic or optical disk.

The storage device 306 is capable of providing mass storage for the computing device 300. In some implementations, the storage device 306 may be or contain a computer-readable medium, such as a floppy disk device, a hard disk device, an optical disk device, or a tape device, a flash memory or other similar solid state memory device, or an array of devices, including devices in a storage area network or other configurations. Instructions can be stored in an information carrier. The instructions, when executed by one or more processing devices (for example, processor 302), perform one or more methods, such as those described above. The instructions can also be stored by one or more storage devices such as computer- or machine-readable mediums (for example, the memory 304, the storage device 306, or memory on the processor 302).

The high-speed interface 308 manages bandwidth-intensive operations for the computing device 300, while the low-speed interface 312 manages lower bandwidth-intensive operations. Such allocation of functions is an example only. In some implementations, the high-speed interface 308 is coupled to the memory 304, the display 316 (e.g., through a graphics processor or accelerator), and to the high-speed

13

expansion ports **310**, which may accept various expansion cards (not shown). In the implementation, the low-speed interface **312** is coupled to the storage device **306** and the low-speed expansion port **314**. The low-speed expansion port **314**, which may include various communication ports (e.g., USB, Bluetooth, Ethernet, wireless Ethernet) may be coupled to one or more input/output devices, such as a keyboard, a pointing device, a scanner, or a networking device such as a switch or router, e.g., through a network adapter.

The computing device **300** may be implemented in a number of different forms, as shown in the figure. For example, it may be implemented as a standard server **320**, or multiple times in a group of such servers. In addition, it may be implemented in a personal computer such as a laptop computer **322**. It may also be implemented as part of a rack server system **324**. Alternatively, components from the computing device **300** may be combined with other components in a mobile device (not shown), such as a mobile computing device **350**. Each of such devices may contain one or more of the computing device **300** and the mobile computing device **350**, and an entire system may be made up of multiple computing devices communicating with each other.

The mobile computing device **350** includes a processor **352**, a memory **364**, an input/output device such as a display **354**, a communication interface **366**, and a transceiver **368**, among other components. The mobile computing device **350** may also be provided with a storage device, such as a micro-drive or other device, to provide additional storage. Each of the processor **352**, the memory **364**, the display **354**, the communication interface **366**, and the transceiver **368**, are interconnected using various buses, and several of the components may be mounted on a common motherboard or in other manners as appropriate.

The processor **352** can execute instructions within the mobile computing device **350**, including instructions stored in the memory **364**. The processor **352** may be implemented as a chipset of chips that include separate and multiple analog and digital processors. The processor **352** may provide, for example, for coordination of the other components of the mobile computing device **350**, such as control of user interfaces, applications run by the mobile computing device **350**, and wireless communication by the mobile computing device **350**.

The processor **352** may communicate with a user through a control interface **358** and a display interface **356** coupled to the display **354**. The display **354** may be, for example, a TFT (Thin-Film-Transistor Liquid Crystal Display) display or an OLED (Organic Light Emitting Diode) display, or other appropriate display technology. The display interface **356** may comprise appropriate circuitry for driving the display **354** to present graphical and other information to a user. The control interface **358** may receive commands from a user and convert them for submission to the processor **352**. In addition, an external interface **362** may provide communication with the processor **352**, so as to enable near area communication of the mobile computing device **350** with other devices. The external interface **362** may provide, for example, for wired communication in some implementations, or for wireless communication in other implementations, and multiple interfaces may also be used.

The memory **364** stores information within the mobile computing device **350**. The memory **364** can be implemented as one or more of a computer-readable medium or media, a volatile memory unit or units, or a non-volatile memory unit or units. An expansion memory **374** may also be provided and connected to the mobile computing device **350** through an expansion interface **372**, which may include, for example, a

14

SIMM (Single In Line Memory Module) card interface. The expansion memory **374** may provide extra storage space for the mobile computing device **350**, or may also store applications or other information for the mobile computing device **350**. Specifically, the expansion memory **374** may include instructions to carry out or supplement the processes described above, and may include secure information also. Thus, for example, the expansion memory **374** may be provided as a security module for the mobile computing device **350**, and may be programmed with instructions that permit secure use of the mobile computing device **350**. In addition, secure applications may be provided via the SIMM cards, along with additional information, such as placing identifying information on the SIMM card in a non-hackable manner.

The memory may include, for example, flash memory and/or NVRAM memory (non-volatile random access memory), as discussed below. In some implementations, instructions are stored in an information carrier, that the instructions, when executed by one or more processing devices (for example, processor **352**), perform one or more methods, such as those described above. The instructions can also be stored by one or more storage devices, such as one or more computer- or machine-readable mediums (for example, the memory **364**, the expansion memory **374**, or memory on the processor **352**). In some implementations, the instructions can be received in a propagated signal, for example, over the transceiver **368** or the external interface **362**.

The mobile computing device **350** may communicate wirelessly through the communication interface **366**, which may include digital signal processing circuitry where necessary. The communication interface **366** may provide for communications under various modes or protocols, such as GSM voice calls (Global System for Mobile communications), SMS (Short Message Service), EMS (Enhanced Messaging Service), or MMS messaging (Multimedia Messaging Service), CDMA (code division multiple access), TDMA (time division multiple access), PDC (Personal Digital Cellular), WCDMA (Wideband Code Division Multiple Access), CDMA2000, or GPRS (General Packet Radio Service), among others. Such communication may occur, for example, through the transceiver **368** using a radio-frequency. In addition, short-range communication may occur, such as using a Bluetooth, WiFi, or other such transceiver (not shown). In addition, a GPS (Global Positioning System) receiver module **370** may provide additional navigation- and location-related wireless data to the mobile computing device **350**, which may be used as appropriate by applications running on the mobile computing device **350**.

The mobile computing device **350** may also communicate audibly using an audio codec **360**, which may receive spoken information from a user and convert it to usable digital information. The audio codec **360** may likewise generate audible sound for a user, such as through a speaker, e.g., in a handset of the mobile computing device **350**. Such sound may include sound from voice telephone calls, may include recorded sound (e.g., voice messages, music files, etc.) and may also include sound generated by applications operating on the mobile computing device **350**.

The mobile computing device **350** may be implemented in a number of different forms, as shown in the figure. For example, it may be implemented as a cellular telephone **380**. It may also be implemented as part of a smart-phone **382**, personal digital assistant, or other similar mobile device.

Various implementations of the systems and techniques described here can be realized in digital electronic circuitry, integrated circuitry, specially designed ASICs (application specific integrated circuits), computer hardware, firmware,

15

software, and/or combinations thereof. These various implementations can include implementation in one or more computer programs that are executable and/or interpretable on a programmable system including at least one programmable processor, which may be special or general purpose, coupled to receive data and instructions from, and to transmit data and instructions to, a storage system, at least one input device, and at least one output device.

These computer programs (also known as programs, software, software applications or code) include machine instructions for a programmable processor, and can be implemented in a high-level procedural and/or object-oriented programming language, and/or in assembly/machine language. As used herein, the terms machine-readable medium and computer-readable medium refer to any computer program product, apparatus and/or device (e.g., magnetic discs, optical disks, memory, Programmable Logic Devices (PLDs)) used to provide machine instructions and/or data to a programmable processor, including a machine-readable medium that receives machine instructions as a machine-readable signal. The term machine-readable signal refers to any signal used to provide machine instructions and/or data to a programmable processor.

To provide for interaction with a user, the systems and techniques described here can be implemented on a computer having a display device (e.g., a CRT (cathode ray tube) or LCD (liquid crystal display) monitor) for displaying information to the user and a keyboard and a pointing device (e.g., a mouse or a trackball) by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback (e.g., visual feedback, auditory feedback, or tactile feedback); and input from the user can be received in any form, including acoustic, speech, or tactile input.

The systems and techniques described here can be implemented in a computing system that includes a back end component (e.g., as a data server), or that includes a middleware component (e.g., an application server), or that includes a front end component (e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the systems and techniques described here), or any combination of such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data communication (e.g., a communication network). Examples of communication networks include a local area network (LAN), a wide area network (WAN), and the Internet.

The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

Although a few implementations have been described in detail above, other modifications are possible. For example, while a client application is described as accessing the delegate(s), in other implementations the delegate(s) may be employed by other applications implemented by one or more processors, such as an application executing on one or more servers. In addition, the logic flows depicted in the figures do not require the particular order shown, or sequential order, to achieve desirable results. In addition, other actions may be provided, or actions may be eliminated, from the described flows, and other components may be added to, or removed

16

from, the described systems. Accordingly, other implementations are within the scope of the following claims.

What is claimed is:

1. A computer-implemented method comprising:

receiving, by a first computing device, audio data that corresponds to an utterance;

before beginning automated speech recognition processing on the audio data, processing the audio data using a classifier that classifies audio data as including a particular hotword or as not including the particular hotword;

determining, based on the processing of the audio data using the classifier that classifies audio data as including a particular hotword or as not including the particular hotword, a first value that reflects a first likelihood that the utterance includes the particular hotword;

receiving a second value that reflects a second likelihood that the utterance includes the particular hotword, as determined by a second computing device;

comparing the first value that reflects the first likelihood that the utterance includes the particular hotword and the second value that reflects the second likelihood that the utterance includes the particular hotword; and

based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword, determining whether to begin performing automated speech recognition processing on the audio data.

2. The method of claim 1, comprising:

determining that the first value satisfies a hotword score; and

based on determining that the first value satisfies the hotword score threshold, transmitting, to the second device, the first value that reflects the first likelihood that the utterance includes the particular hotword.

3. The method of claim 1, comprising:

determining an activation state of the first computing device based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword.

4. The method of claim 3, wherein determining an activation state of the first computing device based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword comprises:

determining that the activation state of the first computing device is an active state.

5. The method of claim 1, comprising:

receiving, by the first computing device, additional audio data that corresponds to an additional utterance;

before beginning automated speech recognition processing on the audio data, processing the audio data using a classifier that classifies audio data as including a particular hotword or as not including the particular hotword;

determining, based on the processing of the audio data using the classifier that classifies audio data as including a particular hotword or as not including the particular hotword, a third value that reflects a third likelihood that the additional utterance includes the particular hotword;

receiving a fourth value that reflects a fourth likelihood that the additional utterance includes the particular hotword, as determined by a third computing device;

comparing the third value that reflects the third likelihood that the additional utterance includes the particular hot-

17

word and the fourth value that reflects the fourth likelihood that the additional utterance includes the particular hotword; and

based on comparing the third value that reflects the third likelihood that the additional utterance includes the particular hotword and the fourth value that reflects the fourth likelihood that the additional utterance includes the particular hotword, determining whether to begin performing automated speech recognition processing on the additional audio data.

6. The method of claim 1, wherein:

receiving a second value that reflects a second likelihood that the utterance includes the particular hotword, as determined by a second computing device comprises:

receiving, from a server, through a local network, or through a short range radio communication channel, a second value that reflects the second likelihood that the utterance includes the particular hotword.

7. The method of claim 1, comprising:

determining that the second computing device is configured to respond to utterances that include the particular hotword,

wherein comparing the first value that reflects the first likelihood that the utterance includes the particular hotword and the second value that reflects the second likelihood that the utterance includes the particular hotword is performed in response to determining that the second computing device is configured to respond to utterances that include the particular hotword.

8. The method of claim 1, wherein:

receiving a second value that reflects a second likelihood that the utterance includes the particular hotword, as determined by a second computing device comprises:

receiving a second identifier of the second computing device.

9. The method of claim 4, wherein determining whether to begin performing automated speech recognition processing on the audio data is further based on determining that a particular amount of time has elapsed since receiving the audio data that corresponds to the utterance.

10. The method of claim 4, comprising:

transmitting, for a particular amount of time, the first value that reflects the first likelihood that the utterance includes the particular hotword based on determining that the activation state is an active state.

11. The method of claim 1, comprising:

based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword, determining that the first value that reflects the first likelihood that the utterance includes the particular hotword is greater than the second value that reflects the second likelihood that the utterance includes the particular hotword,

wherein determining whether to perform automated speech recognition processing on the audio data comprises:

based on determining that the first value that reflects the first likelihood that the utterance includes the particular hotword is greater than the second value that reflects the second likelihood that the utterance includes the particular hotword, determining to begin performing automated speech recognition processing on the audio data.

18

12. The method of claim 1, comprising:

based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword, determining that the first value that reflects the first likelihood that the utterance includes the particular hotword is less than the second value that reflects the second likelihood that the utterance includes the particular hotword,

wherein determining whether to perform automated speech recognition processing on the audio data comprises:

based on determining that the first value that reflects the first likelihood that the utterance includes the particular hotword is less than the second value that reflects the second likelihood that the utterance includes the particular hotword, determining not to begin performing automated speech recognition processing on the audio data.

13. The method of claim 1, wherein processing the audio data using a classifier that classifies audio data as including a particular hotword or as not including the particular hotword comprises:

extracting filterbank energies or mel-frequency cepstral coefficients from the audio data.

14. The method of claim 1, wherein processing the audio data using a classifier that classifies audio data as including a particular hotword or as not including the particular hotword comprises:

processing the audio data using a support vector machine or a neural network.

15. A computing device comprising:

one or more storage devices storing instructions that are operable, when executed by the computing device, to cause the computing device to perform operations comprising:

receiving, by a first computing device, audio data that corresponds to an utterance;

before beginning automated speech recognition processing on the audio data, processing the audio data using a classifier that classifies audio data as including a particular hotword or as not including the particular hotword;

determining, based on the processing of the audio data using the classifier that classifies audio data as including a particular hotword or as not including the particular hotword, a first value that reflects a first likelihood that the utterance includes the particular hotword;

receiving a second value that reflects a second likelihood that the utterance includes the particular hotword, as determined by a second computing device;

comparing the first value that reflects the first likelihood that the utterance includes the particular hotword and the second value that reflects the second likelihood that the utterance includes the particular hotword; and

based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword, determining whether to begin performing automated speech recognition processing on the audio data.

19

16. The device of claim 15, wherein the operations further comprise:

determining that the first value satisfies a hotword score; and

based on determining that the first value satisfies the hotword score threshold, transmitting, to the second device, the first value that reflects the first likelihood that the utterance includes the particular hotword.

17. The device of claim 15, wherein the operations further comprise:

determining an activation state of the first computing device based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword.

18. The device of claim 17, wherein determining an activation state of the first computing device based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword comprises:

determining that the activation state of the first computing device is an active state.

19. The device of claim 15, wherein the operations further comprise:

receiving, by the first computing device, additional audio data that corresponds to an additional utterance;

before beginning automated speech recognition processing on the audio data, processing the audio data using a classifier that classifies audio data as including a particular hotword or as not including the particular hotword;

determining, based on the processing of the audio data using the classifier that classifies audio data as including a particular hotword or as not including the particular hotword, a third value that reflects a third likelihood that the additional utterance includes the particular hotword;

receiving a fourth value that reflects a fourth likelihood that the additional utterance includes the particular hotword, as determined by a third computing device;

comparing the third value that reflects the third likelihood that the additional utterance includes the particular hotword and the fourth value that reflects the fourth likelihood that the additional utterance includes the particular hotword; and

based on comparing the third value that reflects the third likelihood that the additional utterance includes the particular hotword and the fourth value that reflects the fourth likelihood that the additional utterance includes the particular hotword, determining whether to begin performing automated speech recognition processing on the additional audio data.

20

20. The device of claim 15, wherein:

receiving a second value that reflects a second likelihood that the utterance includes the particular hotword, as determined by a second computing device comprises:

receiving, from a server, through a local network, or through a short range radio communication channel, a second value that reflects the second likelihood that the utterance includes the particular hotword.

21. The device of claim 15, wherein the operations further comprise:

determining that the second computing device is configured to respond to utterances that include the particular hotword,

wherein comparing the first value that reflects the first likelihood that the utterance includes the particular hotword and the second value that reflects the second likelihood that the utterance includes the particular hotword is performed in response to determining that the second computing device is configured to respond to utterances that include the particular hotword.

22. A non-transitory computer-readable medium storing software comprising instructions executable by one or more computers which, upon such execution, cause the one or more computers to perform operations comprising:

receiving, by a first computing device, audio data that corresponds to an utterance;

before beginning automated speech recognition processing on the audio data, processing the audio data using a classifier that classifies audio data as including a particular hotword or as not including the particular hotword;

determining, based on the processing of the audio data using the classifier that classifies audio data as including a particular hotword or as not including the particular hotword, a first value that reflects a first likelihood that the utterance includes the particular hotword;

receiving a second value that reflects a second likelihood that the utterance includes the particular hotword, as determined by a second computing device;

comparing the first value that reflects the first likelihood that the utterance includes the particular hotword and the second value that reflects the second likelihood that the utterance includes the particular hotword; and

based on comparing the first value that reflects the first likelihood that the utterance includes the particular hotword to the second value that reflects the second likelihood that the utterance includes the particular hotword, determining whether to begin performing automated speech recognition processing on the audio data.

* * * * *

Exhibit 9



Continuous Speech Recognition

If speech recognition by machine were a solved problem, then work in the area would consist of the application of proven techniques to new tasks, and basic research would not be necessary. By some metrics, this is almost the current situation. Statistical representations

called *Hidden Markov Models* (HMMs) have been developed to the point where impressive recognition performance can be achieved in the laboratory on very large vocabulary speaker-independent continuous speech recognition tasks. Related methods have been transferred to commercial applications on a limited scale, and the current prospect is that many such useful deployments can be done with the existing technology. Further, this technology is now reasonably mature, having been built up over the last twenty years, based on the pioneering work of Baker [3] and Jelinek [36]. Additionally, many of the engineering issues were addressed by speech researchers in the late 1980's and early 1990's under the auspices of the ARPA program [16, 18, 42]. It is now feasible to develop new applications by following mathematical prescriptions that have been developed under these previous programs. Many people have been able to use HMM-based approaches as the core of impressive recognition systems. Recently, one such system, the HMM Tool Kit (HTK) from Cambridge [84] has been widely distributed so that students and developers can build up speech recognition systems based on some of the best known statistical approaches.

However, speech recognition is *not* a solved problem in any fundamental sense. While the existing technology may be sufficient for some specific commercial applications, humans expect speech input systems to behave much as people would. Humans must often contend with unfamiliar accents, background noise, and reflective room acoustics, improper English grammar, and unfamiliar words. Nonetheless, we can generally understand what is being said. Human recognition performance under many realistic conditions is still much better than that of any machine, even for the limited tasks we have undertaken so far.

Since there is great room for improvement, research continues in this area. However, it may be the case that some of the characteristics of mainstream approaches to speech recognition

are limiting factors in the long term road to improvement. For this reason, it is important to examine alternate approaches, such as the neural-network-based method described in this article. This seems particularly justified if there are some theoretical reasons why the

current approaches are suboptimal. However, given the strong base of mathematical tools for statistical speech recognition, it has been useful to modify only a few aspects of the existing approaches at a time, rather than starting from scratch.

The Table shows some of the pros and cons of the dominant speech recognition paradigm, in which HMMs are used to represent the production of speech and are compared with new speech to determine the most likely utterance that could have been said. HMM-based structures and algorithms provide a rich and flexible mathematical framework for building recognition systems. They also feature powerful learning and decoding methods for temporal sequences without requiring any explicit (hand) segmentation in terms of the speech units (typically phones or phonemes) used as a basis for continuous speech training and recognition. Also, they easily accommodate different levels of constraints (e.g., phonological and syntactical constraints), as long as these are expressed in terms of the same statistical formalism. However, to take advantage of this representational power, algorithms must explicitly or implicitly make numerous assumptions about speech, although some of them are obviously unrealistic. For instance, it is often necessary to assume that the features extracted within a phonetic segment are uncorrelated with one another. This is a poor match to most kinds of speech segments and features. Also, HMM training algorithms are based on likelihood maximization, which assumes correctness of the models (which is known not to be true; see, e.g., the correlation problem mentioned above) and implies poor discrimination (since, ideally, training for minimization of the error rate should be based on *a posteriori* probabilities). In a sense, we have settled for a poor model because we know how to use it more effectively than we know how to use more realistic ones.

Connectionist or *Artificial Neural Network* (ANN) methods provide one way to reduce system dependence on unre-

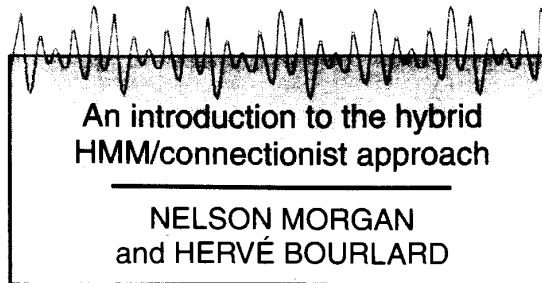


Table: Classical HMM-based Approaches to Continuous Speech Recognition	
Pros	Cons
Rich mathematical framework	Poor discrimination
Powerful learning and decoding methods	Practical requirements for distributional assumptions (e.g., uncorrelated features within an acoustic vector)
Good abstractions for sequences, temporal aspects	1st order Markov model assumption for phone or subphone states
Flexible topology for statistical phonology and syntax	Typically ignore correlation between acoustic vectors

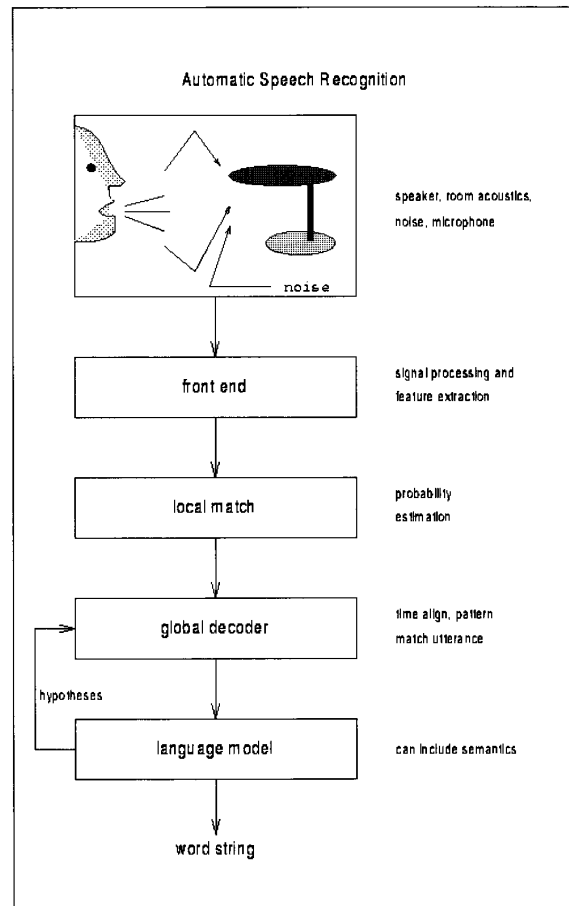
alistic assumptions about speech. Any system can be improved in a variety of ways, and so there is no conclusive evidence that connectionist speech recognition algorithms are better than other approaches. Nonetheless, in a number of controlled experiments, the use of connectionist elements in a larger statistical system has significantly improved performance [2, 47, 65]. Further, connectionist speech research can provide the opportunity to develop a new paradigm that also has a firm theoretical footing, and which may lead to newer innovations that will improve the state of the art. Currently, though, connectionist approaches for continuous speech recognition are embedded within a framework that is essentially a standard HMM, in which only a few of the usual assumptions have been dropped. As research continues, hopefully the models can represent speech more faithfully. However, most laboratories that have published reports of work in continuous speech recognition using ANNs are using them to generate probabilities (or distances) that are used with an underlying model that is explicitly or implicitly an HMM [2, 7, 19, 24, 33, 45, 47]. (Unavoidably, we will tend to give a description that is close to our own implementation; for instance, using a multi-layered perceptron, as described in [56]. The reader should keep in mind that most of the basic ideas should still hold for other hybrid implementations, e.g., using a recurrent net for the probability estimates as done in [33]). For more on this class of hybrid systems, see [9].

Technology Background

Automatic Speech Recognition

The basic task of *Automatic Speech Recognition* (ASR) is to derive a sequence of words from a stream of acoustic information. A more general task is automatic speech understanding, which includes the extraction of meaning (for instance, a query to a database) or producing actions in response to speech. For many applications, interaction between system components devoted to semantics, dialog generation, etc., and the speech recognition subsystem can be critical. However, in order to simplify the focus of this article, we will only consider recognition per se.

ASR systems typically consist of several major compo-



1. Block diagram of continuous speech recognition system.

nents that are illustrated in Fig. 1. Note that the first block, which consists of the acoustic environment plus the transduction equipment (microphone, preamplifier, anti-aliasing filter, sample-and-hold, A/D converter) can have a strong effect on the generated speech representations. For instance, additive noise, room reverberation, microphone position and type, dc offsets in the preamplifier or sample-and-hold, and groundloops in the equipment can all be associated with this part of the process. The second block, the feature extraction subsystem (sometimes called the front end) is intended to deal with these problems, as well as deriving acoustic representations that are both good at separating differing classes of speech sounds and effective at suppressing irrelevant sources of variation. These two blocks, though not discussed further in this article, are worthy of significant study, and like the components devoted to understanding cannot ultimately be completely partitioned from the rest of the recognizer.

The next two blocks in Fig. 1 illustrate the core acoustic pattern matching operations of speech recognition. In nearly all ASR systems, a representation of speech, such as a spectral or cepstral representation [71] is computed over successive intervals, e.g., 100 times per second. These representations or speech *frames* are then compared (in some sense) to the

spectra or cepstra for speech that were used for training, using some measure of similarity or distance. Each of these comparisons can be viewed as a local match. The global match is a search for the best sequence of words (in the sense of the best match to the data), and is determined by integrating many local matches. The local match does not typically produce a single hard choice of the closest speech class, but rather a group of distances or probabilities corresponding to possible sounds. These are then used as part of a global search or decoding to find an approximation to the closest (or most probable) sequence of speech classes, or ideally to the most likely sequence of words. Another key function of this global decoding block is to compensate for temporal distortions that occur in normal speech. For instance, vowels can be shortened in rapid speech, while some consonants may remain nearly the same length.

The most common global decoding approach is some form of dynamic programming [57], in which time warping of the input against possible speech representations results in the most likely sequence of sound categories to match the input. There are many variations to this process, but in general the local computation consists of finding the lowest cost path through possible representations by:

1. For each time step, consider possible transitions from the previous time step.
2. For each such possible transition, take the cost of the sound sequence that has been hypothesized so far, and add it to the cost of the transition.
3. Choose the least costly transition according to this number, and add it to the cost of the local match, keeping track of the pointer to the winning prior sequence. The sum is the current global cost of the sequence that can be backtracked at this point from the pointers that have been saved.
4. At the end of the utterance, backtrack from the lowest global cost to generate the corresponding speech sequence.

This description is greatly oversimplified from what is used in most systems; for instance, the local and transition costs are generally implemented as (negative) log probabilities, so that the sums can be interpreted as giving the most probable sequences. (Note that for a multivariate Gaussian distribution, the exponent is the negative of the Mahalanobis distance between the data vector and the mean vector). These sums then can be viewed as computing products of probabilities in the log domain, which is preferable for a number of reasons (e.g., numerical stability). Additionally, the decoding procedure is often done using different algorithms, for instance using a tree-based search. Nonetheless, the dynamic programming (or Viterbi search for statistical systems) described above is at the base of many recognition systems, including the class described here.

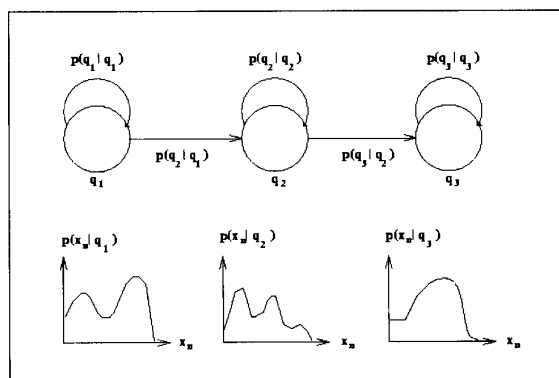
This procedure can also be seen as corresponding to an underlying model of speech, namely that of words consisting of sequences of speech units that can have varying length. Implicitly this means that each speech unit has constant spectral properties until one jumps to the next one, an assumption that is clearly wrong for natural human speech. Nonetheless, it is a simplifying assumption that permits the use of

powerful statistical techniques that will be discussed shortly. The last block in Figure 1 consists of the language model, which determines the hypotheses that are considered in the global search. This block can also process the global decoder output further. For instance, if the decoder generates not only the most likely sentence but rather the N most likely, (e.g., $N = 100$), the language model could rescore the sentence according to grammar or semantics. As with the front end, this research topic will not be described further here.

As noted above, most often the local distance computation is implemented probabilistically. Typically, the probability of an observed spectrum or cepstrum is computed for each possible sound. These probabilities are most commonly estimated using a mixture (weighted sum) of Gaussian distributions, or by vector quantizing the spectra and counting the co-occurrences of spectral prototypes and speech categories in order to derive discrete probability distributions. Additionally, connectionist networks can be used to generate the required probabilities. First, however, we briefly explain the underlying structure of the probabilistic approach.

Hidden Markov Models (HMMs)

HMMs model the sequence of feature vectors as a piecewise stationary process. That is, an utterance $X = \{x_1, \dots, x_n, \dots, x_N\}$ is modeled as a succession of discrete stationary states $Q = \{q_1, \dots, q_K, \dots, q_K\}$, $K < N$ with instantaneous transitions between these states. A HMM is typically defined (and represented) as a stochastic finite state automaton (usually with a left-to-right topology when used for speech). An example of a simple HMM is given in Fig. 2: this could be the model of a short word assumed to be composed of three stationary parts. The approach defines two concurrent stochastic processes: the sequence of HMM states (modeling the temporal structure of speech), and a set of state output processes (modeling the locally stationary character of the speech signal). The HMM is called a "hidden" Markov model because



2. A three state Hidden Markov Model (HMM). A HMM is a stochastic finite state machine, consisting of a set of states and corresponding transitions between states. HMMs are commonly specified by a set of states q_i , an emission probability density $p(x_n | q_i)$ associated with each state, and transition probabilities $p(q_j | q_i)$ for each permissible transition from state q_i to state q_j .

the underlying stochastic process (i.e., the sequence of states) is not directly observable, but still affects the observed sequence of acoustic features.

Ideally, there should be a HMM for every possible utterance. However, this is clearly infeasible for all but extremely constrained tasks; generally a hierarchical scheme must be adopted to reduce the number of possible models. First, a sentence is modeled as a sequence of words. To further reduce the number of parameters (and, consequently, the required amount of training material) and to avoid the need of a new training each time a new word is added to the lexicon, word models are often comprised of concatenated sub-word units. Although there are good linguistic arguments for choosing units such as syllables or demi-syllables, the units most commonly used are speech sounds (phones) that are acoustic realizations of the linguistic categories called phonemes. Phonemes are speech sound categories that are sufficient to differentiate between different words in a language. One or more HMM states are commonly used to model a segment corresponding to a phone. Word models consist of concatenations of phone or phoneme models (constrained by pronunciations from a lexicon), and sentence models consist of concatenations of word models (constrained by a grammar). Theory and methodology for HMMs are described in many sources, including [63]. Briefly, the fundamental equation relevant for this process is a restatement of Bayes' Rule as applied to speech recognition:

$$P(M|X) = \frac{P(X|M)P(M)}{P(X)} \quad (1)$$

in which $P(M|X)$ is the posterior probability of the hypothesized Markov model M (i.e., associated with a specific sequence of words) given an acoustic vector sequence X . Since it is not known how to compute this probability directly (see [6], however, for some theoretical work that suggests an approach to training and recognition with a global $P(M|X)$ criterion), Eq. 1 is usually used to split this posterior probability into a likelihood $P(X|M)$ that represents the contribution of the *acoustic model*, and a prior probability $P(M)$ that represents the contribution of the *language model*. Usually $P(X|M)$ and $P(M)$ are trained separately on independent training corpora. By doing so, and assuming that $P(X)$ in Eq. 1 is independent of the parameters of the models (which is actually not true during training), Eq. 1 turns the acoustic model training into a Maximum Likelihood Estimation (MLE) problem. This actually explains the lack of discrimination mentioned in the table, as systems trained to maximize $P(X|M)$ for the correct model are not trained to minimize that quantity for the incorrect models.

The acoustic likelihood is then computed by expanding it into all possible state paths in M that can generate X :

$$P(X|M) = \sum_{Q} P(X, Q|M) \quad (2)$$

where the sum extends over all possible paths Q of length N in M . This "full" likelihood is sometimes approximated as

$$P^*(X|M) = \max_{Q} P(X, Q|M) \quad (3)$$

which is usually referred to as the "Viterbi" approximation, which is often used for recognition without much loss in performance.

Such a statistical system is trained on acoustic data so that during recognition it produces emission probabilities $p(x_n|q_k)$ (see Fig. 2) that can be multiplied to produce an approximation to the acoustic probability (assuming statistical independence). There exist efficient training algorithms to learn the parameters of the probability estimators. The most common form of this procedure is often called the forward-backward algorithm, in which the estimators for the data likelihoods conditioned on each word model ($P(X|M)$) are iteratively trained [4]. For the Viterbi approximation, the full likelihood is approximated by the likelihood of the most probable path through the states in the models, as given by the dynamic programming procedure. This approximation is sometimes more sensitive to poor initializations, but with a good initialization can be particularly straight forward to implement, and ultimately is more convenient for the approaches described in this article. One form of this iteration, then, could be as follows:

1. Given a set of acoustic training data that is phonetically labeled (possibly with errors), train an estimator to generate the data density for any hypothesized state (speech class); i.e., train an estimator of the emission probability density.

2. Given a probability estimator for the data likelihood of each state, use dynamic programming to find the most likely sequence through the states in all the possible model sequences. This step is sometimes called a forced Viterbi alignment (determining the alignment of the sequence of acoustic training vectors with the corresponding phonetic labels).

This procedure, sometimes called embedded Viterbi learning (as used in the segmental k-means algorithm [63], for instance), can be proved to converge to a local optimum; in practice it is repeated until some stopping criterion has been reached.

Artificial Neural Networks (ANNs)

Multilayer Perceptrons (MLPs)

Our discussion of neural networks for speech will be focused on *Multilayer Perceptrons* (MLPs), which are the most common ANN architecture used for speech recognition. However, all of the basic conclusions about the utility of these structures for estimating probabilities or local costs for an HMM will also hold for other structures such as a recurrent network, as used in [33], or a Time-Delay-Neural-Network (TDNN), as used in [24]. These alternative structures will be briefly discussed later in this section.

Typically, MLPs have a layered feed forward architecture with an input layer (consisting of the input variables), zero or more *hidden* (intermediate) layers, and an output layer, as shown in Fig. 3. Each layer computes a set of linear discriminant functions [22] (via a weight matrix) followed by a nonlinear function, which is often a sigmoid function

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (4)$$

As discussed in [9], this nonlinear function performs a different role for the hidden and the output units. On the hidden units, it serves to generate high order moments of the input; this can be done effectively by many nonlinear functions, not only by sigmoids. On the output units, the nonlinearity can be viewed as a differentiable approximation to the decision threshold of a threshold logic unit or perceptron [69], i.e., essentially to count errors. For this purpose, the output nonlinearity should be a sigmoid or sigmoid-like function. Alternatively, a function called the *softmax* can be used, as it approximates a statistical sigmoid function. For an output layer of K units, this function would be defined as

$$f(x_i) = \frac{\exp(x_i)}{\sum_{n=1}^K \exp(x_n)} \quad (5)$$

MLPs with enough hidden units can (in principle) provide arbitrary mappings $g(x)$ between input and output. MLP parameters (the elements of the weight matrices) are trained to associate a “desired” output vector with an input vector. This is achieved via the *Error Back-Propagation* (EBP) algorithm (see [58], [59], [70] and [81] for multilayer networks; [1], [69], and [83] for single-layer networks; [15] for a control theory version) that uses a steepest descent procedure to iteratively minimize a cost function.

Popular cost functions are, among others, the *Mean Square Error* (MSE) criterion:

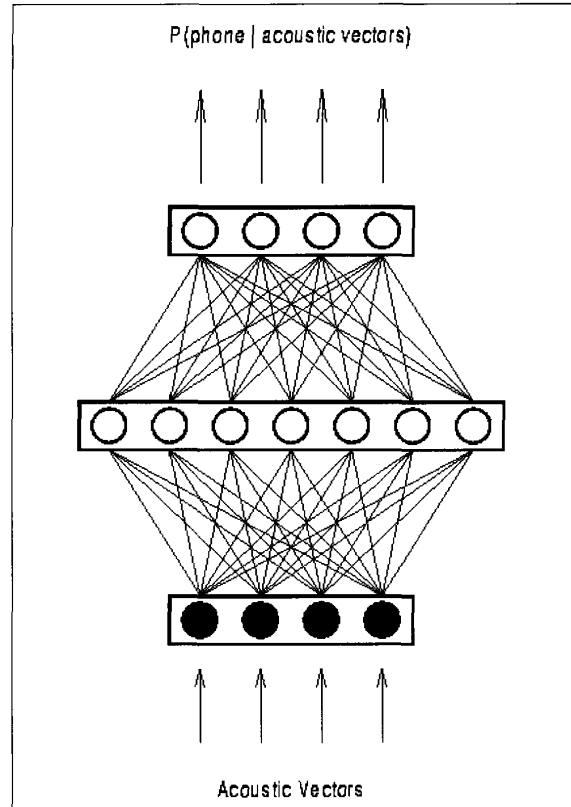
$$E = \sum_{n=1}^N \|g(x_n) - d(x_n)\|^2 \quad (6)$$

or the relative entropy criterion (this criterion can be derived from viewing the network outputs as a posterior distribution over the values of a random variable that is the pattern class (conditioned on the acoustic data), and evaluating a discrete form of the classical definition of relative entropy between the target distribution and this output distribution). The relative entropy criterion is also referred to as Kullback-Leibler distance:

$$E_e = \sum_{n=1}^N \sum_{k=1}^K d_k(x_n) \ln \frac{d_k(x_n)}{g_k(x_n)} \quad (7)$$

where x_n is the pattern to be classified, $d(x_n) = (d_1(x_n), \dots, d_K(x_n), \dots, d_K(x_n))^t$ represents the desired output vector (for classes q_k , $k = 1, \dots, K$), $g(x_n) = (g_1(x_n), \dots, g_K(x_n), \dots, g_K(x_n))^t$ the observed output vector, K the total number of classes, and N the total number of training patterns.

MLPs, as well as other neurally-inspired architectures, have been used in a variety of ways for speech-related tasks. For instance, for some tasks the entire input sequence (e.g., isolated word) is processed at once by the MLP, which views the temporal acoustical sequence as a spatial pattern. For the case of isolated words, for instance, each word can be associated with an output of the network. However, this kind of



3. A generic multi-layer perceptron (MLP). Each horizontal box is a layer of the MLP, and consists of a series of units represented by circles. Each unit represented by an open circle computes a simple function of its inputs, such as a sigmoidal nonlinearity of the weighted sum of its inputs. The darker circles represent the inputs of the network. The intermediate layer is referred to as a hidden layer. The output layer computes phonetic probabilities if the net has been trained under certain conditions, as described in the text.

structure has not been useful for continuous speech recognition and will not be discussed further here.

Other Connectionist Architectures

As we shall see shortly, the hybrid approach is based on a statistical perspective that is valid for any connectionist architecture, given some assumptions about the training procedure. Consequently we are not giving a significant treatment here of the possible neural networks that could be used for this purpose (see [51] or [32] for more about the possible architectures). However, we briefly describe here some of the other kinds of network that can be used in an ANN/HMM hybrid besides the MLP described above.

- **Radial Basis Functions (RBFs)** — many researchers use a variant of a layered feedforward network in which a hidden layer with a fixed-variance Gaussian is used instead of one with sigmoids [13]. Further, these networks commonly use a single Gaussian layer and an output layer that is strictly linear. Once reasonable positions are found for the hidden Gaussians, the optimal output layer weights can be determined analytically with a matrix inversion. These systems can be

trained with much less computation than sigmoid or softmax-based MLPs. The outputs can also be used as probabilities and integrated into a hybrid system [64, 76]. Many researchers have reported, however, that the final performance is somewhat poorer than with the far more computationally expensive MLP.

- **Recurrent Neural Network** — the group at Cambridge University Engineering Department has developed an approach that is very similar to the one described here, except that the emission probability is generated by a recurrent neural network (RNN) [67]. The network uses a set of state units that takes the acoustic input and has a recurrent connection from the output of the state layer back to its input. The state layer output along with the acoustic input is connected to the output layer. The network is trained using back propagation through time. Recent experiments show this method to work as well as the MLP approach, with even fewer parameters. The principal drawback appears to be a somewhat tricky training schedule, due to potential instabilities.

- **Time-Delay-Neural Network** — recurrent networks can be approximated over a finite time period by a feedforward network in which the loops are replaced by the explicit use of several preceding activation values. This feed forward approach was shown to be useful with a single convolutional layer for consonant recognition in 1983 by Makino [48], and later by Waibel et al for a similar task using multiple convolutional layers [79] and finally for hybrid systems such as those described in this article [24]. It is a flexible structure, and in practice many researchers (including us) have used a compromise between the simple MLP and a full TDNN in which delays are only used at the input layer.

Note that in the case of the TDNN and RNN, the hidden and output units can be of the same form as for an MLP (sigmoidal or softmax), and that in all cases the error criteria can be MSE or relative entropy (among others).

ANNs as Statistical Estimators

Estimating HMM Emission Probabilities with an ANN

ANNs can be used to classify speech units such as phonemes or words, typically by mapping temporal representations into spatial ones, or by using recurrences. This is the way ANNs were initially used on simple speech recognition problems (see, e.g., [60, 80, 79]). However, ANNs classifying complete temporal sequences have not been successful for continuous speech recognition. In fact, used as such they are not likely to work well for continuous speech, since the number of possible word sequences in an utterance is generally infinite. Also, we presently do not know of any principled way to translate an input sequence of acoustic vectors into an output sequence of speech units with an ANN only. On the other hand, HMMs provide a reasonable structure for representing sequences of speech sounds or words. Assuming such a structure, one good use for ANNs might be to provide the distance measure for the local match block of Fig. 1.

For statistical recognition systems, the role of the local estimator must be to approximate probabilities. In particular, given the basic HMM equations, we would like to estimate something like the probability $p(x_n | q_k)$ of Fig. 2, that is, the probability of the observed data vector given the hypothesized HMM state (which corresponds to some speech sound). However, HMMs are based on a very strict formalism that is difficult to modify without losing the theoretical foundations or the efficiency of the training and recognition algorithms. Fortunately, ANNs can estimate probabilities that are related to these emission probabilities, and so can be fairly easily integrated into an HMM-based approach. In particular, ANNs can be trained to produce the posterior probability $p(q_k | x_n)$, that is, the a posteriori probability of the HMM state given the acoustic data, if each ANN output is associated with a specific HMM state. This can be converted to emission probabilities using Bayes' rule.

Several authors have shown that the outputs of ANNs used in classification mode can be interpreted as estimates of a posteriori probabilities of output classes conditioned on the input [9], [10], [27], [66]. Before we repeat this proof, the general principle can be geometrically motivated (see Fig. 4).

The figure shows that an equilibrium that is reached in the ideal case does in fact correspond to the network output g being equal to the posterior probability p . Consider an area in feature space around a training pattern x_n , and assume that we consider only two classes: the target class for this pattern, and the class of all patterns that do not belong to the first class. Vectors in the selected area will undergo an upward "force" corresponding to the gradient of the error term for all patterns with a target of 1; the quadratic error term in this case is $\frac{1}{2}(1 - g)^2$, with a derivative of $(1 - g)$, and this case will occur a fraction of the time given by probability p . Therefore, the upward force (average gradient term) applied to the region is equal to $p(1 - g)$. The downward force is similarly defined, and will balance for the equilibrated case. This will only occur when the network output is equal to the posterior probability (thanks to Raul Rojas of Freie Universitaet, Berlin for the figure and for this perspective on the proof).

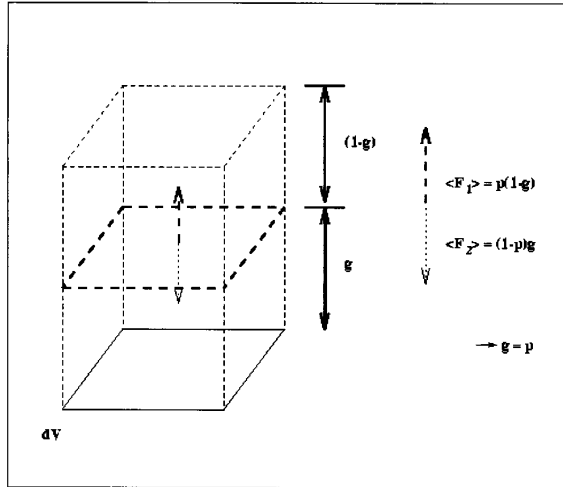
A more formal proof, (see, e.g., [66]), is repeated here. For continuous-valued acoustic input vectors, the MSE in (6) can be expressed as follows:

$$E = \int p(x) \sum_{k=1}^K \sum_{l=1}^K p(q_k | x) [g_l(x) - d_l(x)]^2 dx \quad (8)$$

Since $p(x) = \sum_{i=1}^K p(q_i, x)$, we have:

$$E = \int \sum_{i=1}^K \left[\sum_{k=1}^K \sum_{l=1}^K [g_l(x) - d_l(x)]^2 p(q_k | x) \right] p(q_i, x) dx$$

After a little more algebra, using the assumption that $d_l(x) = \delta_{kl}$ if $x \in q_k$, and adding and subtracting $p^2(q_i | x)$ in the previous equation leads to:



4. Geometric illustration of proof that ANNs will (ideally) produce posterior probabilities. Equilibrium is reached when the upward force, (due to the average deviation of the ANN output from 1 for a target that should be high), is equal to the downward force (due to the average deviation from 0 for a target that should be low).

$$E = \int \sum_{i=1}^K \left[\sum_{j=1}^K (g_i(x) - 2g_i(x)p(q_i|x) + p^2(q_i|x)) \right] p(q_i, x) dx \quad (9)$$

$$+ \int \sum_{i=1}^K \left[\sum_{j=1}^K (p(q_i|x) - p^2(q_i|x)) \right] p(q_i, x) dx$$

$$= \int \sum_{i=1}^K \left[\sum_{j=1}^K (g_i(x) - p(q_i|x))^2 \right] p(q_i, x) dx$$

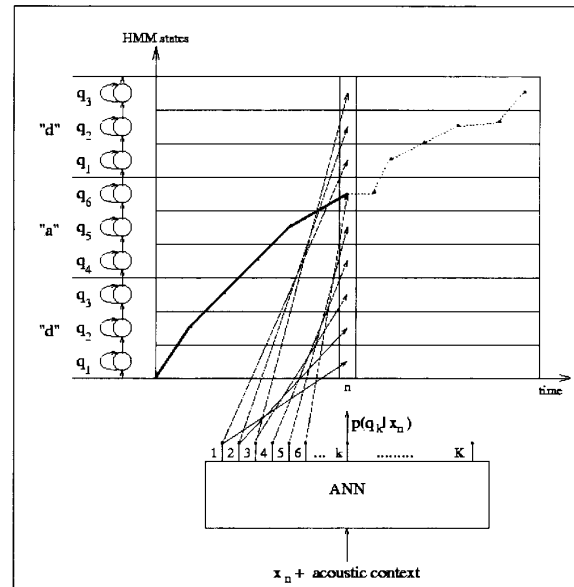
$$+ \int \sum_{i=1}^K \left[\sum_{j=1}^K (p(q_i|x)(1 - p(q_i|x))) \right] p(q_i, x) dx$$

Since the second term in this final expression (Eq. 9) is independent of the network outputs, minimization of the squared-error cost function is achieved by choosing network parameters to minimize the first expectation term. However, the first expectation term is simply the mean squared-error between the network output $g_i(x)$ and the posterior probability $p(q_i|x)$. Minimization of Eq. 8 is thus equivalent to minimization of the first term of Eq. 9, i.e., estimation of $p(q_i|x)$ at the output of the MLP. This shows that a discriminant function obtained by minimizing the MSE retains the essential property of being the best approximation to the Bayes probabilities in the sense of mean square error. A similar proof was given in [66] for the relative entropy cost function.

Since these proofs are only based on the minimized criterion (and not on the architecture of the network), they are valid for any of the ANNs discussed previously, given two conditions:

1. The system must be sufficiently complex (e.g., contain enough parameters) to be trained to a good approximation of the mapping function between input and the output class, and
2. The system must be trained to a global error minimum (where mean squared error and relative entropy are error criteria that will work for this purpose)

It has been experimentally observed that, for systems trained on a large amount of speech, the outputs of a properly



5. At every step n , the acoustic vector x_n with right and left context is presented to the net (Figure 3). This generates local probabilities that are used, after division by priors, as local scaled likelihoods in a Viterbi dynamic programming algorithm. The arrows coming up from each ANN output symbolize the use of these scaled likelihoods (after taking the negative logarithm) as distances from the acoustic input to their corresponding state. The dark solid line shows the best path through the models up to time n (that can be determined by backtracking through pointers), and the dashed path shows its continuation that can be determined once the distances are computed for the last frame in the data.

trained MLP do in fact approximate posterior probabilities (see Fig. 6.1 in [9]), even for error values that are not precisely the global minimum. (Recently, Lippmann [46] has noted that "local minima usually represent alternative solutions which change decision regions and posterior probabilities only far from the training data. They thus have little effect on performance when there is sufficient training data.")

Thus, emission probabilities can be estimated by applying Bayes' rule to the ANN outputs. Commonly, we actually compute

$$\frac{p(x_n|q_k)}{p(x_n)} = \frac{p(q_k|x_n)}{p(q_k)} \quad (10)$$

That is, we divide the posterior estimates from the ANN outputs by estimates of class priors, namely the relative frequencies of each class as determined from the class labels that are produced by a forced Viterbi alignment of the training data. The scaled likelihood of the left hand side can be used as an emission probability for the HMM, since, during recognition, the scaling factor $p(x_n)$ is a constant for all classes and will not change the classification.

Figure 5 shows the basic hybrid scheme, in which the ANN generates posterior estimates that can be transformed

into emission probabilities as described above, and then used in dynamic programming either for forced alignment (when the word sequence is assumed) or for recognition (when word sequences are hypothesized).

Why This is Good

Since we ultimately derive essentially the same probability with an ANN as we would with a conventional (e.g., Gaussian mixture) estimator, what is the point? There are at least two potential advantages that we and others have observed:

1. As noted in the introduction, standard statistical recognizers require strong assumptions about the statistical character of the input, such as parameterizing the input densities as mixtures of Gaussian densities with no correlation between features, or as the product of discrete densities for different features that are assumed to be statistically independent. This type of assumption is not required with an ANN estimator, which will be an advantage particularly when a mixture of feature types are used, e.g., binary and continuous. Specifically, standard HMM approaches require the assumption that successive acoustic vectors are uncorrelated. For the MLP estimator, multiple inputs can be used from a range of time steps, and the network will learn something about the correlation between the acoustic inputs. Note that the use of such a network will lead to a more general emission probability that may also be used in the global decoding. That is, if $c + d + 1$ frames of acoustic vectors $X_{n-c}^{n+d} = \{x_{n-c}, \dots, x_n, \dots, x_{n+d}\}$ are used as input to provide contextual information to the network, the output values of the ANN will estimate $p(q_k | X_{n-c}^{n+d})$, $\forall k = 1, \dots, K$. This provides a simple mechanism for incorporating acoustic context into the statistical formulation. Of course, ANNs are not the only way to incorporate such context. Many current systems use first and second time derivatives [25, 62] computed over a span of a few frames, allowing very limited acoustical context modelling. Some systems transform a context window of a few adjacent frames (typically 3-5 frames in total) with Linear Discriminant Analysis (LDA), which finds a linear transformation that maximizes the between-class variance while minimizing the within-class variance (see, e.g., [28]). The neural network can be seen as a generalization of these approaches that permits arbitrary weights and a nonlinear transformation of the input data.

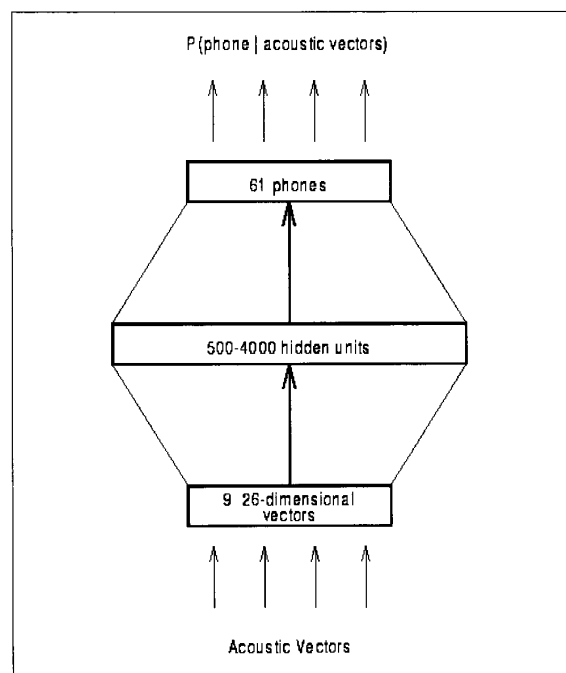
2. ANNs are a good match to discriminative objective functions (e.g., mean squared error), and so the probabilities will be optimized to maximize discrimination between sound classes, rather than to most closely match the distributions within each class. It can be argued that such a training is conservative of parameters, since the parameters of the estimate are trained to split the space between classes, rather than to represent the volumes that compose the division of space constituting each class. Some of the results cited in Section 5 appear to support the notion that fewer parameters are required for these systems than in non-discriminative systems with similar performance (see, in particular, [47]).

We and others have performed numerous experiments that have verified these two points. In some of them, a fixed HMM was used and alternate probability estimators were substituted [7, 56, 9, 65, 67, 47]. When these experiments were controlled for the number of parameters, there have been significant improvements using the approaches described here. Some of this quantitative evidence will be briefly summarized when we discuss the results.

Hybrid HMM/MLP Recognition System

The Basic System

As mentioned above, we and others have discriminatively trained large neural networks to estimate HMM emission probabilities for continuous speech recognition. In particular, systems have been developed to perform speech recognition for 1000 and 5000 word vocabularies, given millions of examples of feature vectors for training. At our laboratory, we have focused on using a simple MLP that is illustrated in Fig. 6, though similar results have been achieved at other labs with structures such as RNNs. It is deceptively simple, consisting of a single large hidden layer, typically with between 500 and 4000 hidden units that receive input from several hundred acoustic variables (e.g., 9 frames of acoustic context consisting of 12th order Perceptual Linear Prediction coefficients (PLP-12) [30] and log energy, along with their derivatives, or 26 features per frame). (Many experiments have been



6. Acoustic vectors from the current frame, four previous, and four following frames are processed by a single large hidden layer. The output corresponds to phonetic categories used for labeling of the TIMIT database.

done in our lab that resulted in this choice of input features. Some of these are reported in [56].) The output typically corresponds to simple context-independent acoustic classes such as phones defined for the TIMIT phonetic database, using 61 phones. Each word model consists of a succession of phone models, and each phone model uses a single density, with emission probabilities calculated from MLP outputs via Bayes' rule.

Despite this apparent simplicity, there are some significant characteristics of this system that have appeared to be necessary for good performance. The major points are summarized in the following sections; for further explanation, see [9].

Training

We and others have used on-line training instead of off-line (true gradient) back propagation. In this approach, the weights are adjusted in the direction of the error gradient with respect to the weight vector, as estimated from a single pattern. With an accurate estimate of the error gradient, one could proceed in the direction of the local training minimum. However, the per-pattern gradient estimate can be viewed as a noisy estimate of the gradient over the entire training set. The size of the learning step can be viewed as the magnitude of the noise; in the limit, very large learning steps move over the error surface randomly, while very small steps closely correspond to the true gradient. In fact, it can be beneficial to have more noise (larger steps) initially, in order to escape from potentially poor local solutions. Additionally, given realistic training data, which is typically quite redundant, each full pass through the data represents many passes through similar subsets, and thus can be relatively efficient.

In practice, using on-line gradient search and a relative entropy error criterion, only a small number of passes through the data are required to phonetically train the network (typically 1 to 5).

In addition to the use of on-line training, other aspects of the training method include:

- **Cross-validation** — It is necessary to use a stopping criterion based on an independent portion of the data, i.e., utterances that are not used for training. While this is a good general rule in training pattern classifiers, most of the early published suggestions for neural network stopping criteria were measures based on the training set, e.g., gradient magnitude or slope. The networks that were ultimately successful for continuous speech recognition are quite large, often using hundreds of thousands to millions of parameters. These nets are susceptible to overfitting the training data, resulting in bad probability estimation and very poor generalization performance on the test set.

In addition to merely halting the training based on performance for an independent validation set, a training procedure can be used in which the learning rate is also adjusted to improve generalization [54]. Specifically, the learning rate is reduced (typically by a factor of 2) when cross-validation indicates that a given rate is no longer useful. Additionally, we have empirically noted that after the first reduction, only

a single epoch at each rate is useful. The heuristic of only permitting a single pass for any learning rate after the initial one cuts down the number of epochs by almost a factor of two, and has little effect on final performance.

- **Training criterion** — Using relative entropy (7) instead of the MSE criterion speeds convergence. The correction resulting from this criterion is always linear and does not saturate when the output values are at the extremes (tails) of the sigmoid (where the correction for the MSE criterion is negligible).

- **Initialization of output biases** — Histograms of the output biases of phonetically trained MLPs showed a narrow distribution around a strongly negative value (typically around -4). This is no coincidence, since the input to the sigmoid nonlinearity for an output unit produces the log odds,

or $\log \frac{p(q|x)}{1-p(q|x)}$ when the output produces $p(q|x)$. When the

evidence from the data is equivocal, this is roughly equal to $\log \frac{p(q)}{1-p(q)}$, and since these each $p(q)$ is much less than 1, the

sigmoid input is roughly equal to $\log p(q)$. This is a rough argument, and for specific distributions (such as a Gaussian) it can be shown to be inaccurate. Nonetheless, the empirical observation (from histograms) is that it is roughly true, at least in the sense that the average output bias of the converged network is close to the average log prior probability. Additionally, it has been confirmed that initializing the biases to the rough range that they will ultimately approach speeds convergence, and slightly improves the results.

- **Random pattern presentation** — In earlier forms of our analysis we presented the data sequentially according to the speech signal. Sequential presentation of the acoustic vectors to the net (i.e., in the order that they were spoken) can cause slow convergence, requiring a very low learning rate in the case of on-line training. In the current method, the speech vectors are presented at random (preserving the relative frequencies of the classes), which speeds up ANN training, and also slightly improves the results. In a variation this approach for practical training using speech databases whose size exceeds the physical memory, blocks of sequential sentences (which can be randomized at the sentence level) are read from disk into physical memory, and frames can be presented randomly from within the block. In both schemes, it does not appear to matter whether random sampling is done with or without replacement (i.e., it does not matter whether each random frame choice is constrained to be a different one than had already been chosen).

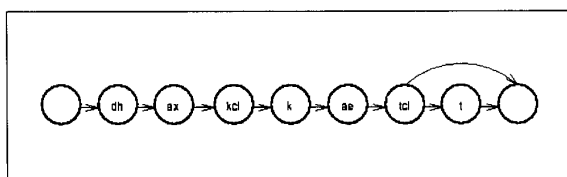
We note here that for the case of RNN training, sequential frame presentation is necessary since the structure is one with an infinite impulse response. However, in practice RNN-based hybrid systems have provided equivalent performance to that provided by MLP-based hybrids.

Division by Priors

As noted earlier, in the current HMM/ANN paradigm, data likelihoods are estimated by applying Bayes' Rule to the ANN

outputs, or, in practice, dividing each posterior probability by the corresponding class priors to get scaled data likelihoods (as shown in Eq. 10). However, it can also be shown that, in theory, HMMs can be trained using local posterior probabilities as emission probabilities [9], resulting in models that are both locally and globally discriminant. (See the discussion on Current Related Research Areas, transition-based approaches, for a brief description of some current work in this area, which is described more fully in [6]). For current systems, there are generally mismatches between the prior class probabilities implicit to the training data and the priors that are implicit to the lexical and syntactic models that are used in recognition. For instance, Fig. 7 shows the HMM for a pronunciation of "the cat." The topological definition given in the figure will (in combination with all of the other models) determine the prior probability for each phone during recognition; for instance, if the sound "ax" only occurs after "dh," then the prior probability of the former would be dependent on the prior probability of the latter sound. Depending on both this collection of pronunciation models and the language model, each phone in a sequence like "the cat" will have some prior probability of occurring that may be a poor match to the relative frequencies in the training set, particularly if the pronunciation models come from dictionaries and if the language model is inferred from large text corpora. This can result in significant degradations in recognition performance when the posteriors are used for recognition directly. Thus, it is generally safer to divide the ANN outputs by class priors, taking care to handle the cases of classes that rarely or never occur in the training set, leading to negligible or zero values for the estimates of the class priors.

On the other hand, it would ultimately be desirable to infer statistical word and word sequence models that are consistent with the acoustic training data (or at least to take advantage of the prior information implicit to the acoustic training data). For this case, it would (in principle) be preferable to use posterior probability estimates from the network. We have recently conducted some experiments with the inference of

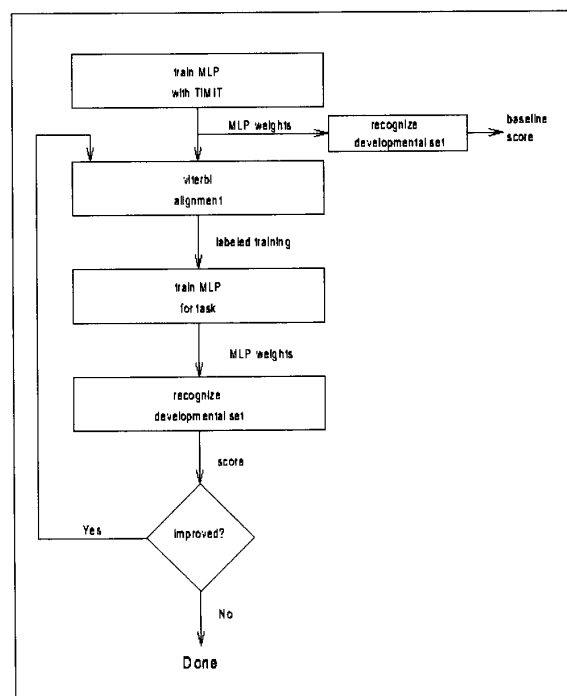


7. Simplified pronunciation model for the phrase "the cat" using Darpa-bet symbols for the phones. The "dh" symbol refers to the voiced form of "th," that is, one in which the vocal cords are vibrating. The initial consonant of the second word is represented by two states, one corresponding to the k-closure and the second corresponding to the k-burst. The final consonant of the second word is represented by two states, one corresponding to the t-closure and the second corresponding to the t-burst, but often the t-burst is omitted; hence the alternate path skipping the sound. The states without labels are non-emitting states representing the start and finish of the phrase.

pronunciation from data (see discussion of results), and the preliminary observation is that division by class priors is not required in this case.

Embedded Alignment

ANNs trained for classification require supervision (labeled targets for each pattern). An early problem in applying ANN methods to speech recognition was the apparent requirement of hand-labeled frames for ANN training. Since the ANN outputs can be used in the dynamic programming for global decoding (after division by the prior probabilities), it is possible to use embedded Viterbi training to iteratively optimize both the segmentation and the ANN parameters. In this procedure, illustrated in Fig. 8, each ANN training is done using labels from the previous Viterbi alignment. In turn, an ANN is used to estimate training set state probabilities, and dynamic programming given the training set models is used to determine the new labels for the next ANN training. Of course, as for standard HMM Viterbi training, one must start this procedure somewhere, and also have a consistent criterion for stopping. Many initializations can be used, including initializing the training set segmentation linearly or in proportion to average phoneme durations. More recently we have achieved better results initializing the procedure by training an ANN on a standard hand-segmented corpus (TIMIT for the case of American English), and using this ANN to align the training set for any new unlabeled corpus.



8. Embedded Viterbi learning with MLP.

Incorporating Constraints

The statistical interpretation of ANNs permits the flexible incorporation of model constraints. For instance, gender or phonetic context can be viewed as an additional random variable whose probability is estimated jointly with that of the phonetic unit. This joint probability can in turn be estimated, without simplifying assumptions, as the product of two simpler probabilities that are each estimated by ANNs.

For a constrained model, we may wish to estimate the joint probability of a current HMM state with a constraint class (such as phonetic context). Using C to represent the set of possible constraints, we wish to estimate $p(q_k, c_j | x_n)$, where $c_j \in C = \{c_1, \frac{1}{4}, c_j\}$. If there are J constraint classes, this will require $K \times J$ output units for an ANN estimator. However, using the definition of conditional probability, the desired expression can be broken down in two ways as follows [8, 19]:

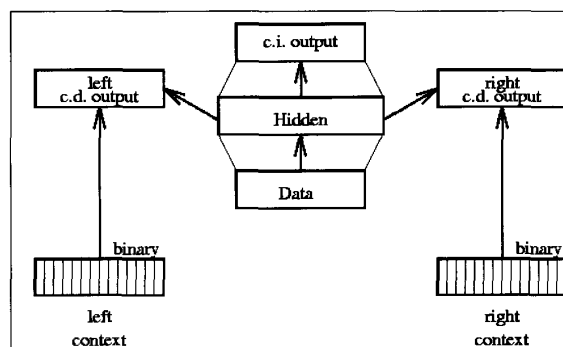
$$p(q_k, c_j | x_n) = p(q_k | x_n) p(c_j | q_k, x_n) = p(c_j | x_n) p(q_k | c_j, x_n) \quad (11)$$

Thus, the desired probability is the product of an unconstrained posterior probability and a new conditional. The former can be realized with a network like that of figure 6 (though for the second case the output targets are context classes). Viewing an ANN as an estimator of the probability of the left side of a conditional given the right side as input, the $K \times J$ output probabilities can be estimated by two MLPs with K and J output units respectively.

This approach can be applied to contextual constraints to estimate biphone probabilities with two networks, or triphone probabilities with three. In each case, the use of extra conditionals removes any assumption of independence.

The conditionals can be implemented in a number of ways. For instance, a net can be constructed with sparsely coded binary inputs representing the classes (where all inputs are zero except for the class that we are conditioning on). These inputs can either be connected to a hidden layer, or, for greater computational simplicity, directly to an output layer. This latter architecture, as applied to phonetic context, is illustrated in Fig. 9.

In collaborative work with SRI International [19], we tested this class of approach for left and right generalized biphone context. In this work, an architecture was used in which the output layer was replicated for each of 8 possible broad phonetic contexts (8 for the left and 8 for the right, each associated with one state of a 3-state model). This worked somewhat better than the architecture of Fig. 9, but at the cost of a larger number of parameters. On the other hand, most of them were not modified for any given training frame, so that the computation was not significantly larger. In both cases, the weights to the context-dependent output layers were initialized to values learned from context-independent training, so that cross-validation (early stopping) determined a smooth compromise between fully relying on the context-dependent data (which is sparser for each class) and the context-independent training [19]. These experiments showed significant improvements over the simpler context-inde-



9. Context dependent neural network. Each output layer corresponds to a state of a 3-state HMM. The central output layer is trained in a context-independent manner. The left context-dependent output layer is only trained for frames that are labeled as the first state, and receives an input from a binary layer that codes the broad category of left context, i.e., only one input is on for any particular context, as indicated by the transcription during training or by the current hypothesis during recognition. The right context-dependent output layer is trained similarly.

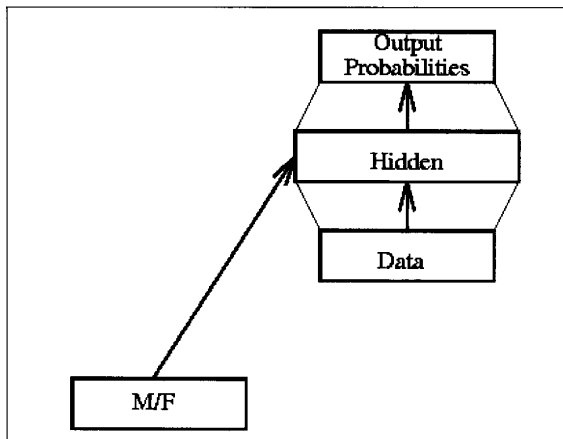
pendent approach, eliminating roughly one-fifth of the errors in a speaker-independent Resource Management test set [23]. The resulting system was comparable in accuracy to an HMM system that used many more parameters and which modeled much more detailed context, including word-specific models for the most common words [19]. Researchers at the Oregon Graduate Institute have also recently reported very large improvements for one of their tasks, using an architecture and method similar to that used by SRI [38].

Thus, statistical factorization of ANN probabilistic estimators appears to have practical significance. The approach has also been applied in other ways, for instance for gender and combinations of speaker-specific models [41]. See Fig. 10 for an illustration of a network trained with gender dependency. This is also an example of the incorporation of long-term consistency constraints, as the input in this case is constrained to either be male or female for the entire utterance.

Some Results

The major focus in this paper is to describe the ideas that are in common to a family of methods that are being investigated by a large number of laboratories. As such, we have felt that a strong emphasis on numerical results would be a diversion from the major message—it is not difficult to come up with sets of results that show improvements of method X over method Y. However, we do include here a brief litany of results from several laboratories that seem to confirm some of the major points of this article:

1. It is possible to train networks that are quite large (over a million weights) on millions of speech training patterns with very few epochs; the resulting networks can be used to estimate emission probabilities for HMMs in large and difficult tasks in continuous speech recognition. This was demonstrated in [52], where we described a 1.6 million-weight



10. A gender-dependent MLP. Gender is represented by 1 or 2 bits that are additional variables presented to the network. For instance, for the 1-bit case, the binary input will be turned on for one gender and off for the other. During recognition, one gender or the other is assumed throughout an utterance, and the gender corresponding to the best overall score is chosen. Alternatively, nets can be trained separately for each gender. A gender classification net is used to weight the two choices, analogous to context dependent networks.

network that was trained on 6 million frames of speech from the *Wall Street Journal* pilot data. This simple estimator was then used to get 16% error on the 5000-word vocabulary evaluation set using a standard bigram grammar. A smaller network (with roughly 1/4 of the parameters) gave over 20% error on the same test set, showing that at least for our training paradigm and architecture we benefited greatly from the larger number of parameters.

2. For a number of tasks, simple tied-Gaussian mixture systems do not perform as well as hybrid HMM/ANN systems that use a similar number of parameters and the same input features. For equivalent performance, the classical HMM system must be made much more complicated (for instance, typically using context-dependent models and many more parameters). For instance, for the 1000 word vocabulary, perplexity 60 wordpair grammar Resource Management continuous speech recognition task, it was reported in [65] that a context-independent version of SRI's DECIPHER system had 11% word error on a particular Feb. 1991 evaluation test set. The same system using MLP outputs as probability estimates achieved 5.8% errors using a similar number of parameters. Even better performance could be achieved by a context-dependent version of DECIPHER (see item 5 below), but this required the use of detailed context and many more parameters. In the same report it was shown that relatively simple forms of context could be incorporated in the network estimators as well and could achieve similar performance as tied-mixture estimators using much more detailed models of context and an order of magnitude more parameters. For further discussion about this with more results, see [68].

In [47], similar conclusions were also drawn for quite a different example, connected digit recognition for a standard

TI database. In this case, string error for a moderate-sized MLP (about 11000 parameters) was about 2.5%, while the string error for a 28000 parameter tied mixture system was 3.8 percent. In order to get comparable performance, the number of mixtures had to be expanded so that there were an order of magnitude more parameters than in the MLP case.

3. The best current pure-HMM systems currently outperform the best current hybrid systems on many tasks. Again, the HMM systems are frequently much more complicated, with many parameters and complex forms of smoothing; frequently the improvement over the simpler hybrid system is small. For instance, in the connected digit case described above [47], by expanding to over 200000 parameters, Lubensky et al were able to achieve a 2% string error, which is an error rate reduction of 25 relative to the system incorporating the MLP estimator. For large vocabulary recognition this has also been true as of this writing.

4. Smoothed combinations of hybrid HMM/ANN systems and purely HMM-based systems appear to often give better performance than either one alone. Often this is done by smoothing together probabilities or log probabilities at the frame level. For instance, this was shown for the connected digits example given above. In that case, combining emission probability estimates for the best Gaussian mixture system with those from the MLP gave roughly a 1.7% string error, which was the best performance reported. Similarly, in [65], the same MLP estimator that yielded a 5.8% word error on the Resource Management task was used to improve a complex Gaussian mixture system from 3.8% error to 3.2% error. In both experiments, the researchers smoothed together log emission probability estimates. Thus, on quite different tasks studied by unrelated laboratories, it was observed that a very good HMM-based system using Gaussian mixture estimators could be further improved by smoothing with estimates from an ANN. Similar results have been observed in other laboratories as well.

5. In work at BBN [2], the subsystems were combined in a different way (by taking a list of the most likely N sentences as estimated by a pure HMM system, and reordering them based on phonetic segment probabilities as estimated by an MLP), but they too reported consistent improvements over the simpler system.

Current Related Research Areas

The following are a few areas of current research interest in our own laboratory, but similar work is being done at a number of labs around the world. Hybrid HMM/ANN methods are also being applied to non-speech applications such as handwriting (see, e.g., [72]).

Learning Pronunciations with Nets

The quality of phonological models have sometimes been observed to have a significant effect on ASR performance [17]. It is also our hypothesis that discriminant models such as a phonetically trained ANN can have severe problems when the

phonological model is very inaccurate (since the ANN may strongly discriminate against the correct phone) and when class priors observed on the training data are not sufficiently representative of priors used in the recognition models.

In recent work at ICSI on a restaurant query speech understanding system called the Berkeley Restaurant Project (BeRP) [39], Chuck Wooters developed an approach for generating models of multiple pronunciations using a modification of our iterated training. Briefly, this procedure consisted of the following steps (sketched in Fig. 11):

1. Initialize the neural network architecture by training on TIMIT, which (as noted previously) is a phonetically hand-labeled database.

2. Initialize the word models to be several equally probable possibilities, as determined by dictionary look-up, phonological rule generation, and/or examples from hand-labeled speech (e.g., TIMIT).

3. Run the training data forward through the network, generating posterior probabilities. Use these probabilities as suggested in [9], our results in these experiments confirmed that division by priors was not actually necessary when the pronunciations were learned from the data. This result was probably due to the fact that the phone priors that are implicit to the induced models are fairly similar to those implied by the relative frequency of phone classes in the training data). Otherwise, use scaled likelihoods (after division by priors) in a dynamic programming procedure (forced Viterbi) to determine the alignment between word models and speech. This will result in the recognition of pronunciations in the data.

4. Using the framewise phone labels determined by the

previous step, train the neural network. Training techniques are the same as those described earlier in this article.

5. Generate new word models by including all examples of pronunciations as labeled in the training data. Merge them using the Stolcke-Omohundro HMM merging technique [77], which uses a Bayesian criterion to create a HMM from examples.

6. Run steps 3 through 5 until performance on an independent cross-validation set does not improve appreciably.

In our internal BeRP task, inclusion of multiple pronunciations reduced word recognition errors by 20% (i.e., one-fifth of the errors went away), and semantic recognition errors (i.e., errors in filling out the database query) by 30 percent. We are currently applying this approach to some other tasks of interest to us.

Segmental Approaches

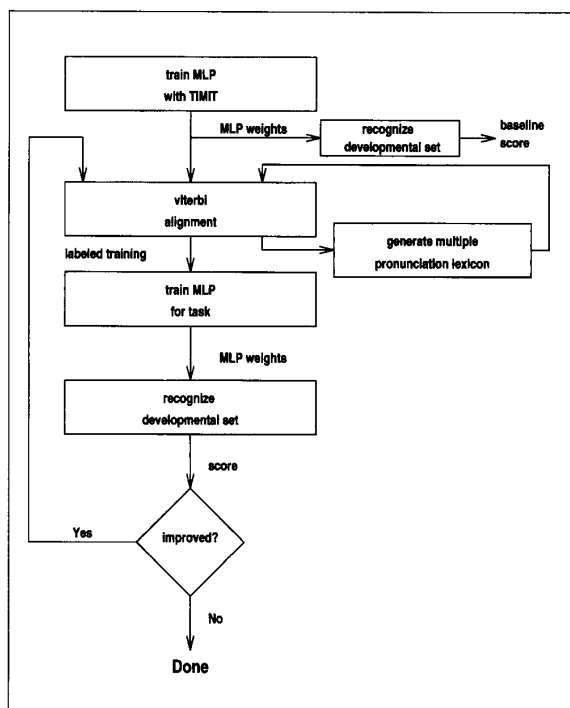
One of the principal limitations of standard HMM-based approaches is the required assumption of independent and identically distributed (i.i.d.) acoustic vectors within the speech segment corresponding to a state. In fact, not only are there significant dependencies between these vectors, but the nature of this dependence may be well represented by a dynamic relationship that succinctly describes the speech sound in a way that will be helpful for discrimination from other sounds. For these reasons, a number of researchers have recently focused on statistical representations of complete segments, as opposed to i.i.d. statistical estimates for sub-segmental frames [20, 21, 26].

In general, these approaches are used to estimate segment likelihoods. An alternate approach would be to estimate segment posteriors, and to incorporate some representation of the segment dynamics into the model. We have done some experimentation in this direction in which we have explicitly incorporated the time index within a segment as part of the network input. In another class of approaches, segments can be modeled as being produced by an explicit articulatory model. Finally, attention can be focused on the transitions rather than the relatively stationary portions of segments in an attempt to model the perceptual sensitivity to such regions. This has been done for diphone likelihoods [26], and a posterior model focusing on transitions is described in [55].

In all of these cases, there is an attempt to better model the nonstationary nature of speech. All of this work is at an early stage, both for connectionist and non-connectionist approaches. However, it is possible that connectionist algorithms may provide a useful paradigm for the incorporation of more faithful representations of speech.

Transition-based Approaches

In the previous section, we briefly mentioned that it might be desirable to model transitions in speech. One approach to this is to use time derivative features [25]. In general, though, the problem of modeling (non-stationary) transitions is still an open one. Another step in this direction is to use highpass or



11. Embedded multi-pronunciation Viterbi learning with MLP

bandpass filtering of critical band trajectories (RASTA processing) to emphasize transitions [31]. While this is sometimes helpful in reducing errors due to mismatches between training and testing conditions, the resulting observation sequence is a representation that has emphasized the regions of strong change and de-emphasized temporal regions without significant spectral change. This is a mismatch to the underlying speech model in standard HMMs, which has been designed to represent piecewise stationary signals. Therefore, it is likely that transition-based systems will require a fundamentally different kind of underlying statistical model. The segment-based approaches mentioned in the previous section offer one possibility. Another is to model speech as a succession of auditory events or *avents*, separated by relatively stationary periods (ca. 50-150 ms). Avents correspond to times when the spectrum and amplitude are rapidly changing, which are believed to be the most important regions for phonetic discrimination [25]. The stationary periods are mapped to a single tied state, and so modeling power is focused on regions of significant change. This is described further in [55].

It is implicit in transition classification that the transition occurs at some unique time (e.g., for a single frame). This is difficult for a classifier to learn, as the neighboring acoustic vectors are often extracted from time regions with very similar acoustics. We are currently investigating an approach in which an iterative procedure reminiscent of the Baum-Welch algorithm (which is used to iteratively converge to Maximum Likelihood estimates) is used to generate "soft" labels for each frame that correspond to estimates of the posterior probability of each possible transition. These estimates are generated in order to optimize the global posterior probability (known to minimize the actual classification error rate) for the sequence of word models, given a sequence of acoustic vectors. A neural network can then be trained to approximate the mapping to this probability. Algorithms to do this and a proof of their convergence can be found in [6].

Merging Neural Networks as Statistical Experts

Multiple networks can, in principle, be used together in order to improve performance over a single "expert." Additionally, this kind of approach can be used to speed training; this is the application that is discussed here.

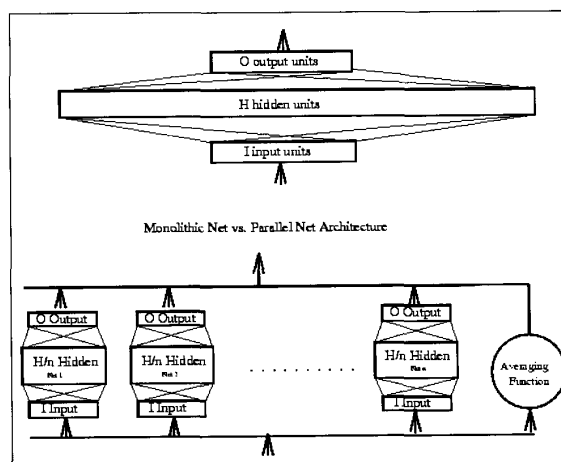
As we have gradually increased the size of our speech corpora and networks, the computation required for training has greatly increased. Surprisingly, the increase has been sub-quadratic, as the number of required passes through the data has decreased. For instance, for training with six million frames of *Wall Street Journal* data, a net with 1.6 million free parameters trained with a single pass through the data (though the initial weights were loaded from a TIMIT-trained network, which required several passes through a significantly smaller database). However, as the corpora get even larger, and the nets correspondingly get larger to take advantage of the increased data, training times could become prohibitive, even with fast hardware. There are a number of possible

solutions to this problem, some of which we are currently investigating:

- **Modularization with broad classes** — while a number of early promising experiments were done with broad class classification, we had found that a single large network with many fine classes appeared to work quite well given sufficient training data. However, it is possible that breaking up the probability estimation into several pieces can ultimately be more computationally efficient. In some preliminary experiments with small networks that discriminate between each class and all others, we have observed comparable performance to what we achieve with a monolithic network [74]. While performance was not improved, and even the number of parameters seemed roughly comparable (for comparable performance), communication between elements was greatly improved, which could be helpful for future parallel implementations.

- **Merging random data splits** — the complete fine category network can also be split into two or more networks that are each trained by random splits of the data. The probability estimates from each network are averaged, as illustrated in Fig. 12. If the corpus is large enough that each split is representative of the complete space, this provides a way to both decrease communication and computation, as each of the nets can be made proportionately smaller as well. (Of course, the smaller network must still be large enough to perform the required mapping function.) Preliminary tests seem to show that this works [50].

- **Merging known categories** — as noted earlier, gender-based splits of the data have been successfully used to train separate networks (or to modify monolithic networks with a gender input variable). Other categories, such as rate of speech, dialect region, or unsupervised data clusters could be used to define a partition of the data for the training of multiple networks. An approach of this kind was proposed in [29]. Speaker-dependent estimators for voiced stop consonant probabilities were weighted and summed with gating elements trained with error back-propagation. More generally, a



12. Training multiple smaller probability estimation MLPs instead of a single monolithic network.

number of authors have recently discussed the mixture of information from separately-trained experts [35]. During recognition, two approaches can be taken: either weight each network with probabilities from the gating network on every frame, or perform multiple recognitions based on the assumption of one partition being the "correct" one for the duration of the utterance. The latter approach probably makes sense for classes such as gender that don't ordinarily change in mid-sentence (speaker consistency).

- **Selective sampling** — for extremely large corpora, a fraction of the data is sufficient to train the parameters to values that are quite close to the optimum values. For further refinement by the rest of the corpus, it is likely that only examples near the boundaries are necessary. It is possible that training time could be reduced significantly by some combination of random and selective choice of training examples. During training, the boundaries are unknown, but the network that has been trained in such a way can nonetheless provide class probabilities for the statistical search. However, the class priors must be compensated for as with the other methods described earlier.

It is often the goal to apply some of these approaches to improve classification performance, in addition to providing modularity as described above. However, for the large vocabulary continuous speech recognition tasks (with a large training database) that we have explored, we have not observed such improvements in comparison to very large monolithic neural networks.

System Issues

We have justified our interest in connectionist approaches by the known limitations of current systems, and potential relief from these limitations with newer methods. However, connectionist algorithms have at least one apparent disadvantage: training requires orders of magnitude more computation than the more common HMM training paradigm. This is a necessary consequence of phonetically discriminant training, in which all parameters are potentially modified by patterns belonging to any class. As noted earlier, large vocabulary HMM systems frequently have many more parameters than are used in ANNs; however, since they are typically not trained discriminatively, most parameters are not modified for any particular pattern (those parameters associated with classes other than the class of the pattern).

As a result, the amount of computation required for ANN training can be orders of magnitude more than is required for traditional HMM training. Because of this, connectionist speech researchers have found it necessary to use fast computational hardware. In our own laboratory, we have worked on developing computational systems (both hardware and software) to facilitate the training [53]. These computers need to be fully programmable machines, since general programs must run on them as we alter our training paradigms in the course of research. On the other hand, having some architectural specialization to make common training operations efficient can save significant amounts of time.

In the recognition process, neural network evaluation is only moderately more computationally intensive than the evaluation of large Gaussian mixtures. Although this cost must be considered, most of the time required for large searches and grammars is typically due to memory access rather than raw computation. Therefore, what is generally required for large vocabulary recognition is a large memory that is connected with high bandwidth to a processor that can be easily programmed, regardless of whether ANNs or Gaussian mixtures are used for the emission probability estimation.

As workstations continue their inevitable climb to higher performance over the rest of this decade, many laboratories will be able to experiment with neural network estimators, as the increased costs (even for training) will no longer be a significant limitation for many problems. However, experimental work with the largest models will still require enough computation for training so that the increased cost (relative to pure HMM systems) may still justify some specialized systems.

Other Connectionist Approaches

This article has focused on the hybrid HMM/ANN approach, in which some kind of network (typically an MLP, RBF, RNN, or TDNN) trained for classification by an MSE or relative entropy criterion is used to estimate probabilities or distances to be used in dynamic programming matching to a HMM. This is currently the most common application of neural networks to continuous speech recognition. However, a range of other approaches and subproblems in speech recognition are being investigated by researchers. We briefly list here the most common of these.

- **Predictive networks** — In this case, ANNs are not trained to perform phonetic (HMM state) classification, but instead are trained (according to a MSE criterion) as an autoregressive (AR) model to predict a feature vector given some previous number of feature vectors and the assumption of a particular HMM state. This can be shown to be a variant of the maximum likelihood estimate in which the dynamic of the system (acoustic correlation) is taken into account. This could either be done using multiple networks (one per state) [78], or with one network that uses the state identity as a control input [43]. Such systems have been successfully trained to do connected digit recognition. They also have sometimes been used for larger tasks, but apparently have not been as successful as classification-based hybrids. In any event, the predictive systems are also used as hybrids, in that a search such as the Viterbi is used to integrate the information from the network, which in this case is a prediction error signal. These networks are the nonlinear equivalent of autoregressive HMMs [61]. Potentially, predictive networks can represent segment dynamics, once we learn how to use them effectively; however, they do not alleviate the piecewise stationarity assumption that is endemic to classical HMM-based approaches.

- **ANN models of HMMs** — In this article, we have

discussed ways in which connectionist algorithms can replace others in generating probabilities used in an HMM. In some other work, researchers have shown that connectionist structures can be used to represent standard HMM-based algorithms. Once such subsystem was the Viterbi network [44]. In this case, each HMM is associated with a neural network in which each output corresponds to a HMM state. Each output unit is complemented by time-delayed connections between the different output units to represent the topology of the HMM, followed by a comparator sub-network to compute the minimum of the activation values of output nodes at the previous time step. This network does implement the core of the dynamic time warp algorithm, but it does not replace the pointer bookkeeping that is required for a practical Viterbi implementation.

In another connectionist formulation of what had previously been considered an entirely non-connectionist algorithm, the Alpha-Net [11] was introduced to simulate the forward recurrence of the forward-backward HMM algorithm. The Alpha Net is recurrent, the recurrent units are linear, and the acoustic vectors enter the loop via a multiplication to simulate the operation of a standard HMM state using a full likelihood criterion (as opposed to the Viterbi criterion).

- Global optimization through nonlinear transformation — Instead of using the network outputs as probabilities, the network can be viewed as providing a general nonlinear transformation of the observation vectors that are then used for an otherwise standard HMM-based system. This permits a global optimization of the input transformation together with a global training of the HMMs [5].

- Preprocessing — Many researchers have used feature map representations, related to one of the formulations from Kohonen and collaborators [40], to generate feature representations for a speech recognizer. In other designs, researchers have experimented with networks to provide mappings from noisy to clean data [75] or from a new speaker to an old speaker [34].

- Postprocessing — Because of interest in more complex models of speech and the difficulty in training such models (such as the segmental models mentioned above), some researchers have used more computationally demanding approaches as a post processing step in speech recognition. In this method, the primary recognizer generates either a list of candidate utterances (commonly referred to as an N-best list), or a lattice of potential words and associated acoustic information (such as emission probabilities from the primary system). The secondary system then rescores the candidate list or reprocesses the lattice hypotheses with some new criterion. For instance, in the case of the Segmental Neural Network [2], networks are trained on phonetic segments as determined from Viterbi alignments in the training set, and then are run to generate probabilities for sequences of segments in each hypothesized utterance. The resulting scores are blended with the scores from the primary system, and have been shown to improve overall performance. Although this approach generates segment probabilities as

opposed to the frame probabilities of the approach focused on in this article, it is still close in spirit, as it generates posterior phone probabilities and then uses them in an HMM-based system as part of a global decoding strategy.

Summary and Challenges for the Future

In this article, we have focused on a tutorial description of the hybrid HMM/ANN method. The approach has been applied to large vocabulary continuous speech recognition, and variants are in use by many researchers. The method provides a mechanism for incorporating a range of sources of evidence without strong assumptions about their joint statistics, and may have applicability to much more complex systems that can incorporate deep acoustic and linguistic context. The method is inherently discriminant and conservative of parameters.

Despite these potential advantages, the hybrid method thus far has been focused on implementing fairly simple systems, which do surprisingly well on large continuous speech recognition tasks. We and others are only beginning to explore the use of more complex structures with this paradigm. In particular, we are just beginning to look at the connectionist inference of language models (including phonology) from data, which may be required in order to take advantage of locally discriminant probabilities rather than simply translating to likelihoods.

Finally, our current intuition is that more advanced versions of the hybrid method can greatly benefit from a perceptual perspective. In such an approach, we may be able to use some simple properties of the auditory system to greatly constrain the space of important dependencies between each state and the prior states and acoustic vectors. (Some initial theory for this extension is described in [55], and the mathematical basis for training of such systems with a global maximum *a posteriori* (MAP) criterion as described in [6]). This could permit a much more comprehensive model than we are currently using in any of our HMM-based systems, including those with connectionist probability estimators. Such models may be necessary in order to determine the ultimate utility of neural networks for speech recognition.

Nelson Morgan and Hervé Boulard are with the International Computer Science Institute, Berkeley, CA 94704. Nelson Morgan is also affiliated with the University of California at Berkeley, and Hervé Boulard with the Faculté Polytechnique de Mons in Belgium.

References

1. S. Amari, "A theory of adaptive pattern classifiers," *IEEE Trans. on Elec. Com.*, vol. EC16, pp. 279-307, 1967.
2. G. Zavaliagkos, Y. Zhao, R. Schwartz, and J. Makhoul, "A hybrid segmental neural net/hidden markov model system for continuous speech recognition" *IEEE Trans. on Speech and Audio Processing*, vol. 2, no. 1, pp. 151-160, 1994.
3. J. Baker, "The DRAGON System - An overview," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 23, no. 1, pp. 24-29, 1975.

4. L. Baum, "An inequality and associated maximization techniques in statistical estimation of probabilistic functions of Markov processes," *Inequalities*, no. 3, pp. 1-8, 1972.
5. Y. Bengio, R. De Mori, G. Flammia, R. Kompe, "Global optimization of a neural network-Hidden Markov Model hybrid," *IEEE Trans. on Neural Networks*, vol. 3, no. 2, pp. 252-259, 1992.
6. H. Bourlard, Y. Konig and N. Morgan, "REMAP: recursive estimation and maximization of a posterior probabilities Application to transition-based connectionist speech recognition," *ICSI Technical Report TR-94-064*, 1994.
7. H. Bourlard and N. Morgan, "A continuous speech recognition system embedding MLP into HMM," in *Advances in Neural Information Processing Systems 2* (D. S. Touretzky, Ed.), pp. 413-416. Morgan Kaufmann, San Mateo CA, 1990.
8. H. Bourlard and N. Morgan, N. "CDNN: A context dependent neural network for noncontinuous speech recognition," *IEEE Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing* (San Francisco, CA), pp. II:349-352, 1992.
9. H. Bourlard and N. Morgan, *Connectionist Speech Recognition - A Hybrid Approach*, Kluwer Academic Press, 1994.
10. H. Bourlard and C.J. Wellekens, "Links between Markov models and multilayer perceptrons," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 1167-1178, 1990.
11. J.S. Bridle, "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition," in *Neurocomputing: Algorithms, Architectures and Applications*, F. Fogelman Soulié and J. Hérault (Eds.), NATO ASI Series, pp. 227-236, 1990.
12. J. S. Bridle, "Alpha-Nets: a recurrent neural network architecture with a hidden Markov model interpretation," *Speech Communication*, vol. 9, pp. 83-92, 1990.
13. D. Broomhead, D. Lowe, "Multi-variable functional interpolation and adaptive networks," *Complex Systems*, vol. 2, pp. 321-355, 1988.
14. P. F. Brown, "The Acoustic-Modelling Problem in Automatic Speech Recognition," *PhD Thesis*, School of Computer Science, Carnegie Mellon University, 1987.
15. A. Bryson and Yu Chi Ho, *Applied Optimal Control*, Blaisdel Publishing Company, 1969.
16. Y. Chow, M. Dunham, O. Kimball, M. Krasner, G. Kubala, J. Makhoul, P. Price, S. Roucos, and R. Schwarz, "BYBLOS: The BBN continuous speech recognition system," *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing*, Dallas, Texas, pp. 89-92, 1987.
17. M. Cohen, "Phonological Structures for Speech Recognition," *PhD Thesis*, University of California at Berkeley, 1989.
18. M. Cohen, H. Murveit, J. Bernstein, P. Price, and M. Weintraub, "The DECIPHER speech recognition system," in *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing* (Albuquerque, NM), pp. 77-80, 1990.
19. M. Cohen, H. Franco, N. Morgan, D. Rumelhart, and V. Abrash, "Context-dependent multiple distribution phonetic modeling," in *Advances in Neural Information Processing Systems 5* (S. J. Hanson, J. D. Cowan, and C. L. Giles, Eds.), pp. 649-657, 1993.
20. L. Deng, "A generalized hidden markov model with state-conditioned trend functions of time for the speech signal," *Signal Processing*, 27:65-78, 1992.
21. V.V. Digalakis, J.R. Rohlicek, and M. Ostendorf, "Segment-based stochastic models of spectral dynamics for continuous speech recognition," *IEEE trans. on Speech and Audio Processing*, 1(4):431-442, October 1993.
22. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, Wiley Interscience, New York, 1973.
23. H. Franco, M. Cohen, N. Morgan, D. Rumelhart, and V. Abrash, "Context-dependent connectionist probability estimation in a hybrid hidden Markov model-neural net speech recognition system," *Computer Speech and Language*, vol. 8, no. 3, pp. 211-222, July 1994.
24. M. Franzini, K. F. Lee, and A. Waibel, "Connectionist Viterbi training: a new hybrid method for continuous speech recognition," *IEEE Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing*, pp. 425-428, Albuquerque, NM, 1990.
25. S. Furui, "Speaker independent isolated word recognizer using dynamic features of speech spectrum," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 34, no. 1, pp. 52-59, 1986.
26. O. Ghitza and M.M. Sondhi, "Hidden markov models with templates as non-stationary states: an application to speech recognition," *Computer Speech and Language*, 2:101-119, 1993.
27. H. Gish, "A probabilistic approach to the understanding and training of neural network classifiers," in *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing* (Albuquerque, NM), pp. 1361-1364, 1990.
28. R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," *IEEE Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing*, pp. I-13-16, San Francisco, CA, 1992.
29. J. Hampshire and A. Waibel, "Connectionist architectures for multi-speaker phoneme recognition," in *Advances in Neural Information Processing Systems 2* (D. S. Touretzky, Ed.), Morgan Kaufmann, CA, 1990.
30. H. Hermansky, "Perceptual Linear Predictive (PLP) analysis of speech," *Journal of the Acoust. Soc. Am.*, vol. 87, no. 4, 1990.
31. H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Transactions on Speech and Audio Processing, special issue on Robust Speech Recognition*, vol. 2 no. 4, pp. 578-589, Oct., 1994
32. J. Hertz, A. Krogh, and R. Palmer, *Introduction to the Theory of Neural Networks*, Addison Wesley, 1991.
33. M. Hochberg, S. Renals, and A. Robinson, "ABBOT: The CUED hybrid connectionist-HMM large-vocabulary recognitionsystem," in *Proc. ARPA Spoken Language Technology Workshop*, 1994.
34. X.D. Huang, K.F. Lee, and A. Waibel, "Connectionist speaker normalization and its application to speech recognition," *Proc. of IEEE Workshop on Neural Networks for Signal Processing*, pp. 357-366, IEEE Press, 1991.
35. R. Jacobs and M. Jordan M. "Linear piecewise control strategies in a modular neural network architecture," *IEEE Trans. on Systems, Man, and Cybernetics*, March/April 1993, vol. 23, nr. 2, pp. 337-345, 1993
36. F. Jelinek, "Continuous speech recognition by statistical methods," *Proceedings of the IEEE*, vol. 64, no. 4, pp. 532-555, 1976.
37. F. Jelinek "Self-organized modelling for speech recognition," in *Readings in Speech Recognition*, A. Waibel and K. Lee (eds.), pp. 450-503, Morgan Kaufmann, 1990.
38. L. Jiang and E. Barnard, "Choosing contexts for neural networks," *Oregon Graduate Institute Technical Report*, 1994.
39. D. Jurafsky, C. Wooters, G. Tajchman, J. Segal, A. Stolcke, and N. Morgan, "The Berkeley restaurant project," in *Proc. Intl. Conf. on Spoken Language Processing* (Yokohama, Japan), In Press, 1994.
40. T. Kohonen, "The 'neural' phonetic typewriter," *IEEE Computer* : 11-22, 1988.
41. Y. Konig, N. Morgan, C. Wooters, V. Abrash, M. Cohen, and H. Franco, "Modeling consistency in a speaker independent continuous speech recognition system," in *Advances in Neural Information Processing Systems 5* (S. J. Hanson, J. D. Cowan, and C.L. Giles, Eds.), pp. 682-687, 1993.
42. K.F. Lee, *Large vocabulary speaker-independent continuous speech recognition: The SPHINX system*, Kluwer Academic Publishers, 1988.
43. E. Levin, "Speech recognition using hidden control neural network architecture," in *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing* (Albuquerque, NM), pp. 433-436, 1990.
44. R.P. Lippmann, "Review of neural networks for speech recognition," *Neural Computation*, vol. 1, no. 1, pp. 1-38, 1989.
45. R.P. Lippmann and E. Singer, "Hybrid neural-network/HMM approaches to wordspotting," *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, Minneapolis, Minn., pp. I-565-568, 1993.
46. R. Lippmann, personal communication, 1994.
47. D.M. Lubensky, A.O. Asadi, and J.M. Naik, "Connected digit recognition using connectionist probability estimators and mixture-gaussian densities," *IEEE Proc. of the Intl. Conf. on Spoken Language Processing*, pp. 295-298, Yokohama, Japan, 1994.
48. S. Makino, T. Kawabata, and K. Kido, "Recognition of consonant based

- on the Perceptron model," *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, Boston, Mass., pp. 738-741, 1983.
49. M. Minsky, S. Papert, *Perceptrons*, Cambridge, MA: MIT Press, 1969.
50. Mirghafori, N., Morgan, N., and Bourlard, H., "Parallel training of MLP probability estimators for speech recognition: a gender-based approach" *IEEE Workshop on Neural Networks for Signal Processing*, Greece, pp. 289-298, 1994.
51. D. P. Morgan and C. L. Scofield, *Neural Networks and Speech Processing*, Kluwer Academic Publishers, 1991.
52. N. Morgan, "Big Dumb Deural Nets (BDNN): a working brute force approach to speech recognition," *Proceedings of the ICNN*, vol. VII, pp. 4462-4465, 1994.
53. N. Morgan, J. Beck, P. Kohn, J. Bilmes, E. Allman, and J. Beer, "The Ring Array Processor (RAP): a multiprocessing peripheral for connectionist applications," *Journal of Parallel and Distributed Computing, Special Issue on Neural Networks*, vol. 14, pp. 248-259, 1992.
54. N. Morgan and H. Bourlard, "Generalization and parameter estimation in feedforward nets: some experiments," in *Advances in Neural Information Processing Systems 2* (D. S. Touretzky, Ed.), San Mateo, CA: Morgan Kaufmann, pp. 630-637, 1990.
55. N. Morgan, H. Bourlard, S. Greenberg, and H. Hermansky, "Stochastic Perceptual Auditory-Event-Based Models (SPAM) for speech recognition," *Intl. Conference on Spoken Language Processing*, pp. 1943-1946, 1994.
56. N. Morgan, H. Hermansky, H. Bourlard, P. Kohn, and C. Wooters, "Continuous speech recognition using PLP analysis with multilayer perceptrons," *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing*, pp. 49-52, Toronto, Canada, 1991.
57. H. Ney, "The use of a one-stage dynamic programming algorithm for connected word recognition," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 32:263-271, 1984.
58. D. Parker, Invention Report S81-64, File 1, Office of Technology Licensing, Stanford University, 1982.
59. D. Parker, "Learning logic," *Technical Report TR-47*, Center for Computational Research in Economics and Management Science, MIT, Cambridge, MA, 1985.
60. S.M. Peeling R.K. Moore, "Isolated digit recognition experiments using the multi-layer perceptron," *Speech Communication*, vol. 7, pp. 403-409, 1988.
61. A. Poritz, "Linear predictive Hidden Markov Models and the speech signal," *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing*, pp. 1291-1294, Paris, 1982.
62. A.B. Poritz and A.L. Richter, "On hidden Markov models in isolated word recognition," *IEEE Proc. Intl. Conf. on Acoustics, Speech, and Signal Processing*, pp. 14. 3. 1-4, Tokyo, Japan, 1986.
63. L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-285, 1989.
64. S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist optimization of tied mixture Hidden Markov Models," in *Advances in Neural Information Processing Systems 4* (J. Moody, S. Hanson, and R. Lippmann, Eds.), San Mateo, CA: Morgan Kaufmann, pp. 167-174, 1992.
65. S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist probability estimators in HMM speech recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 2, no. 1, pp. 161-174, 1994.
66. M.D. Richard and R.P. Lippmann, "Neural network classifiers estimate Bayesian a posteriori probabilities," *Neural Computation*, no. 3, pp. 461-483, 1991.
67. T. Robinson and F. Fallside, "A recurrent error propagation network speech recognition system," *Computer Speech and Language*, no. 5, pp. 259-274, 1991.
68. T. Robinson, L. Almeida, J.M. Boite, H. Bourlard, F. Fallside, M. Hochberg, D. Kershaw, P. Kohn, Y. Konig, N. Morgan, J.P. Neto, S. Renals, M. Saerens, C. Wooters, "A neural network based, speaker independent, large vocabulary, continuous speech recognition system: The WERNICKE Project," *Proc. EUROSPEECH'93* (Berlin, Germany), pp. 1941-1944, 1993.
69. F. Rosenblatt, *Principles of Neurodynamics. Perceptrons and the Theory of Brain Mechanisms*, Spartan Books, Washington, 1962.
70. D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing* (D.E. Rumelhart and J.L. McClelland, Eds.), vol. 1, pp. 318-362. MIT Press, Cambridge MA, 1986.
71. R. Schaefer and L. Rabiner, "Digital representations of speech signals," in *Proceedings of the IEEE*, vol. 63, no. 4, pp. 662-667, 1975.
72. M. Schenkel, I. Guyon, I., D. Henderson, "On-line cursive script recognition using neural networks and hidden Markov models," *Proc. of Intl. Conf. on Acoustics, Speech, and Signal Processing*, pp. II. 637-640, Adelaide, Australia, 1994.
73. R. Schwartz, Oral presentation, *Speech Research Symposium XIII*, Johns Hopkins, 1993.
74. R. Salomon, Oral presentation, ICSI, July, 1994.
75. H. Sorenson, "A cepstral noise reduction multi-layer network," *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing*, Toronto, Canada, pp. 933-936, 1991.
76. E. Singer and R. Lippmann, "A speech recognizer using radial basis function neural networks in an HMM framework," *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing* (San Francisco, CA), pp. 629-632, 1992.
77. A. Stolcke and S. Omohundro, "Hidden Markov model induction by Bayesian model merging," in *Advances in Neural Information Processing Systems 5* (S.J. Hanson, J.D. Cowan, and C.L. Giles, Eds.), San Mateo, CA, Morgan Kaufmann, 1993.
78. J. Tebelskis and A. Waibel, "Large vocabulary recognition using linked predictive neural networks," in *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing* (Albuquerque, NM), pp. 437-440, 1990.
79. A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, "Phoneme recognition: neural networks vs. hidden Markov models," in *Proc. IEEE Intl. Conf. on Acoustic, Speech, and Signal Processing* (NY, NY), pp. 107-110, 1988.
80. R. Watrous and L. Shastri, "Learning phonetic features using connectionist networks: an experiment in speech recognition," in *Proc. First Intl. Conf. on Neural Networks*, (San Diego, CA), vol. 2, pp. 619-627, 1987.
81. P. Werbos, "Beyond regression: new tools for prediction and analysis in the behavioral sciences," *PhD Thesis*, Harvard University, Cambridge, MA, 1974.
82. P.J. Werbos, "Back propagation through time: what it does and how to do it," *Proceedings IEEE*, vol. 78, pp. 1150-1160, 1990.
83. B. Widrow and M. Hoff, "Adaptive Switching Circuits," *Technical Reports 1553-1*, Stanford University, Electron. Labs., Stanford, CA, 1960.
84. P. Woodland, and S. Young, "The HTK tied-state continuous speech recognizer," *Eurospeech '93*, pp. 2207-2210, 1993.
85. C. Wooters, "Lexical modeling in a speaker-independent speech understanding system," *ICSI Technical Report TR-93-068*, also a UC Berkeley PhD Thesis.

Exhibit 10

**UNITED STATES INTERNATIONAL TRADE COMMISSION
WASHINGTON, D.C.**

**Before The Honorable Cameron R. Elliot
Administrative Law Judge**

In the Matter of

**CERTAIN AUDIO PLAYERS AND
COMPONENTS THEREOF II**

Investigation No. 337-TA-1330

**RESPONDENT SONOS, INC.'S INITIAL INVALIDITY CONTENTIONS
(DECEMBER 7, 2022)**

or in combination with one or more other references cited in Exhibits A-1 through A-6 and/or Rider A.

Exhibit	Patent Reference	Asserted Claims	Grounds
A-1, Rider A	US8340975 (“Rosenberger”)	1-4, 7, 9-12, and 14-15	Anticipation and/or Obviousness
A-2, Rider A	US7516068 (“Clark”)	1-4, 7, 9-12, and 14-15	Anticipation and/or Obviousness
A-3, Rider A	US2013/0073293 (“Jang”)	1-4, 7, 9-12, and 14-15	Anticipation and/or Obviousness
A-4, Rider A	US2013/0238326 (“Kim”)	1-4, 7, 9-12, and 14-15	Anticipation and/or Obviousness
A-5, Rider A	JPA2003223188 (“Masahide”)	1-4, 7, 9-12, and 14-15	Anticipation and/or Obviousness
A-6, Rider A	The VoicePod System	1-4, 7, 9-12, and 14-15	Anticipation and/or Obviousness

1. Level of Ordinary Skill in the Art

To assess the level of ordinary skill of a POSITA, the following factors can be considered: (i) the type of problems encountered in the art, (ii) the prior solutions to those problems, (iii) the rapidity at which innovations are made, (iv) the sophistication of the technology, and (v) the level of education of active workers in the relevant field.

Based on the foregoing, Sonos contends that a POSITA for the ’748 Patent as of the earliest possible priority date (i.e., June 24, 2014) would have had the equivalent of a four-year degree from an accredited institution (typically denoted as a B.S. degree) in electrical or computer engineering, or a related field, and approximately 2-4 years of experience in the field of voice and/or audio processing, or an equivalent level of skill, knowledge, and experience.

2. Obviousness & Motivations to Combine

Each prior art reference cited herein may be combined with one or more other prior art references to render obvious the Asserted Claims of the ’748 Patent, as explained in more detail below and in the attached Initial Invalidity Charts and Rider. The disclosures of these references

1. Lack of Enablement and/or Written Description

By way of example, the following limitations of the Asserted Claims of the '748 Patent lack enablement and/or written description:

- ***“determine ... whether the first computing device should perform speech recognition on the spoken audio input,” “responsive to determining that the first computing device should perform speech recognition on the spoken audio input: ... send ... perform ... determine ...”*** – Independent claims 1, 7, and 11 each recites this sequence of limitations. However, the specification lacks enablement and written description support for this recited sequence. Indeed, the specification only mentions the term “speech recognition” twice in a more general sense. *See, e.g., id.*, 4:57-63, 10:21-25. There is no disclosure of a first computing device “determin[ing] whether [it] should perform speech recognition on the spoken audio input,” much less performing additional functions (i.e., send, perform, and determine) based on such a determination. Thus, independent claims 1, 7, and 11 are invalid. Dependent claims 2-4, 9-10, 12, and 14-15 are invalid for at least the same reasons.
- ***“a command to temporarily refrain from performing speech recognition”*** – Independent claims 1, 7, and 11 each recite this limitation. However, the specification lacks enablement and written description support for this limitation, as there is no disclosure of “a command to temporarily refrain from performing speech recognition.” Worse yet, Google has interpreted this limitation such that the limitation can be met by a message that does not cause the recipient to “temporarily refrain from performing speech recognition.” In this respect, the specification lacks enablement and written description support for such a message. Thus, independent claims 1, 7, and 11 are invalid. Dependent claims 2-4, 9-10, 12, and 14-15 are invalid for at least the same reasons.
- ***“determine, based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input ... responsive to determining that the first computing device should perform speech recognition on the spoken audio input: ... perform speech recognition on the spoken audio input”*** – Independent claims 1, 7, and 11 each recite this sequence of limitations. Assuming that there is any support for “perform[ing] speech recognition on the spoken audio input” in the '748 Patent, Google’s infringement reads involve an interpretation of this sequence of limitations that involves the “computing devices” to “perform speech recognition on the spoken audio input” in order to (i.e., before) “determine, based at least in part on information received from the one or more additional computing devices, whether the first computing device should perform speech recognition on the spoken audio input.” The specification lacks enablement and written description support for Google’s interpretation. Thus, independent claims 1, 7, and 11 are invalid. Dependent claims 2-4, 9-10, 12, and 14-15 are invalid for at least the same reasons.

3. Level of Ordinary Skill in the Art

To assess the level of ordinary skill of a POSITA, the following factors can be considered: (i) the type of problems encountered in the art, (ii) the prior solutions to those problems, (iii) the rapidity at which innovations are made, (iv) the sophistication of the technology, and (v) the level of education of active workers in the relevant field.

Based on the foregoing, Sonos contends that a POSITA for the '311 and '128 Patents as of the earliest possible priority date (i.e., October 9, 2014) would have had the equivalent of a four-year degree from an accredited institution (typically denoted as a B.S. degree) in electrical or computer engineering, or a related field, and approximately 2-4 years of experience in the field of voice and/or audio processing, or an equivalent level of skill, knowledge, and experience.

4. Obviousness & Motivations to Combine

Each prior art reference cited herein may be combined with one or more other prior art references to render obvious the Asserted Claims of the '128 and/or '311 Patents, as explained in more detail below and in the attached Initial Invalidity Charts and Riders. The disclosures of these references also may be combined with information known to POSITAs, before the earliest possible effective filing date of the '311 and '128 Patents, and understood and supplemented in view of the common sense of POSITAs at the time of the alleged invention, including any statements in the intrinsic record of the '311 and '128 Patents and related applications.

A POSITA would have been motivated to combine the above prior art based on the nature of the problem to be solved, the teachings of the prior art, and the knowledge of POSITAs. The identified prior art references, including portions cited in the Initial Invalidity Charts and Riders, address the same or similar technical issues and suggest the same or similar solutions to those issues as the Asserted Claims of the '311 and '128 Patents. Thus, on a limitation-by-limitation basis, Sonos expressly intends to combine one or more prior art

6. Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to one of ordinary skill in the art; and
7. Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

Certain of these rationales are discussed more specifically in the attached Initial Invalidity Charts and Riders. The fact that others are not discussed more specifically should not be interpreted as an admission or concession that it does not apply. To the contrary, the discussion in the Initial Invalidity Charts and Riders simply provides more explanation of these specific rationales.

Sonos further contends that the prior art identified in these Initial Invalidity Contentions is evidence of simultaneous or near-simultaneous independent invention by others of the alleged invention as recited in one or more of the Asserted Claims of the '311 and '128 Patents. Sonos reserves its right to rely on the simultaneous or near-simultaneous independent invention by others as further evidence of the obviousness of the Asserted Claims of the '311 and '128 Patents.

As noted above, each limitation of the Asserted Claims of the '311 and '128 Patents was well-known to POSITAs before October 9, 2014 (the earliest possible priority date), as detailed in the attached Initial Invalidity Charts and Riders. As noted, Google has not provided evidence indicating that the Asserted Claims are entitled to this earliest possible priority date.

The limitations recited in the Asserted Claims of the '311 and '128 Patents are mere combinations and modifications of these well-known limitations. A POSITA would have been able, and motivated, to improve the existing technology in the same or similar manner by

combining or modifying the individual limitations that were already known in the art to yield predictable results.

5. Secondary Considerations

Sonos incorporates by reference its objections and response to Google's Interrogatory No. 87. Sonos is not aware of any secondary-considerations evidence demonstrating non-obviousness of the Asserted Claims of the '311 and '128 Patents. Indeed, Google has not met its burden to demonstrate a nexus between the alleged inventions of the Asserted Claims and the evidence cited by Google in its response to Sonos's Interrogatory No. 6. See Google's Objections and Responses to Sonos's First Set of Interrogatories (Nos. 1-44), pp. 16-18.

As discussed in the attached Initial Invalidity Charts and Riders, the prior art references confirm that the Asserted Claims of the '311 and '128 Patents would have been obvious to a POSITA before the earliest possible priority date. Sonos reserve the right to supplement or modify these factors to address any evidence or arguments later identified by Google.

E. Invalidity Grounds Under 35 U.S.C. § 112

Subject to Sonos's reservation of rights above, Sonos identifies its grounds of invalidity for each Asserted Patent based on lack of enablement, written description, and indefiniteness pursuant to 35 U.S.C. §§112 (a), (b).

1. '311 Patent

a. Lack of Enablement and/or Written Description

By way of example, the following limitations of the Asserted Claims of the '311 Patent lack enablement and/or written description:

- ***“communicating to the other devices of the plurality of electronic devices to forgo responding to the detected output”*** – Dependent claim 12 recites this limitation. The specification lacks enablement and written description support for “communicating to the other devices of the plurality of electronic devices to forgo responding to the detected

output.” Indeed, no such communication is found in the specification. For at least these reasons, dependent claim 12 is invalid.

- ***“in accordance with a determination that the first value is higher than the second values, responding to the detected input ... in accordance with a determination that the first value is not higher than the second scores, forgoing responding to the detected input”*** – Dependent claims 2 and 3 recite these limitations. As explained, it is impossible for an “electronic device” to perform both of these method steps for a given voice input. *Supra* §III.C.2. Thus, the specification lacks enablement for these limitations for at least the reason that the recited sequence in a method claim is internally inconsistent and thus, inoperable.

b. Indefiniteness

By way of example, the following limitations of the Asserted Claims of the '311 Patent are indefinite:

- ***“the second scores”*** – Dependent claim 2 recites the term “the second scores,” which is indefinite for at least the reason that the term lacks antecedent basis.
- ***“in accordance with a determination that the first value is higher than the second values, responding to the detected input ... in accordance with a determination that the first value is not higher than the second scores, forgoing responding to the detected input”*** – Dependent claims 2 and 3 recite these limitations. As explained, it is impossible for an “electronic device” to perform both of these method steps for a given voice input. *Supra* §III.C.2. The internal inconsistency with this recited sequence renders dependent claims 2 and 3 indefinite.
- ***“the detected output”*** – Dependent claim 12 recites the term “the detected output,” which is indefinite for at least the reason that the term lacks antecedent basis.

2. '128 Patent

a. Lack of Enablement and/or Written Description

By way of example, the following limitations of the Asserted Claims of the '128 Patent lack enablement and/or written description:

- ***“in accordance with a determination that the quality score generated by the first electronic device is not the highest ..., forgoing outputting a response to the detected voice input”*** – Independent claims 1, 6, and 11 each recites this limitation. Google has interpreted this limitation such that the limitation can be met even after an electronic device makes the claimed “determination” but still outputs a visual and/or audible response to the detected voice input. However, the specification lacks enablement and written description support for this limitation under such an interpretation. Thus,

independent claims 1, 6, and 11 are invalid. Dependent claims 2-3, 5, 7-8, 10, 12-13, and 15 are invalid for at least the same reasons.

- ***“in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting ... in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting”*** – Independent claim 1 and dependent claims 2-3 and 5 recite these limitations. As explained, it is impossible for an “electronic device” to perform both of these method steps for a given voice input. *Supra* §III.C.2. Thus, the specification lacks enablement for these limitations for at least the reason that the recited sequence in a method claim is internally inconsistent and thus, inoperable.

b. Indefiniteness

By way of example, the following limitations of the Asserted Claims of the '128 Patent are indefinite:

- ***“in accordance with a determination that the quality score generated by the first electronic device is not the highest ..., forgoing outputting a response to the detected voice input”*** – Independent claims 1, 6, and 11 each recites this limitation. Google has interpreted this limitation such that the limitation can be met even after an electronic device makes the claimed “determination” but still outputs a visual and/or audible response to the detected voice input. Such an interpretation renders all Asserted Claims indefinite since a party is unable to ascertain when it may or may not infringe the claim.
- ***“in accordance with a determination that the quality score generated by the first electronic device is the highest amongst the generated quality score and received quality scores for the voice input, outputting ... in accordance with a determination that the quality score generated by the first electronic device is not the highest amongst the quality scores for the voice input generated by the plurality of electronic devices, forgoing outputting”*** – Independent claim 1 and dependent claims 2-3 and 5 recite these limitations. As explained, it is impossible for an “electronic device” to perform both of these method steps for a given voice input. *Supra* §III.C.2. The internal inconsistency with this recited sequence renders claims 1-3 and 5 indefinite.
- ***“cause the electronic device”*** – Independent claim 11 recites the term “the electronic device,” which is indefinite for at least the reason that the term lacks antecedent basis. For instance, it unclear whether “the electronic device” refers back to “a first electronic device” or “one of the plurality of electronic devices” or is meant to instead recite “the electronics devices” recited in independent claim 11. For at least this reason, dependent claims 12-13 and 15 are also indefinite.

- ***“in response to receiving the request, receiving one or more network configuration packets from the wireless device via the short-range signal transmission path”*** – Independent claims 1, 11, and 16 recite this limitation. The specification lacks enablement and written description support for ***“in response to receiving the request, receiving one or more network configuration packets*** from the wireless device via the short-range signal transmission path.” For at least these reasons, the Asserted Claims of the ’615 Patent are invalid.
- ***“wherein the device configuration data includes at least one of an Internet Protocol address, proxy, and gateway configuration”*** – Dependent claim 8 recites this limitation. The specification lacks enablement and written description support for “wherein the device configuration data includes at ***least one of an Internet Protocol address, proxy, and gateway configuration.***” For at least these reasons, claim 8 is invalid.

7. Indefiniteness

By way of example, the following limitations of the Asserted Claims of the ’615 Patent are indefinite:

- ***“short range signal transmission path”*** – Independent claims 1, 11, and 16 recite this limitation, which is indefinite because one of ordinary skill in the art would not understand what is “short range” and what is not, and the specification does not provide any concrete guidance.
- ***“in response to receiving the request, receiving one or more network configuration packets”*** – Independent claims 1, 11, and 16 recite this limitation, which is indefinite because one of ordinary skill in the art would not understand how a device receives configuration packets *in response to* receiving the claimed request, and the specification does not provide any guidance.

V. CONCLUSION

For the reasons discussed above, and those revealed through further discovery, the Asserted Claims of the Asserted Patents are invalid.

Dated: December 7, 2022

Respectfully submitted,

/s/ Bas de Blank
 Jordan L. Coyle, Esq.
 ORRICK, HERRINGTON & SUTCLIFFE LLP
 Columbia Center
 1152 15th Street, NW
 Washington, DC 20005

Tel.: (202) 339-8400
Email: Sonos-1330-service@orrick.com

Bas de Blank, Esq.
ORRICK, HERRINGTON & SUTCLIFFE LLP
1000 Marsh Road
Menlo Park, CA 94025-1015
Tel.: (650) 614-7400
Email: Sonos-1330-service@orrick.com

Clement S. Roberts, Esq.
ORRICK, HERRINGTON & SUTCLIFFE LLP
The Orrick Building
405 Howard Street
San Francisco, CA 94105-2669
Tel: (415) 773-5700
Fax: (415) 773-5759
Email: Sonos-1330-service@orrick.com

Alyssa M. Caridis, Esq.
Geoffrey Moss, Esq.
ORRICK, HERRINGTON & SUTCLIFFE LLP
777 South Figueroa Street, Suite 3200
Los Angeles, CA 90017-5855
Tel: (213) 629-2020
Fax: (213) 612-2499
Email: Sonos-1330-service@orrick.com

George I. Lee, Esq.
Sean M. Sullivan, Esq.
Rory P. Shea, Esq.
J. Dan Smith, Esq.
Cole B. Richter, Esq.
Michael P. Boyea, Esq.
Jae Y. Pak, Esq.
LEE SULLIVAN SHEA & SMITH LLP
656 W Randolph St, Floor 5W
Chicago, IL 60661
Tel: (312) 754-0002
Fax: (312) 754-0003
Email: Sonos-1330-service@orrick.com

Counsel for Respondent Sonos, Inc.

Exhibit 11

Automatic Speech and Speaker Recognition

Large Margin and Kernel Methods

Joseph Keshet

IDIAP Research Institute, Martigny, Switzerland

Samy Bengio

Google Inc., Mountain View, CA, USA



A John Wiley and Sons, Ltd, Publication

This edition first published 2009
© 2009 John Wiley & Sons Ltd

Registered office

John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ,
United Kingdom

For details of our global editorial offices, for customer services and for information about how to apply for permission to reuse the copyright material in this book please see our website at www.wiley.com.

The right of the author to be identified as the author of this work has been asserted in accordance with the Copyright, Designs and Patents Act 1988.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by the UK Copyright, Designs and Patents Act 1988, without the prior permission of the publisher.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Designations used by companies to distinguish their products are often claimed as trademarks. All brand names and product names used in this book are trade names, service marks, trademarks or registered trademarks of their respective owners. The publisher is not associated with any product or vendor mentioned in this book. This publication is designed to provide accurate and authoritative information in regard to the subject matter covered. It is sold on the understanding that the publisher is not engaged in rendering professional services. If professional advice or other expert assistance is required, the services of a competent professional should be sought.

Library of Congress Cataloging-in-Publication Data

Automatic speech and speaker recognition : large margin and kernel methods /
edited by Joseph Keshet, Samy Bengio.
p. cm.

Includes bibliographical references and index.

ISBN 978-0-470-69683-5 (cloth)

1. Automatic speech recognition. I. Keshet, Joseph. II. Bengio, Samy.

TK7895.S65A983 2009

006.4'54-dc22

2008038551

A catalogue record for this book is available from the British Library.

ISBN 9780470696835 (H/B)

Set in 10/12pt Times by Sunrise Setting Ltd, Torquay, UK.
Printed in Singapore by Markono Print Media Pte Ltd.

1

Introduction

Samy Bengio and Joseph Keshet

One of the most natural communication tools used by humans is their voice. It is hence natural that a lot of research has been devoted to analyzing and understanding human uttered speech for various applications. The most obvious one is *automatic speech recognition*, where the goal is to transcribe a recorded speech utterance into its corresponding sequence of words. Other applications include *speaker recognition*, where the goal is to determine either the claimed identity of the speaker (verification) or who is speaking (identification), and speaker segmentation or diarization, where the goal is to segment an acoustic sequence in terms of the underlying speakers (such as during a dialog).

Although an enormous amount of research has been devoted to speech processing, there appears to be some form of local optimum in terms of the fundamental tools used to approach these problems. The aim of this book is to introduce the speech researcher community to radically different approaches based on more recent kernel based machine learning methods. In this introduction, we first briefly review the predominant speech processing approach, based on hidden Markov models, as well as its known problems; we then introduce the most well known kernel based approach, the Support Vector Machine (SVM), and finally outline the various contributions of this book.

1.1 The Traditional Approach to Speech Processing

Most speech processing problems, including speech recognition, speaker verification, speaker segmentation, etc., proceed with basically the same general approach, which is described here in the context of speech recognition, as this is the field that has attracted most of the research in the last 40 years. The approach is based on the following statistical framework.

A sequence of acoustic feature vectors is extracted from a spoken utterance by a front-end signal processor. We denote the sequence of acoustic feature vectors by $\bar{\mathbf{x}} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T)$,

Automatic Speech and Speaker Recognition: Large Margin and Kernel Methods Joseph Keshet and Samy Bengio
© 2009 John Wiley & Sons, Ltd

where $\mathbf{x}_t \in \mathcal{X}$ and $\mathcal{X} \subset \mathbb{R}^d$ is the domain of the acoustic vectors. Each vector is a compact representation of the short-time spectrum. Typically, each vector covers a period of 10 ms and there are approximately $T = 300$ acoustic vectors in a 10 word utterance. The spoken utterance consists of a sequence of words $\bar{v} = (v_1, \dots, v_N)$. Each of the words belongs to a fixed and known vocabulary $\mathcal{V}$, that is, $v_i \in \mathcal{V}$. The task of the speech recognizer is to predict the most probable word sequence $\bar{v}'$ given the acoustic signal $\bar{\mathbf{x}}$. Speech recognition is formulated as a *Maximum a Posteriori* (MAP) decoding problem as follows:

$$\bar{v}' = \arg \max_{\bar{v}} P(\bar{v}|\bar{\mathbf{x}}) = \arg \max_{\bar{v}} \frac{p(\bar{\mathbf{x}}|\bar{v})P(\bar{v})}{p(\bar{\mathbf{x}})}, \quad (1.1)$$

where we used Bayes' rule to decompose the posterior probability in Equation (1.1). The term $p(\bar{\mathbf{x}}|\bar{v})$ is the probability of observing the acoustic vector sequence $\bar{\mathbf{x}}$ given a specified word sequence $\bar{v}$ and it is known as *the acoustic model*. The term $P(\bar{v})$ is the probability of observing a word sequence $\bar{v}$ and it is known as *the language model*. The term $p(\bar{\mathbf{x}})$ can be disregarded, since it is constant under the max operation.

The acoustic model is usually estimated by a Hidden Markov Model (HMM) (Rabiner and Juang 1993), a kind of graphical model (Jordan 1999) that represents the joint probability of an observed variable and a hidden (or latent) variable. In order to understand the acoustic model, we now describe the basic HMM decoding process. By decoding we mean the calculation of the $\arg \max_{\bar{v}}$ in Equation (1.1). The process starts with an assumed word sequence $\bar{v}$. Each word in this sequence is converted into a sequence of basic spoken units called *phones*¹ using a pronunciation dictionary. Each phone is represented by a single HMM, where the HMM is a probabilistic state machine typically composed of three states (which are the hidden or latent variables) in a left-to-right topology. Assume that $\mathcal{Q}$ is the set of all states, and let $\bar{q}$ be a sequence of states, that is $\bar{q} = (q_1, q_2, \dots, q_T)$, where it is assumed there exists some latent random variable $q_t \in \mathcal{Q}$ for each frame $\mathbf{x}_t$ of $\bar{\mathbf{x}}$. Wrapping up, the sequence of words $\bar{v}$ is converted into a sequence of phones $\bar{p}$ using a pronunciation dictionary, and the sequence of phones is converted to a sequence of states, with in general at least three states per phone. The goal now is to find the most probable sequence of states.

Formally, the HMM is defined as a pair of random processes $\bar{q}$ and $\bar{\mathbf{x}}$, where the following first order Markov assumptions are made:

1. $P(q_t|q_1, q_2, \dots, q_{t-1}) = P(q_t|q_{t-1})$;
2. $p(\mathbf{x}_t|\mathbf{x}_1, \dots, \mathbf{x}_{t-1}, \mathbf{x}_{t+1}, \dots, \mathbf{x}_T, q_1, \dots, q_T) = p(\mathbf{x}_t|q_t)$.

The HMM is a *generative model* and can be thought of as a generator of acoustic vector sequences. During each time unit (frame), the model can change a state with probability $P(q_t|q_{t-1})$, also known as the *transition probability*. Then, at every time step, an acoustic vector is emitted with probability $p(\mathbf{x}_t|q_t)$, sometimes referred to as the *emission probability*. In practice the sequence of states is not observable; hence the model is called hidden. The probability of the state sequence $\bar{q}$ given the observation sequence $\bar{\mathbf{x}}$ can be found using Bayes' rule as follows:

$$P(\bar{q}|\bar{\mathbf{x}}) = \frac{p(\bar{\mathbf{x}}, \bar{q})}{p(\bar{\mathbf{x}})},$$

¹A *phone* is a consonant or vowel speech sound. A *phoneme* is any equivalent set of phones which leaves a word meaning invariant (Allen 2005).

where the joint probability of a vector sequence $\bar{\mathbf{x}}$ and a state sequence $\bar{q}$ is calculated simply as a product of the transition probabilities and the output probabilities:

$$p(\bar{\mathbf{x}}, \bar{q}) = P(q_0) \prod_{t=1}^T P(q_t|q_{t-1}) p(\mathbf{x}_t|q_t), \quad (1.2)$$

where we assumed that q_0 is constrained to be a non-emitting initial state. The emission density distributions $p(\mathbf{x}_t|q_t)$ are often estimated using diagonal covariance Gaussian Mixture Models (GMMs) for each state q_t , which model the density of a d -dimensional vector $\mathbf{x}$ as follows:

$$p(\mathbf{x}) = \sum_i w_i \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i), \quad (1.3)$$

where $w_i \in \mathbb{R}$ is positive with $\sum_i w_i = 1$, and $\mathcal{N}(\cdot; \boldsymbol{\mu}, \boldsymbol{\sigma})$ is a Gaussian with mean $\boldsymbol{\mu}_i \in \mathbb{R}^d$ and standard deviation $\boldsymbol{\sigma}_i \in \mathbb{R}^d$. Given the HMM parameters in the form of the transition probability and emission probability (as GMMs), the problem of finding the most probable state sequence is solved by maximizing $p(\bar{\mathbf{x}}, \bar{q})$ over all possible state sequences using the *Viterbi algorithm* (Rabiner and Juang 1993).

In the training phase, the model parameters are estimated. Assume one has access to a training set of m examples $\mathcal{T}_{\text{train}} = \{(\bar{\mathbf{x}}_i, \bar{v}_i)\}_{i=1}^m$. Training of the acoustic model and the language model can be done in two separate steps. The acoustic model parameters include the transition probabilities and the emission probabilities, and they are estimated by a procedure known as the *Baum–Welch algorithm* (Baum *et al.* 1970), which is a special case of the Expectation-Maximization (EM) algorithm, when applied to HMMs. This algorithm provides a very efficient procedure to estimate these probabilities iteratively. The parameters of the HMMs are chosen to maximize the probability of the acoustic vector sequence $p(\bar{\mathbf{x}})$ given a virtual HMM composed as the concatenation of the phone HMMs that correspond to the underlying sequence of words $\bar{v}$. The Baum–Welch algorithm monotonically converges in polynomial time (with respect to the number of states and the length of the acoustic sequences) to local stationary points of the likelihood function.

Language models are used to estimate the probability of a given sequence of words, $P(\bar{v})$. The language model is often estimated by n -grams (Manning and Schutze 1999), where the probability of a sequence of N words $(\bar{v}_1, \bar{v}_2, \dots, \bar{v}_N)$ is estimated as follows:

$$p(\bar{v}) \approx \prod_t p(v_t|v_{t-1}, v_{t-2}, \dots, v_{t-N}), \quad (1.4)$$

where each term can be estimated on a large corpus of written documents by simply counting the occurrences of each n -gram. Various smoothing and back-off strategies have been developed in the case of large n where most n -grams would be poorly estimated even using very large text corpora.

1.2 Potential Problems of the Probabilistic Approach

Although most state-of-the-art approaches to speech recognition are based on the use of HMMs and GMMs, also called Continuous Density HMMs (CD-HMMs), they have several drawbacks, some of which we discuss hereafter.

- Consider the logarithmic form of Equation (1.2),

$$\log p(\bar{\mathbf{x}}, \bar{q}) = \log P(q_0) + \sum_{t=1}^T \log P(q_t | q_{t-1}) + \sum_{t=1}^T \log p(\mathbf{x}_t | q_t). \quad (1.5)$$

There is a known structural problem when mixing densities $p(\mathbf{x}_t | q_t)$ and probabilities $P(q_t | q_{t-1})$: the global likelihood is mostly influenced by the emission distributions and almost not at all by the transition probabilities, hence temporal aspects are poorly taken into account (Bourlard *et al.* 1996; Young 1996). This happens mainly because the variance of densities of the emission distribution depends on d , the actual dimension of the acoustic features: the higher d , the higher the expected variance of $p(\bar{\mathbf{x}} | \bar{q})$, while the variance of the transition distributions mainly depend on the number of states of the HMM. In practice, one can observe a ratio of about 100 between these variances; hence when selecting the best sequence of words for a given acoustic sequence, only the emission distributions are taken into account. Although the latter may well be very well estimated using GMMs, they do not take into account most temporal dependencies between them (which are supposed to be modeled by transitions).

- While the EM algorithm is very well known and efficiently implemented for HMMs, it can only converge to local optima, and hence optimization may greatly vary according to initial parameter settings. For CD-HMMs the Gaussian means and variances are often initialized using K-Means, which is itself also known to be very sensitive to initialization.
- Not only is EM known to be prone to local optimal, it is basically used to maximize the likelihood of the observed acoustic sequence, in the context of the expected sequence of words. Note however that the performance of most speech recognizers is estimated using other measures than the likelihood. In general, one is interested in minimizing the number of errors in the generated word sequence. This is often done by computing the Levenshtein distance between the expected and the obtained word sequences, and is often known as the *word error rate*. There might be a significant difference between the best HMM models according to the maximum likelihood criterion and the word error rate criterion.

Hence, throughout the years, various alternatives have been proposed. One line of research has been centered around proposing more discriminative training algorithms for HMMs. That includes Maximum Mutual Information Estimation (MMIE) (Bahl *et al.* 1986), Minimum Classification Error (MCE) (Juang and Katagiri 1992), Minimum Phone Error (MPE) and Minimum Word Error (MWE) (Povey and Woodland 2002). All these approaches, although proposing better training criteria, still suffer from most of the drawbacks described earlier (local minima, useless transitions).

The last 15 years of research in the machine learning community has welcomed the introduction of so-called large margin and kernel approaches, of which the SVM is its best known example. An important role of this book is to show how these recent efforts from the machine learning community can be used to improve research in the speech processing domain. Hence, the next section is devoted to a brief introduction to SVMs.

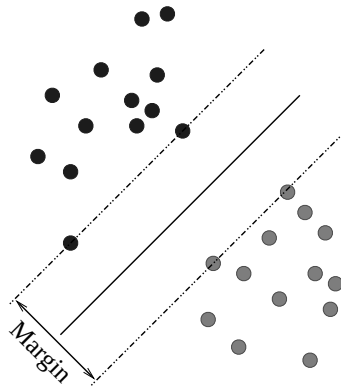


Figure 1.1 Illustration of the notion of margin.

1.3 Support Vector Machines for Binary Classification

The most well known kernel based machine learning approach is the SVM (Vapnik 1998). While it was not developed in particular for speech processing, most of the chapters in this book propose kernel methods that are in one way or another inspired by the SVM.

Let us assume we are given a training set of examples $\mathcal{T}_{\text{train}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ where $\mathbf{x}_i \in \mathbb{R}^d$ is a d -dimensional input vector and $y_i \in \{-1, 1\}$ is the target class. The simplest binary classifier one can think of is the linear classifier, where we are looking for parameters $(\mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R})$ such that

$$\hat{y}(\mathbf{x}) = \text{sign}(\mathbf{w} \cdot \mathbf{x} + b). \quad (1.6)$$

When the training set is said to be linearly separable, there is potentially an infinite number of solutions $(\mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R})$ that satisfy (1.6). Hence, the SVM approach looks for the one that maximizes the *margin* between the two classes, where the margin can be defined as the sum of the smallest distances between the separating hyper-plane and points of each class. This concept is illustrated in Figure 1.1.

This can be expressed by the following optimization problem:

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{subject to } & \forall i \quad y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1. \end{aligned} \quad (1.7)$$

While this is difficult to solve, its following dual formulation is computationally more efficient:

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m y_i y_j \alpha_i \alpha_j \mathbf{x}_i \cdot \mathbf{x}_j \\ \text{subject to } & \begin{cases} \forall i \quad \alpha_i \geq 0 \\ \sum_{i=1}^m \alpha_i y_i = 0. \end{cases} \end{aligned} \quad (1.8)$$

One problem with this formulation is that if the problem is not linearly separable, there might be no solution to it. Hence one can relax the constraints by allowing errors with an additional hyper-parameter C that controls the trade-off between maximizing the margin and minimizing the number of training errors (Cortes and Vapnik 1995), as follows:

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i \\ \text{subject to} \quad & \begin{cases} \forall i & y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i \\ \forall i & \xi_i \geq 0 \end{cases} \end{aligned} \quad (1.9)$$

which has dual formulation

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^n y_i y_j \alpha_i \alpha_j \mathbf{x}_i \cdot \mathbf{x}_j \\ \text{subject to} \quad & \begin{cases} \forall i & 0 \leq \alpha_i \leq C \\ \sum_{i=1}^m \alpha_i y_i = 0. \end{cases} \end{aligned} \quad (1.10)$$

In order to look for nonlinear solutions, one can easily replace $\mathbf{x}$ by some nonlinear function $\phi(\mathbf{x})$. It is interesting to note that $\mathbf{x}$ only appears in dot products in (1.10). It has thus been proposed to replace all occurrences of $\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$ by some kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$. As long as $k(\cdot, \cdot)$ lives in a reproducing kernel Hilbert space (RKHS), one can guarantee that there exists some function $\phi(\cdot)$ such that

$$k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j).$$

Thus, even if $\phi(\mathbf{x})$ projects $\mathbf{x}$ in a very high (possibly infinite) dimensional space, $k(\mathbf{x}_i, \mathbf{x}_j)$ can still be efficiently computed.

Problem (1.10) can be solved using off-the-shelf quadratic optimization tools. Note however that the underlying computational complexity is at least quadratic in the number of training examples, which can often be a serious limit for most speech processing applications.

After solving (1.10), the resulting SVM solution takes the form of

$$\hat{y}(\mathbf{x}) = \text{sign} \left(\sum_{i=1}^m y_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b \right) \quad (1.11)$$

where most α_i are zero except those corresponding to examples in the margin or misclassified, often called *support vectors* (hence the name of SVMs).

1.4 Outline

The book has four parts. The first part, *Foundations*, covers important aspects of extending the binary SVM to speech and speaker recognition applications. Chapter 2 provides a detailed review of efficient and practical solutions to large scale convex optimization problems one encounters when using large margin and kernel methods with the enormous datasets used

in speech applications. Chapter 3 presents an extension of the binary SVM to multiclass, hierarchical and categorical classification. Specifically, the chapter presents a more complex setting in which the possible labels or categories are many and organized.

The second part, *Acoustic Modeling*, deals with large margin and kernel method algorithms for sequence prediction required for acoustic modeling. Chapter 4 presents a large margin algorithm for forced alignment of a phoneme sequence to a corresponding speech signal, that is, proper positioning of a sequence of phonemes in relation to a corresponding continuous speech signal. Chapter 5 describes a kernel wrapper for the task of phoneme recognition, which is based on the Gaussian kernel. This chapter also presents a kernel based iterative algorithm aimed at minimizing the Levenshtein distance between the predicted phoneme sequence and the true one. Chapter 6 reviews the use of dynamic kernels for acoustic models and especially describes the Augmented Statistical Models, resulting from the generative kernel, a generalization of the Fisher kernel. Chapter 7 investigates a framework for large margin parameter estimation for continuous-density HMMs.

The third part of the book is devoted to *Language Modeling*. Chapter 8 reviews past and present work on discriminative training of language models, and focuses on three key issues: training data, learning algorithms, features. Chapter 9 describes different large margin algorithms for the application of part-of-speech tagging. Chapter 10 presents a proposal for large vocabulary continuous speech recognition, which is solely based on large margin and kernel methods, incorporating the acoustic models described in Part II and the discriminative language models.

The last part is dedicated to *Applications*. Chapter 11 covers a discriminative keyword spotting algorithm, based on a large margin approach, which aims at maximizing the area under the receiver operating characteristic (ROC) curve, the most common measure to evaluate keyword spotters. Chapter 12 surveys recent work on the use of kernel approaches to text-independent speaker verification. Finally, Chapter 13 introduces the main concepts and algorithms together with recent advances in learning a similarity matrix from data. The techniques in the chapter are illustrated on the blind one-microphone speech separation problem, by casting the problem as one of segmentation of the spectrogram.

References

- Allen JB 2005 *Articulation and Intelligibility*. Morgan & Claypool.
- Bahl LR, Brown PF, de Souza PV and Mercer RL 1986 Maximum mutual information of hidden Markov model parameters for speech recognition. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pp. 49–53.
- Baum LE, Petrie T, Soules G and Weiss N 1970 A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics* **41**(1), 164–171.
- Bourlard H, Hermansky H and Morgan N 1996 Towards increasing speech recognition error rates. *Speech Communication* **18**, 205–231.
- Cortes C and Vapnik V 1995 Support-vector networks. *Machine Learning* **20**(3), 273–297.
- Jordan MI (ed.) 1999 *Learning in Graphical Models*. MIT Press.
- Juang BH and Katagiri S 1992 Discriminative learning for minimum error classification. *IEEE Transactions on Signal Processing* **40**(12), 3043–3054.
- Manning CD and Schutze H 1999 *Foundations of Statistical Natural Language Processing*. MIT Press.

- Povey D and Woodland PC 2002 Minimum phone error and I-smoothing for improved discriminative training. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*.
- Rabiner L and Juang BH 1993 *Fundamentals of speech recognition*, 1st edn. Prentice Hall.
- Vapnik VN 1998 *Statistical Learning Theory*. John Wiley & Sons.
- Young S 1996 A review of large-vocabulary continuous speech recognition. *IEEE Signal Processing Mag.* pp. 45–57.

Discriminative Keyword Spotting

David Grangier, Joseph Keshet and Samy Bengio

This chapter introduces a discriminative method for detecting and spotting keywords in spoken utterances. Given a word represented as a sequence of phonemes and a spoken utterance, the keyword spotter predicts the best time span of the phoneme sequence in the spoken utterance along with a confidence. If the prediction confidence is above certain level the keyword is declared to be spoken in the utterance within the predicted time span, otherwise the keyword is declared as not spoken. The problem of keyword spotting training is formulated as a discriminative task where the model parameters are chosen so the utterance in which the keyword is spoken would have higher confidence than any other spoken utterance in which the keyword is not spoken. It is shown theoretically and empirically that the proposed training method resulted with a high area under the Receiver Operating Characteristic (ROC) curve, the most common measure to evaluate keyword spotters. We present an iterative algorithm to train the keyword spotter efficiently. The proposed approach contrasts with standard spotting strategies based on Hidden Markov Models (HMMs), for which the training procedure does not maximize a loss directly related to the spotting performance. Several experiments performed on TIMIT and WSJ corpora show the advantage of our approach over HMM-based alternatives.

11.1 Introduction

Keyword spotting aims at detecting any given keyword in spoken utterances. This task is important in numerous applications, such as voice mail retrieval, voice command detection and spoken term detection and retrieval. Previous work has focused mainly on several variants of HMMs to address this intrinsically sequential problem. While the HMM-based

approaches constitute the state-of-the-art, they suffer from several known limitations. Most of these limitations are not specific to the keyword spotting problem, and are common to other tasks such as speech recognition, as pointed out in Chapter 1. For instance, the predominance of the emission probabilities in the likelihood, which tends to neglect duration and transition models, or the Expectation Maximization (EM) training procedure, which is prone to convergence to local optima. Other drawbacks are specific to the application of HMMs to the keyword spotting task. In particular, the scarce occurrence of some keywords in the training corpora often requires ad-hoc modifications of the HMM topology, the transition probabilities or the decoding algorithm. The most acute limitation of HMM-based approaches lies in their training objective. Typically, HMM training aims at maximizing the likelihood of transcribed utterances, and does not provide any guarantees in terms of keyword spotting performance.

The performance of a keyword spotting system is often measured by the ROC curve, that is, a plot of the true positive (spotting a keyword correctly) rate as a function of the false positive (mis-spotting a keyword) rate; see for example Benayed *et al.* (2004), Ketabdar *et al.* (2006), Silaghi and Bourlard (1999). Each point on the ROC curve represents the system performance for a specific trade-off between achieving a high true positive rate and a low false positive rate. Since the preferred trade-off is not always defined in advance, systems are commonly evaluated according to the averaged performance over all operating points. This corresponds to preferring the systems that attain the highest Area Under the ROC Curve (AUC).

In this study, we devise a discriminative large margin approach for learning to spot any given keyword in any given speech utterance. The keyword spotting function gets as input a phoneme sequence representing the keyword and a spoken utterance and outputs a prediction of the time span of the keyword in the spoken utterance and a confidence. If the confidence is above some predefined threshold, the keyword is declared to be spoken in the predicted time span, otherwise the keyword is declared as not spoken. The goal of the training algorithm is to maximize the AUC on the training data and on unseen test data. We call an utterance in the training set in which the keyword is spoken a *positive utterance*, and respectively, an utterance in which the keyword is not spoken a *negative utterance*. Using the Wilcoxon–Mann–Whitney statistics (Cortes and Mohri 2004), we formulate the training as a problem of estimating the model parameters such that the confidence of the correct time span in a positive utterance would be higher than the confidence of any time span in any negative utterance. Formally this problem is stated as a convex optimization problem with constraints. The solution to this optimization problem is a function which is shown analytically to attain high AUC on the training set and is likely to have good generalization properties on unseen test data as well. Moreover, comparing with HMMs, our approach is based on a convex optimization procedure, which converges to the global optima, and it is based on a non-probabilistic framework, which offers greater flexibility in selecting the relative importance of duration modeling with respect to acoustic modeling.

The remainder of this chapter is organized as follows: Section 11.2 describes previous work on keyword spotting, Section 11.3 introduces our discriminative large margin approach, Section 11.4 presents different experiments comparing the proposed model with an HMM-based solution. Finally, Section 11.5 draws some conclusions and delineates possible directions for future research.

11.2 Previous Work

The research on keyword spotting has paralleled the development of the automatic speech recognition (ASR) domain in the last thirty years. Like ASR, keyword spotting has first been addressed with models based on Dynamic Time Warping (DTW) (Bridle 1973; Higgins and Wohlford 1985). Then, approaches based on discrete HMMs were introduced (Kawabata *et al.* 1988). Finally, discrete HMMs have been replaced by continuous HMMs (Rabiner and Juang 1993).

The core objective of a keyword spotting system is to discriminate between utterances in which a given keyword is uttered and utterances in which the keyword is not uttered. For this purpose, the first approaches based on DTW proposed to compute the alignment distance between a template utterance representing the target keyword and all possible segments of the test signal (Bridle 1973). In this context, the keyword is considered as detected in a segment of the test utterance whenever the alignment distance is below some predefined threshold. Such approaches are, however, greatly affected by speaker mismatch and varying recording conditions between the template sequence and the test signal. To gain some robustness, it has then been proposed to compute alignment distances not only with respect to the target keyword template, but also with respect to other keyword templates (Higgins and Wohlford 1985). Precisely, given a test utterance, the system identifies the concatenation of templates with the lowest distance to the signal and the keyword is considered as detected if this concatenation contains the target keyword template. Therefore, the keyword alignment distance is not considered as an absolute number, but relatively to the distances to other templates, which increase robustness with respect to changes in the recording conditions.

Along with the development of the speech research, increasingly large amounts of labeled speech data were collected, and DTW-based techniques started showing their shortcomings to leverage from large training sets. In particular, large corpora contain thousands of words and dozens of instances for each word. Considering each instance as a template makes the search for the best template concatenation a prohibitively expensive task. Consequently, discrete HMMs were introduced for ASR (Bahl *et al.* 1986), and then for keyword spotting (Kawabata *et al.* 1988; Wilpon *et al.* 1990). A discrete HMM assumes that the quantized acoustic feature vectors representing the input utterance are independent conditioned on the hidden state variables. This type of model introduces several advantages compared to DTW-based approaches, including an improved robustness to speaker and channel changes, when several training utterances of the targeted keyword are available. However, the most important evolution introduced with the HMM certainly lies in the development of phone or triphone-based modeling (Kawabata *et al.* 1988; Lee and Hon 1988; Rose and Paul 1990), in which a word model is composed of several sub-unit models shared across words. This means that the model of a given word not only benefits from the training utterances containing this word, but also from all the utterances containing its sub-units. Additionally, great scalability is achieved with such an approach since the number of acoustic parameters is proportional to the number of sub-units and does not grow linearly with the corpus size, as opposed to template-based approaches. A further advantage of phone-based modeling is the ability to spot words unavailable at training time, as this paradigm allows one to build a new word model by composing already trained sub-unit models. This aspect is very important, since in most applications the set of test keywords is not known in advance.

Soon after the application of discrete HMMs to speech problems, continuous density HMMs have been introduced in the ASR community (Rabiner and Juang 1993). Continuous HMMs eliminate the need of acoustic vector quantization, as the distributions associated with the HMM states are continuous densities, often modeled by Gaussian Mixture Models (GMMs). The learning of both GMM parameters and the state transition probabilities is performed in a single integrated framework, maximizing the likelihood of the training data given its transcription through the EM algorithm (Bilmes 1998). This approach has been shown to be more effective and allows greater flexibility for speaker or channel adaptation (Rabiner and Juang 1993). It is now the most widely used approach for both ASR and keyword spotting.

In the context of keyword spotting, different strategies based on continuous HMMs have been proposed. In most cases, a sub-unit based HMM is trained over a large corpus of transcribed data and a new model is then built from the sub-unit models. Such a model is composed of two parts, a keyword HMM and a *filler* or *garbage* HMM, which respectively model the keyword and non-keyword parts of the signal. This topology is depicted in Figure 11.1. Given such a model, keyword detection is performed by searching for the sequence of states that yields the highest likelihood for the provided test sequence through Viterbi decoding. Keyword detection is determined by checking whether the Viterbi best-path passes through the keyword model or not. In such a model, the selection of the transition probabilities in the keyword sets the trade-off between low false alarm rate (detecting a keyword when it is not presented), and low false rejection rate (not detecting a keyword when it is indeed presented). Another important aspect of this approach lies in the modeling of non-keyword parts of the signal, and several choices are possible for the garbage HMM. The simplest choice models garbage with an HMM that fully connects all sub-unit models (Rose and Paul 1990), while the most complex choice models garbage with a full-large vocabulary HMM, where the lexicon excludes the keyword (Weintraub 1993). The latter approach obviously yields a better garbage model, using additional linguistic knowledge. This advantage, however, induces a higher decoding cost and requires a larger amount of training data, in particular for language model training. Besides practical concerns, one can conceptually wonder whether an automatic spotting approach should require such a large linguistic knowledge. Of course, several variations of garbage models exist between the two extreme examples pointed out above (see for instance Boite *et al.* (1993)).

Viterbi decoding relies on a sequence of local decisions to determine the best path, which can be fragile with respect to local model mismatch. In the context of HMM-based keyword spotting, a keyword can be missed if only its first phoneme suffers such a mismatch, for instance. To gain some robustness, likelihood ratio approaches have been proposed (Rose and Paul 1990; Weintraub 1995). In this case, the confidence score output from the keyword spotter corresponds to the ratio between the likelihood estimated by an HMM including the occurrence of the target keyword, and the likelihood estimated by an HMM excluding it. These HMM topologies are depicted in Figure 11.2. Detection is then performed by comparing the output scores to a predefined threshold. Different variations on this likelihood ratio approach have then been devised, such as computing the ratio only on the part of the signal where the keyword is assumed to be detected (Junkawitsch *et al.* 1997). Overall, all the above methods are variations over the same HMM paradigm, which consists in training a generative model through likelihood maximization, before introducing different modifications prior to decoding in order to address the keyword spotting task. In other

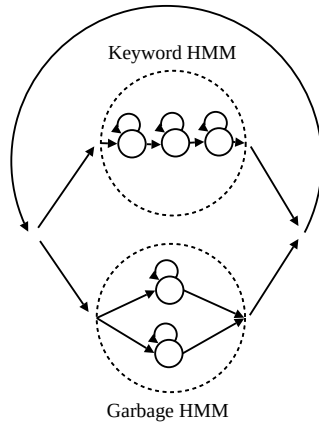


Figure 11.1 HMM topology for keyword spotting with a Viterbi best-path strategy. This approach verifies whether the Viterbi best-path passes through the keyword sub-model.

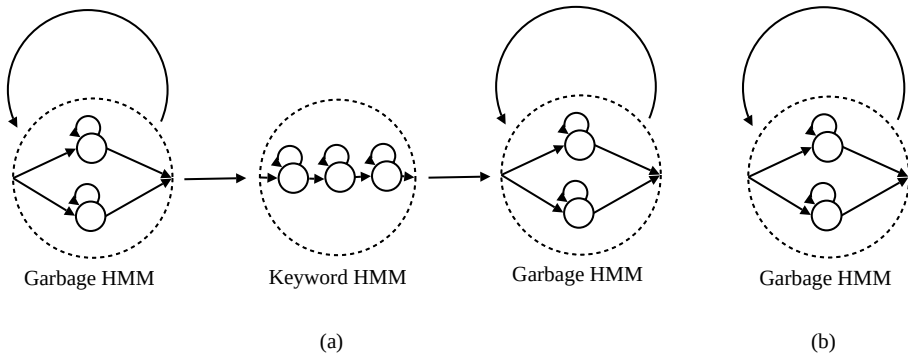


Figure 11.2 HMM topology for keyword spotting with a likelihood ratio strategy. This approach compares the likelihood of the sequence given that the keyword is uttered (a) with the likelihood of the sequence given that the keyword is not uttered (b).

words, these approaches do not propose to train the model so as to maximize the spotting performance, and the keyword spotting task is only introduced in the inference step after training.

Only few studies have proposed discriminative parameter training approaches to circumvent this weakness (Benayed *et al.* 2003; Sandness and Hetherington 2000; Sukkar *et al.* 1996; Weintraub *et al.* 1997). Sukkar *et al.* (1996) proposed to maximize the likelihood ratio between the keyword and garbage models for keyword utterances and to minimize it over a set of false alarms generated by a first keyword spotter. Sandness and Hetherington (2000) proposed to apply Minimum Classification Error (MCE) to the keyword spotting problem.

The training procedure updates the acoustic models to lower the score of non-keyword models in the part of the signal where the keyword is uttered. However, this procedure does not focus on false alarms, and does not aim at lowering the score of the keyword models in parts of the signal where the keyword is not uttered. Other discriminative approaches have been focused on combining different HMM-based keyword detectors. For instance, Weintraub *et al.* (1997) trained a neural network to combine likelihood ratios from different models. Benayed *et al.* (2003) relied on Support Vector Machines (SVMs) to combine different averages of phone-level likelihoods. Both of these approaches propose to minimize the error rate, which equally weights the two possible spotting errors, false positive (or false alarm) and false negative (missed keyword occurrence, often called keyword deletion). This measure is, however, barely used to evaluate keyword spotters, due to the *unbalanced* nature of the problem. Precisely, the targeted keywords generally occur rarely and hence the number of potential false alarms highly exceeds the number of potential missed detections. In this case, the useless model which never predicts the keyword avoids all false alarms and yields a very low error rate, with which it is difficult to compete. For that reason the AUC is more informative and is commonly used to evaluate models. Attaining high AUC would hence be an appropriate learning objective for the discriminative training of a keyword spotter. To the best of our knowledge, only Chang (1995) proposed an approach targeting this goal. This work introduces a methodology to maximize the figure-of-merit (FOM), which corresponds to the AUC over a specific range of false alarm rates. However, the proposed approach relies on various heuristics, such as gradient smoothing and sorting approximations, which does not ensure any theoretical guarantee on obtaining high FOM. Also, these heuristics involve the selection of several hyperparameters, which challenges a practical use.

In the following, we introduce a model that aims at achieving high AUC over a set of training examples, and constitutes a truly discriminative approach to the keyword spotting problem. The proposed model relies on large margin learning techniques for sequence prediction and provides theoretical guarantees regarding the generalization performance. Furthermore, its efficient learning procedure ensures scalability toward large problems and simple practical use.

11.3 Discriminative Keyword Spotting

This section formalizes the keyword spotting problem, and introduces the proposed approach. First, we describe the problem of keyword spotting formally. This allows us to introduce a loss derived from the definition of the AUC. Then, we present our model parameterization and the training procedure to minimize efficiently a regularized version of this loss. Finally, we give an analysis of the iterative algorithm, and show that it achieves a high cumulative AUC in the training process and high expected AUC on unseen test data.

11.3.1 Problem Setting

In the keyword spotting task, we are provided with a speech signal composed of a sequence of acoustic feature vectors $\tilde{\mathbf{x}} = (\mathbf{x}_1, \dots, \mathbf{x}_T)$, where $\mathbf{x}_t \in \mathcal{X} \subset \mathbb{R}^d$, for all $1 \leq t \leq T$, is a feature vector of length d extracted from the t th frame. Naturally, the length of the acoustic signal varies from one signal to another and thus T is not fixed. We denote a keyword by $k \in \mathcal{K}$, where $\mathcal{K}$ is a lexicon of words. Each keyword k is composed of a sequence of phonemes

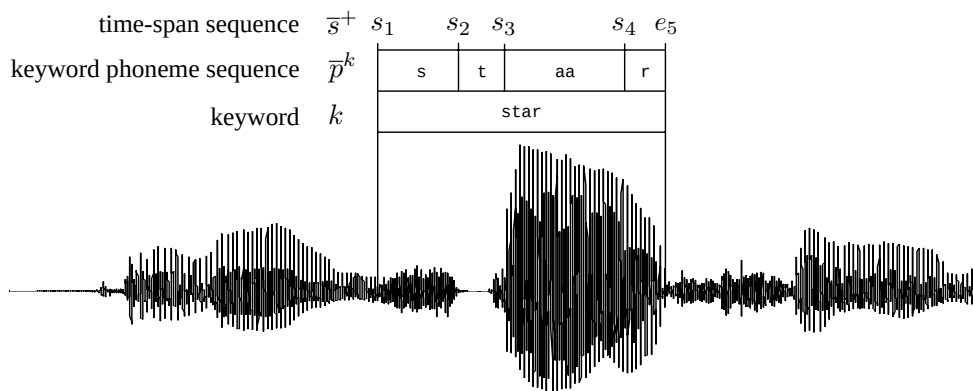


Figure 11.3 Example of our notation. The waveform of the spoken utterance ‘a lone star shone. . .’ taken from the TIMIT corpus. The keyword k is the word *star*. The phonetic transcription $\bar{p}^k$ along with the time span sequence $\bar{s}^+$ is depicted in the figure.

$\bar{p}^k = (p_1, \dots, p_L)$, where $p_l \in \mathcal{P}$ for all $1 \leq l \leq L$ and $\mathcal{P}$ is the domain of the phoneme symbols. The number of phonemes in each keyword may vary from one keyword to another and hence L is not fixed. We denote by $\mathcal{P}^*$ (and similarly $\mathcal{X}^*$) the set of all finite length sequences over $\mathcal{P}$. Let us further define the time span of the phoneme sequence $\bar{p}^k$ in the speech signal $\bar{x}$. We denote by $s_l \in \{1, \dots, T\}$ the start time (in frame units) of phoneme p_l in $\bar{x}$, and by $e_l \in \{1, \dots, T\}$ the end time of phoneme p_l in $\bar{x}$. We assume that the start time of any phoneme p_{l+1} is equal to the end time of the previous phoneme p_l , that is, $e_l = s_{l+1}$ for all $1 \leq l \leq L - 1$. We define the time span (or segmentation) sequence as $\bar{s}^k = (s_1, \dots, s_L, e_L)$. An example of our notation is given in Figure 11.3. Our goal is to learn a *keyword spotter*, denoted $f : \mathcal{X}^* \times \mathcal{P}^* \rightarrow \mathbb{R}$, which takes as input the pair $(\bar{x}, \bar{p}^k)$ and returns a real valued score expressing the confidence that the targeted keyword k is uttered in $\bar{x}$. By comparing this score with a threshold $b \in \mathbb{R}$, we can determine whether $\bar{p}^k$ is uttered in $\bar{x}$.

In discriminative supervised learning we are provided with a training set of examples and a test set (or evaluation set). Specifically, in the task of discriminative keyword spotting we are provided with a two sets of keywords. The first set $\mathcal{K}_{\text{train}}$ is used for training and the second set $\mathcal{K}_{\text{test}}$ is used for evaluation. Note that the lexicon of keywords is a union of both the training set and the test set, $\mathcal{K} = \mathcal{K}_{\text{train}} \cup \mathcal{K}_{\text{test}}$. Algorithmically, we do not restrict a keyword to be only in one set and a keyword that appears in the training set can appear also in the test set. Nevertheless, in our experiments we picked different keywords for training and test and hence $\mathcal{K}_{\text{train}} \cap \mathcal{K}_{\text{test}} = \emptyset$.

A keyword spotter f is often evaluated using the ROC curve. This curve plots the true positive rate (TPR) as a function of the false positive rate (FPR). The TPR measures the fraction of keyword occurrences correctly spotted, while the FPR measures the fraction of negative utterances yielding a false alarm. The points on the curve are obtained by sweeping the threshold b from the largest value output by the system to the smallest one. These values hence correspond to different trade-offs between the two types of errors a keyword spotter

can make, i.e. missing a keyword utterance or raising a false alarm. In order to evaluate a keyword spotter over various trade-offs, it is common to report the AUC as a single value. The AUC hence corresponds to an averaged performance, assuming a flat prior over the different operational settings. Given a keyword k , a set of positive utterances X_k^+ in which k is uttered, and a set of negative utterances X_k^- in which k is not uttered, the AUC can be written as

$$A_k = \frac{1}{|X_k^+||X_k^-|} \sum_{\bar{\mathbf{x}}^+ \in X_k^+} \sum_{\bar{\mathbf{x}}^- \in X_k^-} \mathbb{1}_{\{f(\bar{p}^k, \bar{\mathbf{x}}^+) > f(\bar{p}^k, \bar{\mathbf{x}}^-)\}},$$

where $|\cdot|$ refers to set cardinality and $\mathbb{1}_{\{\pi\}}$ refers to the indicator function and its value is 1 if the predicate π holds and 0 otherwise. The AUC of the keyword k , A_k , hence estimates the probability that the score assigned to a positive utterance is greater than the score assigned to a negative utterance. This quantity is also referred to as the *Wilcoxon–Mann–Whitney statistics* (Cortes and Mohri 2004; Mann and Whitney 1947; Wilcoxon 1945).

As one is often interested in the expected performance over any keyword, it is common to plot the ROC averaged over a set of evaluation keywords $\mathcal{K}_{\text{test}}$, and to compute the corresponding averaged AUC:

$$A_{\text{test}} = \frac{1}{|\mathcal{K}_{\text{test}}|} \sum_{k \in \mathcal{K}_{\text{test}}} A_k.$$

In this study, we introduce a large-margin approach to learn a keyword spotter f from a training set, which achieves a high averaged AUC.

11.3.2 Loss Function and Model Parameterization

In order to build our keyword spotter f , we are given training data consisting of a set of training keywords $\mathcal{K}_{\text{train}}$ and a set of training utterances. For each keyword $k \in \mathcal{K}_{\text{train}}$, we denote with X_k^+ the set of utterances in which the keyword is spoken and with X_k^- the set of all other utterances, in which the keyword is not spoken. Furthermore, for each positive utterance $\bar{\mathbf{x}}^+ \in X_k^+$, we are also given the timing sequence $\bar{s}^+$ of the keyword phoneme sequence $\bar{p}^k$ in $\bar{\mathbf{x}}^+$. Such a timing sequence provides the start and end points of each of the keyword phonemes, and can either be provided by manual annotators or localized with a forced alignment algorithm, as discussed in Chapter 4. Let us define the training set as $\mathcal{T}_{\text{train}} = \{(p^{k_i}, \bar{\mathbf{x}}_i^+, \bar{s}_i^+, \bar{\mathbf{x}}_i^-)\}_{i=1}^m$. For each keyword in the training set there is only one positive utterance and one negative utterance, hence $|X_k^+| = 1$, $|X_k^-| = 1$ and $|\mathcal{K}_{\text{train}}| = m$, and the AUC over the training set becomes

$$A_{\text{train}} = \frac{1}{m} \sum_{i=1}^m \mathbb{1}_{\{f(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^+) > f(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^-)\}}.$$

The selection of a model maximizing this AUC is equivalent to minimizing the loss:

$$\mathcal{L}^{0/1}(f) = 1 - A_{\text{train}} = \frac{1}{m} \sum_{i=1}^m \mathbb{1}_{\{f(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^+) > f(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^-)\}}.$$

The loss $\mathcal{L}^{0/1}$ is unfortunately not suitable for model training since it is a combinatorial quantity that is difficult to minimize directly. We instead adopt a strategy commonly used in

large margin classifiers and employ the convex hinge loss function:

$$\mathcal{L}(f) = \frac{1}{m} \sum_{i=1}^m [1 - f(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^+) + f(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^-)]_+, \quad (11.1)$$

where $[a]_+$ denotes $\max\{0, a\}$. The hinge loss $\mathcal{L}(f)$ upper bounds $\mathcal{L}^{0/1}(f)$: since for any real numbers a and b , $[1 - a + b]_+ \geq \mathbb{1}_{\{a \leq b\}}$, and moreover, when $\mathcal{L}(f) = 0$, then $A_{\text{train}} = 1$, and for any a and b , $[1 - a + b]_+ = 0 \Rightarrow a > b + 1 \Rightarrow a > b$. The hinge loss is related to the ranking loss used in both SVMs for ordinal regression (Herbrich *et al.* 2000) and Ranking SVM (Joachims 2002). These approaches have shown to be successful over highly unbalanced problems, such as information retrieval (Grangier and Bengio 2008; Joachims 2002); using the hinge loss is hence appealing to the keyword spotting problem. The analysis of the relationships between the hinge loss presented in Equation (11.1) and the generalization performance of our keyword spotter is deferred to Section 11.3.4, where we show that minimizing the hinge loss yields a keyword spotter likely to attain a high AUC over unseen data.

Our keyword spotter f is parameterized as

$$f_{\mathbf{w}}(\bar{\mathbf{x}}, \bar{p}^k) = \max_{\bar{s}} \mathbf{w} \cdot \boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{p}^k, \bar{s}), \quad (11.2)$$

where $\mathbf{w} \in \mathbb{R}^n$ is a vector of importance weights, $\boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{p}^k, \bar{s})$ is a feature function vector, measuring different characteristics related to the confidence that the phoneme sequence $\bar{p}^k$ representing the keyword k is uttered in $\bar{\mathbf{x}}$ with the time span $\bar{s}$. Formally, $\boldsymbol{\phi}$ is a function defined as $\boldsymbol{\phi} : \mathcal{X}^* \times (\mathcal{P} \times \mathbb{N})^* \rightarrow \mathbb{R}^n$. In this study we used seven feature functions ($n = 7$), which are similar to those employed in Chapter 4. These functions are described only briefly for the sake of completeness.

There are four *phoneme transition functions*, which aim at detecting transition between phonemes. For this purpose, they compute the frame distance between the frames before and after a hypothesized transition point. Formally,

$$\forall i = 1, 2, 3, 4, \quad \phi_i(\bar{\mathbf{x}}, \bar{p}^k, \bar{s}) = \frac{1}{L} \sum_{j=2}^{L-1} d(\mathbf{x}_{s_j-i}, \mathbf{x}_{s_j+i}), \quad (11.3)$$

where d refers to the Euclidean distance and L refers to the number of phonemes in keyword k .

The *frame based phoneme classifier function* relies on a frame based phoneme classifier to measure the match between each frame and the hypothesized phoneme class,

$$\phi_5(\bar{\mathbf{x}}, \bar{p}^k, \bar{s}) = \frac{1}{L} \sum_{i=1}^L \sum_{t=s_i}^{s_{i+1}-1} \frac{1}{s_{i+1} - s_i} g(\mathbf{x}_t, p_i) \quad (11.4)$$

where $g : \mathcal{X} \times \mathcal{P} \rightarrow \mathbb{R}$ refers to the phoneme classifier, which returns a confidence that the acoustic feature vector at the t th frame, $\mathbf{x}_t$, represents a specific phoneme p_i . Different phoneme classifiers might be applied for this feature. In our case, we conduct experiments relying on two alternative solutions. The first assessed classifier is the hierarchical large-margin classifier presented in Dekel *et al.* (2004), while the second classifier is a Bayes

classifier with one Gaussian Mixture per phoneme class. In the first case, g is defined as the phoneme confidence output by the classifier, while, in the second case, g is defined as the log posterior of the class $g(\mathbf{x}, p) = \log(P(p|\mathbf{x}))$. The presentation of the training setup, as well as the empirical comparison of both solutions, is deferred to Section 11.4.

The *phoneme duration function* measures the adequacy of the hypothesized segmentation $\bar{s}$, with respect to a duration model,

$$\phi_6(\bar{\mathbf{x}}, \bar{p}^k, \bar{s}) = \frac{1}{L} \sum_{i=1}^L \log \mathcal{N}(s_{i+1} - s_i; \mu_{p_i}, \sigma_{p_i}^2), \quad (11.5)$$

where $\mathcal{N}$ denotes the likelihood of a Gaussian duration model, whose mean μ_p and variance σ_p^2 parameters for each phoneme p are estimated over the training data.

The *speaking rate function* measures the stability of the speaking rate,

$$\phi_7(\bar{\mathbf{x}}, \bar{p}^k, \bar{s}) = \frac{1}{L} \sum_{i=2}^L (r_i - r_{i-1})^2, \quad (11.6)$$

where r_i denotes the estimate of the speaking rate for the i th phoneme,

$$r_i = \frac{s_{i+1} - s_i}{\mu_{p_i}}.$$

This set of seven functions has been used in our experiments. Of course, this set can easily be extended to incorporate further features, such as confidences from a triphone frame based classifier or the output of a more refined duration model.

In other words, our keyword spotter outputs a confidence score by maximizing a weighted sum of feature functions over all possible time spans. This maximization corresponds to a search over an exponentially large number of time spans. Nevertheless, it can be performed efficiently by selecting decomposable feature functions, which allows the application of dynamic programming techniques, like those used in HMMs (Rabiner and Juang 1993). Chapter 4 gives a detailed discussion about the efficient computation of Equation (11.2).

11.3.3 An Iterative Training Algorithm

In this section we describe an iterative algorithm for finding the weight vector $\mathbf{w}$. We show in the sequel that the weight vector $\mathbf{w}$ found in this process minimizes the loss $\mathcal{L}(f_{\mathbf{w}})$, hence minimizes the loss $\mathcal{L}^{0/1}$ and in turn resulted in a keyword spotting which attains a high AUC over the training set. We also show that the learned weight vector has good generalization properties on the test set.

The procedure starts by initializing the weight vector to be the zero vector, $\mathbf{w}_0 = 0$. Then, at iteration $i \geq 1$, the algorithm examines the i th training example $(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^+, \bar{s}_i^+, \bar{\mathbf{x}}_i^-)$. The algorithm first predicts the best time span of the keyword phoneme sequence $\bar{p}^{k_i}$ in the negative utterance $\bar{\mathbf{x}}_i^-$:

$$\bar{s}_i^- = \arg \max_{\bar{s}} \mathbf{w}_{i-1} \cdot \boldsymbol{\phi}(\bar{\mathbf{x}}_i^-, \bar{p}_i^k, \bar{s}). \quad (11.7)$$

Then, the algorithm considers the loss on the i th training example and checks that the difference between the score assigned to the positive utterance and the score assigned to

the negative example is greater than 1. Formally, define

$$\Delta\phi_i = \phi(\bar{\mathbf{x}}_i^+, \bar{p}_i^k, \bar{s}_i^+) - \phi(\bar{\mathbf{x}}_i^-, \bar{p}_i^k, \bar{s}_i^-).$$

If $\mathbf{w}_{i-1} \cdot \Delta\phi_i \geq 1$ the algorithm keeps the weight vector for the next iteration, namely, $\mathbf{w}_i = \mathbf{w}_{i-1}$. Otherwise, the algorithm updates the weight vector to minimize the following optimization problem:

$$\mathbf{w}_i = \arg \min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w} - \mathbf{w}_{i-1}\|^2 + c[1 - \mathbf{w} \cdot \Delta\phi_i]_+, \quad (11.8)$$

where the hyperparameter $c \geq 1$ controls the trade-off between keeping the new weight vector close to the previous one and satisfying the constraint for the current example. Equation (11.8) can analytically be solved in closed form (Crammer *et al.* 2006), yielding

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \alpha_i \Delta\phi_i,$$

where

$$\alpha_i = \min \left\{ c, \frac{[1 - \mathbf{w}_{i-1} \cdot \Delta\phi_i]_+}{\|\Delta\phi_i\|^2} \right\}. \quad (11.9)$$

This update is referred to as *passive-aggressive*, since the algorithm *passively* keeps the previous weight ($\mathbf{w}_i = \mathbf{w}_{i-1}$) if the loss of the current training example is already zero ($[1 - \mathbf{w}_{i-1} \cdot \Delta\phi_i]_+ = 0$), while it *aggressively* updates the weight vector to compensate this loss otherwise. At the end of the training procedure, when all training examples have been visited, the best weight $\mathbf{w}$ among $\{\mathbf{w}_0, \dots, \mathbf{w}_m\}$ is selected over a set of validation examples $\mathcal{T}_{\text{valid}}$. The hyperparameter c is also selected to optimize performance on the validation data. The pseudo-code of the algorithm is given in Figure 11.4.

11.3.4 Analysis

In this section, we derive theoretical bounds on the performance of our keyword spotter. Let us first define the *cumulative AUC* on the training set $\mathcal{T}_{\text{train}}$ as follows:

$$\hat{A}_{\text{train}} = \frac{1}{m} \sum_{i=1}^m \mathbb{1}_{\{\mathbf{w}_{i-1} \cdot \phi(\bar{\mathbf{x}}_i^+, \bar{p}_i^{k_i}, \bar{s}_i^+) > \mathbf{w}_{i-1} \cdot \phi(\bar{\mathbf{x}}_i^-, \bar{p}_i^{k_i}, \bar{s}_i^-)\}}, \quad (11.10)$$

where $\bar{s}_i^-$ is generated every iteration step according to Equation (11.7). The examination of the cumulative AUC is of great interest as it provides an estimator for the generalization performance. Note that at each iteration step the algorithm receives new example $(p_i^{k_i}, \bar{\mathbf{x}}_i^+, \bar{s}_i^+, \bar{\mathbf{x}}_i^-)$ and predicts the time span of the keyword in the negative instance $\bar{\mathbf{x}}_i^-$ using the previous weight vector $\mathbf{w}_{i-1}$. Only after the prediction is made the algorithm suffers loss by comparing its prediction to the true time span $\bar{s}_i^+$ of the keyword on the positive utterance $\bar{\mathbf{x}}_i^+$. The cumulative AUC is a weighted sum of the performance of the algorithm on the next unseen training example and hence it is a good estimation of the performance of the algorithm on unseen data during training.

Our first theorem states a competitive bound. It compares the cumulative AUC of the weight vectors series, $\{\mathbf{w}_1, \dots, \mathbf{w}_m\}$, resulted from the iterative algorithm to the best fixed weight vector, $\mathbf{w}^*$, chosen in hindsight, and essentially proves that, for any sequence of

Input: Training set $\mathcal{T}_{\text{train}}$, validation set $\mathcal{T}_{\text{valid}}$; parameter c .

Initialize: $\mathbf{w}_0 = 0$.

Loop: for each $(p^{k_i}, \bar{\mathbf{x}}_i^+, \bar{s}_i^+, \bar{\mathbf{x}}_i^-) \in \mathcal{T}_{\text{train}}$

1. let $\bar{s}_i^- = \arg \max_{\bar{s}} \mathbf{w}_{i-1} \cdot \boldsymbol{\phi}(\bar{\mathbf{x}}_i^-, \bar{p}^{k_i}, \bar{s})$
2. let $\Delta \boldsymbol{\phi}_i = \boldsymbol{\phi}(\bar{\mathbf{x}}_i^+, \bar{p}^{k_i}, \bar{s}_i^+) - \boldsymbol{\phi}(\bar{\mathbf{x}}_i^-, \bar{p}^{k_i}, \bar{s}_i^-)$
3. if $\mathbf{w}_{i-1} \cdot \Delta \boldsymbol{\phi}_i < 1$ then

let $\alpha_i = \min \left\{ c, \frac{1 - \mathbf{w}_{i-1} \cdot \Delta \boldsymbol{\phi}_i}{\|\Delta \boldsymbol{\phi}_i\|^2} \right\}$
 update $\mathbf{w}_i = \mathbf{w}_{i-1} + \alpha_i \cdot \Delta \boldsymbol{\phi}_i$

Output: $\mathbf{w}$ achieving the highest AUC over $\mathcal{T}_{\text{valid}}$:

$$\mathbf{w} = \arg \min_{\mathbf{w} \in \{\mathbf{w}_1, \dots, \mathbf{w}_m\}} \frac{1}{m_{\text{valid}}} \sum_{j=1}^{m_{\text{valid}}} \mathbb{1}_{\{\max_{\bar{s}^+} \mathbf{w} \cdot \boldsymbol{\phi}(\bar{\mathbf{x}}_j^+, \bar{p}^{k_j}, \bar{s}^+) > \max_{\bar{s}^-} \mathbf{w} \cdot \boldsymbol{\phi}(\bar{\mathbf{x}}_j^-, \bar{p}^{k_j}, \bar{s}^-)\}}$$

Figure 11.4 Passive-aggressive training.

examples, our algorithms cannot do much worse than the best fixed weight vector. Formally, it shows that the cumulative area *above* the curve, $1 - \hat{A}_{\text{train}}$, is smaller than the weighted average loss $\mathcal{L}(f_{\mathbf{w}^*})$ of the best fixed weight vector $\mathbf{w}^*$ and its weighted complexity, $\|\mathbf{w}^*\|$. That is, the cumulative AUC of the iterative training algorithm is going to be high, given that the loss of the best solution is small, the complexity of the best solution is small and that the number of training examples, m , is sufficiently large.

Theorem 11.3.1 Let $\mathcal{T}_{\text{train}} = \{(\bar{p}^{k_i}, \bar{\mathbf{x}}_i^+, \bar{s}_i^+, \bar{\mathbf{x}}_i^-)\}_{i=1}^m$ be a set of training examples and assume that for all $k, \bar{\mathbf{x}}$ and $\bar{s}$ we have that $\|\boldsymbol{\phi}(\bar{\mathbf{x}}, \bar{p}^k, \bar{s})\| \leq 1/\sqrt{2}$. Let $\mathbf{w}^*$ be the best weight vector selected under some optimization criterion by observing all instances in hindsight. Let $\mathbf{w}_1, \dots, \mathbf{w}_m$ be the sequence of weight vectors obtained by the algorithm in Figure 11.4 given the training set $\mathcal{T}_{\text{train}}$. Then,

$$1 - \hat{A}_{\text{train}} \leq \frac{1}{m} \|\mathbf{w}^*\|^2 + \frac{2c}{m} \mathcal{L}(f_{\mathbf{w}^*}) \quad (11.11)$$

where $c \geq 1$ and $\hat{A}_{\text{train}}$ is the cumulative AUC defined in Equation 11.10.

Proof. Denote by $\ell_i(\mathbf{w})$ the instantaneous loss the weight vector $\mathbf{w}$ suffers on the i th example, that is,

$$\ell_i(\mathbf{w}) = [1 - \mathbf{w} \cdot \boldsymbol{\phi}(\bar{\mathbf{x}}_i^+, \bar{p}^{k_i}, \bar{s}_i^+) + \max_{\bar{s}} \mathbf{w} \cdot \boldsymbol{\phi}(\bar{\mathbf{x}}_i^-, \bar{p}^{k_i}, \bar{s})]_+.$$

The proof of the theorem relies on Lemma 1 and Theorem 4 in Crammer *et al.* (2006). Lemma 1 in Crammer *et al.* (2006) implies that

$$\sum_{i=1}^m \alpha_i (2\ell_i(\mathbf{w}_{i-1}) - \alpha_i \|\Delta \boldsymbol{\phi}_i\|^2 - 2\ell_i(\mathbf{w}^*)) \leq \|\mathbf{w}^*\|^2. \quad (11.12)$$

Now if the algorithm makes a prediction mistake and the predicted confidence of the best time span of the keyword in a negative utterance is higher than the confidence of the true time span of the keyword in the positive example, then $\ell_i(\mathbf{w}_{i-1}) \geq 1$. Using the assumption that $\|\phi(\bar{\mathbf{x}}, \bar{p}^k, \bar{s})\| \leq 1/\sqrt{2}$, which means that $\|\Delta\phi(\bar{\mathbf{x}}, \bar{p}^k, \bar{s})\|^2 \leq 1$, and the definition of α_i given in Equation 11.9, when substituting $[1 - \mathbf{w}_{i-1} \cdot \Delta\phi_i]_+$ for $\ell_i(\mathbf{w}_{i-1})$ in its numerator, we conclude that if a prediction mistake occurs then it holds that

$$\alpha_i \ell_i(\mathbf{w}_{i-1}) \geq \min \left\{ \frac{\ell_i(\mathbf{w}_{i-1})}{\|\Delta\phi_i\|^2}, c \right\} \geq \min\{1, c\} = 1. \quad (11.13)$$

Summing over all the prediction mistakes made on the entire training set $\mathcal{T}_{\text{train}}$ and taking into account that $\alpha_i \ell_i(\mathbf{w}_{i-1})$ is always non-negative, we have

$$\sum_{i=1}^m \alpha_i \ell_i(\mathbf{w}_{i-1}) \geq \sum_{i=1}^m \mathbb{1}_{\{\mathbf{w}_{i-1} \cdot \phi(\bar{\mathbf{x}}_i^+, \bar{p}^{k_i}, \bar{s}_i^+) \leq \mathbf{w}_{i-1} \cdot \phi(\bar{\mathbf{x}}_i^-, \bar{p}^{k_i}, \bar{s}_i^-)\}}. \quad (11.14)$$

Again using the definition of α_i , we know that $\alpha_i \ell_i(\mathbf{w}^*) \leq c \ell_i(\mathbf{w}^*)$ and that $\alpha_i \|\Delta\phi_i\|^2 \leq \ell_i(\mathbf{w}_{i-1})$. Plugging these two inequalities and Equation (11.14) into Equation (11.12) we get

$$\sum_{i=1}^m \mathbb{1}_{\{\mathbf{w}_{i-1} \cdot \phi(\bar{\mathbf{x}}_i^+, \bar{p}^{k_i}, \bar{s}_i^+) \leq \mathbf{w}_{i-1} \cdot \phi(\bar{\mathbf{x}}_i^-, \bar{p}^{k_i}, \bar{s}_i^-)\}} \leq \|\mathbf{w}^*\|^2 + 2c \sum_{i=1}^m \ell_i(\mathbf{w}^*). \quad (11.15)$$

The theorem follows by replacing the sum over prediction mistakes to a sum over prediction hits and plugging the definition of the cumulative AUC given in Equation (11.10). $\square$

The next theorem states that the output of our algorithm is likely to have good generalization, namely, the expected value of the AUC resulted from decoding on unseen test set is likely to be large.

Theorem 11.3.2 *Under the same conditions of Theorem 11.3.1 assume that the training set $\mathcal{T}_{\text{train}}$ and the validation set $\mathcal{T}_{\text{valid}}$ are both sampled i.i.d. from a distribution $\mathcal{D}$. Denote by m_{valid} the size of the validation set. With probability of at least $1 - \delta$ we have*

$$\begin{aligned} 1 - A &= \mathbb{E}_{\mathcal{D}}[\mathbb{1}_{\{f(\bar{\mathbf{x}}^+, \bar{p}^k) \leq f(\bar{\mathbf{x}}^-, \bar{p}^k)\}}] = \Pr_{\mathcal{D}}[f(\bar{\mathbf{x}}^+, \bar{p}^k) \leq f(\bar{\mathbf{x}}^-, \bar{p}^k)] \\ &\leq \frac{1}{m} \sum_{i=1}^m \ell_i(\mathbf{w}^*) + \frac{\|\mathbf{w}^*\|^2}{m} + \frac{\sqrt{2 \ln(2/\delta)}}{\sqrt{m}} + \frac{\sqrt{2 \ln(2m/\delta)}}{\sqrt{m_{\text{valid}}}}, \end{aligned} \quad (11.16)$$

where A is the mean AUC defined as $A = \mathbb{E}_{\mathcal{D}}[\mathbb{1}_{\{f(\bar{\mathbf{x}}^+, \bar{p}^k) > f(\bar{\mathbf{x}}^-, \bar{p}^k)\}}]$ and

$$\ell_i(\mathbf{w}) = \left[1 - \mathbf{w} \cdot \phi(\bar{\mathbf{x}}_i^+, \bar{p}^{k_i}, \bar{s}_i^+) + \max_{\bar{s}} \mathbf{w} \cdot \phi(\bar{\mathbf{x}}_i^-, \bar{p}^{k_i}, \bar{s}) \right]_+.$$

The proof of the theorem goes along the lines of the proof of Theorem 4.5.2 in Chapter 4. The theorem states that the resulted $\mathbf{w}$ of the iterative algorithm generalizes, with high probability, and is going to have high expected AUC on unseen test data.

11.4 Experiments and Results

We started by training the iterative algorithm on the TIMIT training set. We then conducted two types of experiments to evaluate the effectiveness of the proposed discriminative method. First, we compared the performance of the discriminative method with a standard monophone HMM keyword spotter on the TIMIT test set. Second, we compared the robustness of both the discriminative method and the monophone HMM with respect to changing recording conditions by using the models trained on the TIMIT, evaluated on the Wall Street Journal (WSJ) corpus.

11.4.1 The TIMIT Experiments

The TIMIT corpus (Garofolo 1993) consists of read speech from 630 American speakers, with 10 utterances per speaker. The corpus provides manually aligned phoneme and word transcriptions for each utterance. It also provides a standard split into training and test data. From the training part of the corpus, we extracted three disjoint sets consisting of 1500, 300 and 200 utterances. The first set was used as the training set of the phoneme classifier and was used by our fifth feature function ϕ_5 . The second set was used as the training set for our discriminative keyword spotter, while the third set was used as the validation set to select the hyperparameter c and the best weight vector $\mathbf{w}$ seen during training. The test set was solely used for evaluation purposes. From each of the last two splits of the training set, 200 words of length greater than or equal to four phonemes were chosen in random. From the test set 80 words were chosen in random as described below.

Mel-frequency cepstral coefficients (MFCC), along with their first (Δ) and second derivatives ($\Delta\Delta$), were extracted every 10 ms. These features were used by the first five feature functions $\phi_1, \dots, \phi_5$. Two types of phoneme classifier were used for the fifth feature function ϕ_5 , namely, a large margin phoneme classifier (Dekel *et al.* 2004) and a GMM model. Both classifiers were trained to predict 39 phoneme classes (Lee and Hon 1989) over the first part of the training set. The large margin classifier corresponds to a hierarchical classifier with Gaussian kernel, as presented in Dekel *et al.* (2004), where the score assigned to each frame for a given phoneme was used as the function g in Equation (11.4). The GMM model corresponded to a Bayes classifier combining one GMM per class and the phoneme prior probabilities, both learned from the training data. In that case, the log posterior of a phoneme given the frame vector was used as the function g in Equation (11.4). The hyperparameters of both phoneme classifiers were selected to maximize the frame accuracy over part of the training data held out during parameter fitting. In the following, the discriminative keyword spotter relying on the features from the hierarchical phoneme classifier is referred to as *Discriminative/Hier*, while the model relying on the GMM log posteriors is referred to as *Discriminative/GMM*.

We compared the results of both *Discriminative/Hier* and *Discriminative/GMM* with a monophone HMM baseline, in which each phoneme was modeled with a left–right HMM of five emitting states. The density of each state was modeled with a 40-Gaussian GMM. Training was performed over the whole TIMIT training set. *Embedded training* was applied, i.e. after an initial training phase relying on the provided phoneme alignment, a second training phase which dynamically determines the most likely alignment was applied. The hyperparameters of this model (the number of states per phoneme, the number of Gaussians

Table 11.1 Area under the ROC curve (AUC) of different models trained on the TIMIT training set and evaluated on the TIMIT test set (the higher the better)

Model	AUC
HMM/Viterbi	0.942
HMM/Ratio	0.952
Discriminative/GMM	0.971
Discriminative/Hier	0.996

per state, as well as the number of expectation-maximization iterations) were selected to maximize the likelihood of a held-out validation set.

The phoneme models of the trained HMM were then used to build a keyword spotting HMM, composed of two sub-models: the keyword model and the garbage model, as illustrated in Figure 11.1. The keyword model was an HMM which estimated the likelihood of an acoustic sequence given that the sequence represented the keyword phoneme sequence. The garbage model was an HMM composed of all phoneme HMMs fully connected to each other, which estimated the likelihood of any phoneme sequence. The overall HMM fully connected the keyword model and the garbage model. The detection of a keyword in a given utterance was performed by checking whether the Viterbi best path passes through the keyword model, as explained in Section 11.2. In this model, the keyword transition probability set the trade-off between the true positive rate and the ROC curve was plotted by varying this probability. This model is referred to as *HMM/Viterbi*.

We also experimented with an alternative decoding strategy, in which the system output the ratio of the likelihood of the acoustic sequence knowing the keyword was uttered versus the likelihood of the sequence knowing the keyword was *not* uttered, as discussed in Section 11.2. In this case, the first likelihood was determined by an HMM forcing an occurrence of the keyword, and the second likelihood was determined by the garbage model, as illustrated in Figure 11.2. This likelihood-ratio strategy is referred to as *HMM/Ratio* in the following.

The evaluation of discriminative and HMM-based models was performed over 80 keywords, randomly selected among the words occurring in the TIMIT test set. This random sampling of the keyword set aimed at evaluating the expected performance over any keyword. For each keyword k , we considered a spotting problem, which consisted of a set of positive utterances X_k^+ and a set of negative utterance X_k^- . Each positive set X_k^+ contained between 1 and 20 sequences, depending on the number of occurrences of k in the TIMIT test set. Each negative set contained 20 sequences, randomly sampled among the utterances of TIMIT which do not contain k . This setup represented an unbalanced problem, with only 10% of the sequences being labeled as positive.

Table 11.1 reports the average AUC results of the 80 test keywords, for different models trained on the TIMIT training set and evaluated on the TIMIT test set. These results show the advantage of our discriminative approach. The two discriminative models outperform the two HMM-based models. The improvement introduced by our discriminative model algorithm can be observed when comparing the performance of Discriminative/GMM with the performance of the HMM spotters. In that case, both spotters rely on GMMs to estimate the frame likelihood given a phoneme class. In our case we use that probability to compute the feature ϕ_5 , while the HMM uses it as the state emission probability.

Table 11.2 The distribution of the 80 keywords among the models which better spotted them. Each row in the table represents the keywords for which the model written at the beginning of the row received the highest area under the ROC curve. The models were trained on the TIMIT training set and evaluated on the TIMIT test set

Best model	Keywords
Discriminative/Hier	absolute admitted apartments apparently argued controlled depicts dominant drunk efficient followed freedom introduced millionaires needed obvious radiation rejected spilled street superb sympathetically weekday (23 keywords)
HMM/Ratio	materials (1 keyword)
No differences	aligning anxiety bedrooms brand camera characters cleaning climates creeping crossings crushed decaying demands dressy episode everything excellent experience family firing forgiveness fulfillment functional grazing henceforth ignored illnesses imitate increasing inevitable January mutineer package paramagnetic patiently pleasant possessed pressure recriminations redecorating secularist shampooed solid spreader story strained streamlined stripped stupid surface swimming unenthusiastic unlined urethane usual walking (56 keywords)

Moreover, our keyword spotter can benefit from non-probabilistic frame based classifiers, as illustrated with *Discriminative/Hier*. This model relies on the output of a large margin classifier, which outperforms all other models, and reaches a mean AUC of 0.996. In order to verify whether the differences observed on averaged AUC could be due only to a few keywords, we applied the Wilcoxon test (Rice 1995) to compare the results of both HMM approaches (HMM/Viterbi and HMM/Ratio) with the results of both discriminative approaches (Discriminative/GMM and Discriminative/Hier). At the 90% confidence level, the test rejected this hypothesis, showing that the performance gain of the discriminative approach is consistent over the keyword set.

Table 11.2 further presents the performance per keyword and compares the results of the best HMM configuration, HMM/Ratio with the performance of the best discriminative configuration, Discriminative/Hier. Out of a total of 80 keywords, 23 keywords were better spotted with the discriminative model, one keyword was better spotted with the HMM, and both models yielded the same spotting accuracy for 56 keywords. The discriminative model seems to be better for shorter keywords, as it outperforms the HMM for most of the keywords of five phonemes or less (e.g. *drunk*, *spilled*, *street*).

11.4.2 The WSJ Experiments

WSJ (Paul and Baker 1992) is a large corpus of American English. It consists of read and spontaneous speech corresponding to the reading and the dictation of articles from the *Wall Street Journal*. In the following, all models were trained on the TIMIT training set and evaluated on the *si_tr_s* subset of WSJ. This subset corresponds to the recordings of 200 speakers. Compared with TIMIT, this subset introduces several variations, both regarding the type of sentences recorded and the recording conditions (Paul and Baker 1992).

Table 11.3 Area under the ROC curve (AUC) of different models trained on the TIMIT training set and evaluated on the `si_tr_s` subset of WSJ (the higher the better)

Model	AUC
HMM/Viterbi	0.868
HMM/Ratio	0.884
Discriminative/GMM	0.922
Discriminative/Hier	0.914

These experiments hence evaluate the robustness of the different approaches when they encounter differing conditions for training and testing. As for TIMIT, the evaluation is performed over 80 keywords randomly selected from the corpus transcription. For each keyword k , the evaluation was performed over a set X_k^+ , containing between 1 and 20 positive sequences, and a X_k^- , containing 20 randomly selected negative sequences. This setup also represents an unbalanced problem, with 27% of the sequences being labeled as positive.

Table 11.3 reports the average AUC results of the 80 test keywords, for different models trained on the TIMIT training set and evaluated on the `si_tr_s` subset of WSJ. Overall, the results show that the differences between the TIMIT training conditions and the WSJ test conditions affect the performance of all models. However, the measured performance still yields acceptable performance in all cases (AUC of 0.868 in the worst case). Comparing the individual model performance, the WSJ results confirm the conclusions of TIMIT experiments and the discriminative spotters outperform the HMM-based alternatives. For the HMM models, HMM/Ratio outperforms HMM/Viterbi as in the TIMIT experiments. For the discriminative spotters, Discriminative/GMM outperforms Discriminative/Hier, which was not the case over TIMIT. Since these two models only differ in the frame based classifier used as the feature function ϕ_5 , this result certainly indicates that the hierarchical frame based classifier on which Discriminative/Hier relies is less robust to the acoustic condition changes than the GMM alternative. As with TIMIT, we checked whether the differences observed on the whole set could be due to a few keywords. The Wilcoxon test rejected this hypothesis at the 90% confidence level, for the four tests comparing Discriminative/GMM and Discriminative/Hier with HMM/Viterbi and HMM/Hier.

We further compared the best discriminative spotter, Discriminative/GMM, and the best HMM spotter HMM/Ratio over each keyword. These results are summarized in Table 11.4. Out of the 80 keywords, the discriminative model outperforms the HMM for 50 keywords, the HMM outperforms the discriminative model for 20 keywords and both models yield the same results for 10 keywords. As with the TIMIT experiments, the discriminative model is shown to be especially advantageous for short keywords, with five phonemes or less (e.g. *Adams*, *kings*, *servng*).

Overall, the experiments over both WSJ and TIMIT highlight the advantage of our discriminative learning method.

11.5 Conclusions

This chapter introduced a discriminative method for the keyword spotting problem. In this task, the model receives a keyword and a spoken utterance as input and should decide whether

Table 11.4 The distribution of the 80 keywords among the models which better spotted them. Each row in the table represents the keywords for which the model written at the beginning of the row received the highest area under the ROC curve. The models were trained on the TIMIT training set but evaluated on the `si_tr_s` subset of WSJ

Best model	Keywords
Discriminative/Hier	Adams additions Allen Amerongen apiece buses Bushby Colombians consistently cracked dictate drop fantasy fills gross Higa historic implied interact kings list lobby lucrative measures Melbourne millions Munich nightly observance owning plus proudly queasy regency retooling Rubin scramble Seidler serving significance sluggish strengthening Sutton’s tariffs Timberland today truths understands withhold Witter’s (50 keywords)
HMM/Ratio	artificially Colorado elements Fulton itinerary longer lunchroom merchant mission multilateral narrowed outlets Owens piper replaced reward sabotaged shards spurt therefore (20 keywords)
No differences	aftershocks Americas farms Flamson hammer homosexual philosophically purchasers sinking steel-makers (10 keywords)

the keyword is uttered in the utterance. Keyword spotting corresponds to an unbalanced detection problem, since, in standard setups, most tested utterances do not contain the targeted keyword. In that unbalanced context, the AUC is generally used for evaluation. This work proposed a learning algorithm, which aims at maximizing the AUC over a set of training spotting problems. Our strategy is based on a large margin formulation of the task, and relies on an efficient iterative training procedure. The resulting model contrasts with standard approaches based on HMMs, for which the training procedure does not rely on a loss directly related to the spotting task. Compared with such alternatives, our model is shown to yield significant improvements over various spotting problems on the TIMIT and the WSJ corpus. For instance, the best HMM configuration over TIMIT reaches an AUC of 0.953, compared with an AUC of 0.996 for the best discriminative spotter.

Several potential directions of research can be identified from this work. In its current configuration, our keyword spotter relies on the output of a pre-trained frame based phoneme classifier. It would be of great interest to learn the frame based classifier and the keyword spotter jointly, so that all model parameters are selected to maximize the performance on the final spotting task.

Also, our work currently represents keywords as sequence of phonemes, without considering the neighboring context. Possible improvement might result from the use of phonemes in context, such as triphones. We hence plan to investigate the use of triphones in a discriminative framework, and to compare the resulting model with triphone-based HMMs. More generally, our model parameterization offers greater flexibility to incorporate new features compared with probabilistic approaches such as HMMs. Therefore, in addition to triphones, features extracted from the speaker identity, the channel characteristics or the linguistic context could possibly be included to improve performance.

Acknowledgments

This research was partly performed while David Grangier was visiting Google Inc. (Mountain View, USA), and while Samy Bengio was with the IDIAP Research Institute (Martigny, Switzerland). This research was supported by the European PASCAL Network of Excellence and the DIRAC project.

References

- Bahl LR, Brown PF, de Souza P and Mercer RL 1986 Maximum mutual information estimation of hidden Markov model parameters for speech recognition. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Benayed Y, Fohr D, Haton JP and Chollet G 2003 Confidence measures for keyword spotting using support vector machines. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Benayed Y, Fohr D, Haton JP and Chollet G 2004 Confidence measure for keyword spotting using support vector machines. *Proceedings of the International Conference on Audio, Speech and Signal Processing*, pp. 588–591.
- Bilmes JA 1998 A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixture and hidden Markov models. International Computer Science Institute, Berkeley, CA, USA. *Technical Report TR-97-021*.
- Boite JM, Bourlard H, D’hoore B and Haesen M 1993 Keyword recognition using template concatenation. *Proceedings of the European Conference on Speech and Communication Technologies (EUROSPEECH)*. International Speech Communication Association.
- Bridle JS 1973 An efficient elastic-template method for detecting given words in running speech. *Proceedings of the British Acoustic Society Meeting*. British Acoustic Society.
- Chang E 1995 Improving word spotting performance with limited training. Data Massachusetts Institute of Technology (MIT), PhD thesis.
- Cortes C and Mohri M 2004 Confidence intervals for the area under the ROC curve. *Advances in Neural Information Processing Systems (NIPS)*. MIT Press.
- Crammer K, Dekel O, Keshet J, Shalev-Shwartz S and Singer Y 2006 Online passive aggressive algorithms. *Journal of Machine Learning Research* 7.
- Dekel O, Keshet J and Singer Y 2004 Online algorithm for hierarchical phoneme classification. *Workshop on Multimodal Interaction and Related Machine Learning Algorithms; Lecture Notes in Computer Science*. Springer-Verlag, pp. 146–159.
- Garofolo JS 1993 TIMIT acoustic-phonetic continuous speech corpus. Linguistic Data Consortium, Philadelphia, PA, USA. *Technical Report LDC93S1*.
- Grangier D and Bengio S 2008 A discriminative kernel-based model to rank images from text queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.
- Herbrich R, Graepel T and Obermayer K 2000 Large margin rank boundaries for ordinal regression, in *Advances in Large Margin Classifiers* (eds. A Smola, B Schölkopf and D Schuurmans). MIT Press.
- Higgins AL and Wohlford RE 1985 Keyword recognition using template concatenation. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Joachims T 2002 Optimizing search engines using clickthrough data. *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*. Association for Computing Machinery.

- Junkawitsch J, Ruske G and Hoege H 1997 Efficient methods in detecting keywords in continuous speech. *Proceedings of the European Conference on Speech and Communication Technologies (EUROSPEECH)*. International Speech Communication Association.
- Kawabata T, Hanazawa T and Shikano K 1988 Word spotting method based on HMM phoneme recognition. *Journal of the Acoustical Society of America (JASA)* **1**(84).
- Ketabdar H, Vepa J, Bengio S and Bourlard H 2006 Posterior based keyword spotting with a priori thresholds *Proceedings of the InterSpeech Conference*. International Speech Communication Association.
- Lee KF and Hon HF 1988 Large-vocabulary speaker-independent continuous speech recognition using HMM. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Lee KF and Hon HW 1989 Speaker independent phone recognition using hidden Markov models. *IEEE Transactions on Acoustics, Speech and Signal Processing (TASSP)* **11**(37).
- Mann H and Whitney D 1947 On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics* **1**(18).
- Paul D and Baker J 1992 The design for the Wall Street Journal-based CSR corpus. *Proceedings of the Human Language Technology Conference (HLT)*. Morgan Kaufmann.
- Rabiner L and Juang B 1993 *Fundamentals of Speech Recognition*. Prentice-Hall, Upper Saddle River, NJ, USA.
- Rice J 1995 *Rice, Mathematical Statistics and Data Analysis*. Duxbury Press.
- Rose RC and Paul DB 1990 A hidden Markov model based keyword recognition system. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Sandness ED and Hetherington IL 2000 Keyword-based discriminative training of acoustic models. *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*. IEEE Computer Society.
- Silaghi MC and Bourlard H 1999 Iterative posterior-based keyword spotting without filler models. *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, Keystone, USA. IEEE Computer Society, pp. 213–216.
- Sukkar RA, Seltur AR, Rahim MG and Lee CH 1996 Utterance verification of keyword strings using word-based minimum verification error training. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Weintraub M 1993 Keyword spotting using SRI's DECIPHER large vocabulary speech recognition system. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Weintraub M 1995 LVCSR log-likelihood ratio scoring for keyword spotting. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Weintraub M, Beaufays F, Rivlin Z, Konig Y and Stolcke A 1997 Neural-network based measures of confidence for word recognition. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE Computer Society.
- Wilcoxon F 1945 Individual comparisons by ranking methods. *Biometrics Bulletin* **1**.
- Wilpon JG, Rabiner LR, Lee CH and Goldman ER 1990 Automatic recognition of keywords in unconstrained speech using hidden Markov models. *IEEE Transactions on Acoustics, Speech and Signal Processing (TASSP)* **38**(11).

Exhibit 12

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of:	Kenneth Mixter et al.	Confirmation No.:	5709
Serial No.:	16/786,943	Art Unit:	2658
Filed:	February 10, 2020	Examiner:	Not yet assigned
For:	Device Leadership Negotiation Among Voice Interface Devices	Attorney Docket No:	060963-7419-US

**RESPONSE TO THE NOTICE TO FILE MISSING PARTS
OF NONPROVISIONAL APPLICATION**

Mail Stop: MISSING PARTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the Notice to File Missing Parts mailed February 28, 2020, enclosed is the Notice to File Missing Parts of Nonprovisional Application for the above-identified application.

An Application Data Sheet was submitted with the application and no changes need to be made.

Also enclosed please find a Fee Transmittal Form setting forth the calculation for the statutory basic filing, search and examination fees, together with the surcharge due in this matter. The fees calculated for the statutory basic filing, search and examination fees together with the surcharge are estimated to be \$1880. The Commissioner is hereby authorized to charge the

required fees or credit any overpayments to Morgan, Lewis & Bockius LLP Deposit Account
No. 50-0310 (order no. 060963-7419-US).

Respectfully submitted,

Date: March 19, 2020

/Douglas J. Crisman/ 39,951
Douglas J. Crisman (Reg. No.)
MORGAN, LEWIS & BOCKIUS LLP
1400 Page Mill Road
Palo Alto, CA 94304
(650) 843-4000

Electronically Filed March 19, 2020

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of:	Kenneth Mixter et al.	Confirmation No.:	5709
Serial No.:	16/786,943	Art Unit:	2658
Filed:	February 10, 2020	Examiner:	To be assigned
For:	<i>Device Leadership Negotiation Among Voice Interface Devices</i>	Atty. Docket No.:	060963-7419-US

PRELIMINARY AMENDMENT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

The enclosed Preliminary Amendment is submitted prior to receiving a first office action.

The Commissioner is hereby authorized to charge any required fee(s) to Morgan, Lewis & Bockius LLP Deposit Account No. 50-0310 (order no. 060963-7419-US).

IN THE CLAIMS:

Rewrite the pending claims and add new claims as follows:

1. (Currently Amended) A method, ~~comprising~~ performed at a first electronic device of a plurality of electronic devices, each ~~electronic device~~ of the plurality of electronic devices comprising one or more microphones, ~~a speaker~~, one or more processors, and memory storing one or more programs for execution by the one or more processors, the method comprising:
 - detecting a voice input from a user;
 - ~~determining a state of the first electronic device and a relevance of the voice input to the first electronic device;~~
 - ~~identifying a subset of electronic devices from the plurality to which the voice input is relevant;~~
 - ~~in accordance with a determination that the subset includes the first electronic device;~~
 - determining a first value score ~~of a criterion~~ associated with the voice input; ~~and~~
 - receiving second values associated with the voice input scores of the criterion ~~scores of the criterion~~ from other electronic devices of the plurality of electronic devices in the subset; and
 - in accordance with a determination that the first value score ~~score~~ is higher than the second values scores, responding to the detected input.
2. (New) The method of claim 1, further comprising:
 - in accordance with a determination that the first score is not higher than the second scores, forgoing responding to the detected input.
3. (New) The method of claim 2, wherein forgoing responding to the detected input further comprises changing a state of operation of the first electronic device.
4. (New) The method of claim 1, further comprising:
 - in accordance with a determination that the first value is not higher than the second values, identifying a criterion associated with the voice input; and
 - in accordance with a determination that the criterion is the most relevant to the first electronic device, responding to the detected input.

5. (New) The method of claim 4, further comprising:
determining a state of the first electronic device;
wherein the determination that the criterion is the most relevant to the first electronic device is based on the state of the electronic device.
6. (New) The method of claim 4, wherein the criterion comprises one or more of: a type of request, a command in the voice input, and a state of each electronic device of the plurality of electronic devices.
7. (New) The method of claim 1, further comprising:
at the first electronic device:
detecting subsequent voice inputs from the user;
pre-processing the subsequent voice inputs; and
transmitting the pre-processed voice inputs to a server system;
wherein the other electronic devices forgo the detecting, pre-processing, and transmitting.
8. (New) The method of claim 1, wherein responding to the detected input comprises outputting an audible and/or a visual response to the detected voice input.
9. (New) The method of claim 1, wherein the first electronic device further comprises a speaker, and responding to the detected input further comprises outputting an audible response.
10. (New) A first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising:
one or more microphones;
one or more processors; and
memory storing one or more programs to be executed by the one or more processors, the one or more programs comprising instructions for:
detecting a voice input from a user;
determining a first value associated with the voice input;
receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices; and

in accordance with a determination that the first value is higher than the second values, responding to the detected input.

11. (New) The first electronic device of claim 10, the one or more programs further comprising instructions for:

recognizing a command in the voice input; and

in accordance with a determination that a type of the command is related to the first electronic device, responding to the detected voice input.

12. (New) The first electronic device of claim 10, the one or more programs further comprising instructions for:

communicating to the other devices of the plurality of electronic devices to forgo responding to the detected output.

13. (New) The first electronic device of claim 10, wherein the instruction for detecting the voice input is performed at intervals of a predefined number of voice inputs.

14. (New) The first electronic device of claim 10, wherein:

the plurality of electronic devices is communicatively coupled through a local network; and

the receiving is performed through the local network.

15. (New) The first electronic device of claim 10, wherein the first electronic device is configured to be awakened by a plurality of affordances including a voice-based affordance; and wherein detecting the voice input is in response to the awakening of the first electronic device.

16. (New) A system comprising a plurality of electronic devices, each of the electronic devices including one or more microphones, one or more processors, and memory storing one or more programs for execution by the one or more processors, each of the electronic devices programmed to:

detect a voice input;

determine a first value associated with the detected voice input;

receive respective second values from other electronic devices of the plurality of electronic devices;

compare the first value and the respective second values; and

in accordance with a determination that the first value is higher than the respective second values, respond to the voice input.

17. (New) The system of claim 16, wherein the first value comprises a first quality score of the voice input.

18. (New) The system of claim 17, wherein the first quality score comprises a confidence level of detection of the voice input.

19. (New) The system of claim 17, wherein the first quality score comprises a signal-to-noise rating of detection of the voice input.

20. (New) The system of claim 16, wherein each of the second values is indicative of a quality of the voice input detected by a respective one of the other electronic devices.

Remarks

In this Preliminary Amendment, claim 1 has been amended. New claims 2-20 have been added. The amendments are supported by at least paragraphs [0086] to [0099] and [0133], and Figures 2 to 7 of the application as filed. No new matter has been added.

Conclusion

The Examining Office is invited to call the undersigned attorney at (650) 843-4000, if a telephone call could help resolve any remaining items.

Respectfully submitted,

Date: March 19, 2020

<u>/Douglas J. Crisman/</u>	39,951
Douglas J. Crisman	(Reg. No.)
MORGAN, LEWIS & BOCKIUS LLP	
1400 Page Mill Road	
Palo Alto, CA 94304	
(650) 843-4000	



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
16/786,943	02/10/2020	Kenneth Mixter	060963-7419-US	5709
82750	7590	11/05/2020		
Morgan, Lewis & Bockius LLP (PA)(Google)			EXAMINER	
1400 Page Mill Road			COLUCCI, MICHAEL C	
Palo Alto, CA 94304-1124				
			ART UNIT	PAPER NUMBER
			2656	
			NOTIFICATION DATE	DELIVERY MODE
			11/05/2020	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

padocketingdepartment@morganlewis.com
vskliba@morganlewis.com

Office Action Summary

Application No.

16/786,943

Applicant(s)

Mixer et al.

Examiner

MICHAEL COLUCCI

Art Unit

2656

AIA (FITF) Status

Yes

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTHS FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☐ Responsive to communication(s) filed on _____.
☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.
- 4) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims*

- 5) ☒ Claim(s) 1-20 is/are pending in the application.
5a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 6) ☐ Claim(s) _____ is/are allowed.
- 7) ☒ Claim(s) 1-20 is/are rejected.
- 8) ☐ Claim(s) _____ is/are objected to.
- 9) ☐ Claim(s) _____ are subject to restriction and/or election requirement

* If any claims have been determined allowable, you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to PPHfeedback@uspto.gov.

Application Papers

- 10) ☐ The specification is objected to by the Examiner.
- 11) ☒ The drawing(s) filed on 02/10/2020 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

- a) ☐ All b) ☐ Some** c) ☐ None of the:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

** See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Information Disclosure Statement(s) (PTO/SB/08a and/or PTO/SB/08b)
Paper No(s)/Mail Date _____.
- 3) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____.
- 4) ☐ Other: _____.

Notice of Pre-AIA or AIA Status

The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

DETAILED ACTION

Double Patenting

1. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the “right to exclude” granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to

be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

1. Claims 1, 2, 8, 10, 11, 15, and 16 are rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1, 2, 9-11, 13, and 16 of U.S. Patent No. 10559306. Although the conflicting claims are not identical, they are not patentably distinct from each other because the above claims of the instant application includes all of the features of the cited claims of U.S. Patent No. 10559306. It would have been obvious to one of ordinary skill in the art to omit the steps of using subsets and relevance since a score/value and device selection implies such activities in a precise way, In re Karlson 136 USPQ 184 (1963): "Omission of an element and its function is an obvious expedient if the remaining elements perform the same functions as before"

Present invention

Conflicting claims

Claims 1, 10, and 16: 1. (Currently Amended) A method, ~~comprising:~~ <u>performed</u> at a first electronic device of a plurality of electronic devices, each ~~electronic device~~ of the plurality of electronic devices comprising one or more microphones, ~~a speaker~~ , one or more processors, and memory storing one or more programs for execution by the one or more processors, the method	Claims 1, 10, and 16: 1. A method, comprising: at a first electronic device of a plurality of electronic devices, each electronic device of the plurality of electronic devices comprising one or more microphones, a speaker, one or more processors, and memory storing one or more programs for

<del>comprising: detecting a voice input from a user; determining a state of the first electronic device and a relevance of the voice input to the first electronic device; identifying a subset of electronic devices from the plurality to which the voice input is relevant; in accordance with a determination that the subset includes the first electronic device; determining a first <u>value score</u> of a criterion associated with the voice input; and receiving second <u>values</u> associated with the voice input ~~scores of the criterion~~ from other electronic devices <u>of the plurality of electronic devices in the subset</u>; and in accordance with a determination that the first <u>value score</u> is higher than the second <u>values</u> ~~scores~~, responding to the detected input.</del> 2. (New) The method of claim 1, further comprising: in accordance with a determination that the first score is not higher than the second scores, forgoing responding to the detected input. 8. (New) The method of claim 1, wherein responding to the detected input comprises	execution by the one or more processors: detecting a voice input; determining a state of the first electronic device and a relevance of the voice input to the first electronic device; identifying a subset of electronic devices from the plurality to which the voice input is relevant; in accordance with a determination that the subset includes the first electronic device; determining a first score of a criterion associated with the voice input; and receiving second scores of the criterion from other electronic devices in the subset; and in accordance with a determination that the first score is higher than the second scores, responding to the detected input. 2. The method of claim 1, further comprising: in accordance with a determination that the first score is not higher than the second scores, foregoing responding to the detected input. 9. The method of claim 1, wherein responding to the detected input

outputting an audible and/or a visual response to the detected voice input.	comprises outputting an audible and/or a visual response to the detected voice input.
11. (New) The first electronic device of claim 10, the one or more programs further comprising instructions for: recognizing a command in the voice input; and in accordance with a determination that a type of the command is related to the first electronic device, responding to the detected voice input.	11. The electronic device of claim 10, wherein the one or more programs further comprise instructions for: in accordance with a determination that a type of the command is related to the first electronic device, responding to the detected voice input.
15. (New) The first electronic device of claim 10, wherein the first electronic device is configured to be awakened by a plurality of affordances including a voice-based affordance; and wherein detecting the voice input is in response to the awakening of the first electronic device.	13. The electronic device of claim 10, wherein the first electronic device is further configured to be awakened by a plurality of affordances including a voice-based affordance; and wherein detecting the voice input is in response to the awakening of the first electronic device.

DETAILED ACTION

Allowable Subject Matter

1. Claims 1-20 allowed.
2. The following is an examiner's statement of reasons for allowance:

After a full review of the complex claims as a whole, the examiner believes that the prior art taken alone or in combination fails to teach the claims as a whole such as a method, performed at a first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more microphones, , one or more processors, and memory storing one or more programs for execution by the one or more processors, the method comprising: detecting a voice input from a user; determining a first value associated with the voice input; receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices ; and in accordance with a determination that the first value is higher than the second values , responding to the detected input.

The above claims are deemed allowable given the complex nature of comparing values from multiple devices for received voice inputs as precisely claimed as a whole in light of the earliest priority date. The closest prior art takes weights from devices unconditionally, meaning that if there are X device present then the system takes X weights to determine the highest among such devices, whereas assuming that weights

are a value metric, the present invention has X devices and takes a first weight and compares the first value to X second values, in which case it must be higher. The highest value is not selected per se but the first weight must be higher than the second weight(s), in other words there is a first value being compared, unlike the prior art which at best uses a first value unrelated to the voice input (e.g. threshold) to compare to the second values. Therefore the prior art in light of the priority date fails to teach or suggest an obvious modification that would result in the present claims.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Pertinent:

Rosenberger; Theodore Alfred	US 8340975 B1	USPAT	20121225
Bhagwan; Varun	US 20190304443 A1	US-PGPUB	20191003
Ganong, III; William F. et al.	US 20140278435 A1	US-PGPUB	20140918
Basye; Kenneth John et al.	US 20140163978 A1	US-PGPUB	20140612
KIM; Yongsin et al.	US 20130238326 A1	US-PGPUB	20130912

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to MICHAEL C COLUCCI whose telephone number is (571)270-1847. The examiner can normally be reached on M-F 9 AM - 5 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Richemond Dorvil can be reached on 571-272-7602. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/MICHAEL COLUCCI/
Primary Examiner, Art Unit 2656

(571)-270-1847
Examiner FAX: (571)-270-2847
Michael.Colucci@uspto.gov

Notice of References Cited	Application/Control No. 16/786,943		Applicant(s)/Patent Under Reexamination Mixer et al.	
	Examiner MICHAEL COLUCCI		Art Unit 2656	Page 1 of 2

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-20080288256-A1	11-2008	AGAPI; CIPRIAN	G10L13/04	704/260
*	B	US-20050171766-A1	08-2005	Albesano, Dario	G10L15/16	704/202
*	C	US-5455889-A	10-1995	Bahl; Lalit R.	G10L15/142	704/236
*	D	US-20040158464-A1	08-2004	Baker, James K.	G10L15/083	704/231
*	E	US-20140163978-A1	06-2014	Basye; Kenneth John	G10L15/28	704/233
*	F	US-20190304443-A1	10-2019	Bhagwan; Varun	G10L15/22	1/1
*	G	US-20140278435-A1	09-2014	Ganong, III; William F.	G10L15/22	704/275
*	H	US-20180196776-A1	07-2018	Hershko; Yuval Corey	G06F13/4291	1/1
*	I	US-20180182390-A1	06-2018	Hughes; Christopher Thaddeus	G10L15/08	1/1
*	J	US-20150032443-A1	01-2015	Karov; Yael	G06F16/36	704/9
*	K	US-20150248885-A1	09-2015	Koulomzin; Daniel G.	G10L15/08	704/251
*	L	US-20110246181-A1	10-2011	Liang; Jisheng	G06F40/289	704/9
*	M	US-20180061419-A1	03-2018	Melendo Casado; Diego	G10L15/30	1/1

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)

Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

<i>Notice of References Cited</i>	Application/Control No. 16/786,943		Applicant(s)/Patent Under Reexamination Mixer et al.	
	Examiner MICHAEL COLUCCI		Art Unit 2656	Page 2 of 2

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-6961704-B1	11-2005	Phillips; Michael S.	G10L13/10	704/267
*	B	US-8244534-B2	08-2012	Qian; Yao	G10L13/06	704/256.3
*	C	US-7778816-B2	08-2010	Reynar; Jeff	G06K9/2054	704/9
*	D	US-8340975-B1	12-2012	Rosenberger; Theodore Alfred	G10L15/22	704/275
*	E	US-20160012465-A1	01-2016	Sharp; Jeffrey A.	G06Q20/384	705/14.17
*	F	US-20040236570-A1	11-2004	Tato, Raquel	G10L15/02	704/206
*	G	US-20040078191-A1	04-2004	Tian, Jilei	G06F40/263	704/9
*	H	US-20130238326-A1	09-2013	KIM; Yongsin	G10L15/22	704/231
	I					
	J					
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Electronically filed November 25, 2020

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of:	Kenneth Mixter et al.	Confirmation No.:	5709
Serial No.:	16/786,943	Art Unit:	2656
Filed:	February 10, 2020	Examiner:	Colucci, Michael C.
For:	<i>Device Leadership Negotiation Among Voice Interface Devices</i>	Atty. Docket No.:	060963-7419-US

RESPONSE TO OFFICE ACTION

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

The enclosed Amendment is in response to the Office Action dated November 5, 2020 for the above identified patent application.

The Commissioner is hereby authorized to charge any required fee(s) to Morgan, Lewis & Bockius LLP Deposit Account No. 50-0310 (order no. 060963-7419-US).

IN THE CLAIMS:

Rewrite the pending claims and add new claims as follows:

1. (Previously Presented) A method performed at a first electronic device of a plurality of electronic devices, each of the plurality of electronic devices comprising one or more microphones, one or more processors, and memory storing one or more programs for execution by the one or more processors, the method comprising:
 - detecting a voice input from a user;
 - determining a first value associated with the voice input;
 - receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices; and
 - in accordance with a determination that the first value is higher than the second values, responding to the detected input.
2. (Currently Amended) The method of claim 1, further comprising:
 - in accordance with a determination that the first value score ~~score~~ is not higher than the second scores, forgoing responding to the detected input.
3. (Previously Presented) The method of claim 2, wherein forgoing responding to the detected input further comprises changing a state of operation of the first electronic device.
4. (Previously Presented) The method of claim 1, further comprising:
 - in accordance with a determination that the first value is not higher than the second values, identifying a criterion associated with the voice input; and
 - in accordance with a determination that the criterion is the most relevant to the first electronic device, responding to the detected input.
5. (Previously Presented) The method of claim 4, further comprising:
 - determining a state of the first electronic device;
 - wherein the determination that the criterion is the most relevant to the first electronic device is based on the state of the electronic device.
6. (Previously Presented) The method of claim 4, wherein the criterion comprises one or more of: a type of request, a command in the voice input, and a state of each electronic device of the plurality of electronic devices.

7. (Previously Presented) The method of claim 1, further comprising:
at the first electronic device:
detecting subsequent voice inputs from the user;
pre-processing the subsequent voice inputs; and
transmitting the pre-processed voice inputs to a server system;
wherein the other electronic devices forgo the detecting, pre-processing, and transmitting.
8. (Previously Presented) The method of claim 1, wherein responding to the detected input comprises outputting an audible and/or a visual response to the detected voice input.
9. (Previously Presented) The method of claim 1, wherein the first electronic device further comprises a speaker, and responding to the detected input further comprises outputting an audible response.
10. (Currently Amended) A first electronic device of a plurality of electronic devices, the first electronic device ~~each of the plurality of electronic devices~~ comprising:
one or more microphones;
one or more processors; and
memory storing one or more programs to be executed by the one or more processors,
the one or more programs comprising instructions for:
detecting a voice input from a user;
determining a first value associated with the voice input;
receiving second values associated with the voice input from other electronic devices of the plurality of electronic devices; and
in accordance with a determination that the first value is higher than the second values, responding to the detected input.
11. (Previously Presented) The first electronic device of claim 10, the one or more programs further comprising instructions for:
recognizing a command in the voice input; and
in accordance with a determination that a type of the command is related to the first electronic device, responding to the detected voice input.
12. (Previously Presented) The first electronic device of claim 10, the one or more programs further comprising instructions for:

communicating to the other devices of the plurality of electronic devices to forgo responding to the detected output.

13. (Previously Presented) The first electronic device of claim 10, wherein the instruction for detecting the voice input is performed at intervals of a predefined number of voice inputs.

14. (Previously Presented) The first electronic device of claim 10, wherein:
the plurality of electronic devices is communicatively coupled through a local network; and
the receiving is performed through the local network.

15. (Previously Presented) The first electronic device of claim 10, wherein the first electronic device is configured to be awakened by a plurality of affordances including a voice-based affordance; and
wherein detecting the voice input is in response to the awakening of the first electronic device.

16. (Previously Presented) A system comprising a plurality of electronic devices, each of the electronic devices including one or more microphones, one or more processors, and memory storing one or more programs for execution by the one or more processors, each of the electronic devices programmed to:

detect a voice input;
determine a first value associated with the detected voice input;
receive respective second values from other electronic devices of the plurality of electronic devices;
compare the first value and the respective second values; and
in accordance with a determination that the first value is higher than the respective second values, respond to the voice input.

17. (Previously Presented) The system of claim 16, wherein the first value comprises a first quality score of the voice input.

18. (Previously Presented) The system of claim 17, wherein the first quality score comprises a confidence level of detection of the voice input.

19. (Previously Presented) The system of claim 17, wherein the first quality score comprises a signal-to-noise rating of detection of the voice input.

20. (Currently Amended) The system of claim 16, wherein each of the respective second values is indicative of a quality of the voice input detected by a respective one of the other electronic devices.

REMARKS

This amendment responds to the office action mailed November 5, 2020. In the office action the Examiner:

- rejected claims 1, 2, 8, 10, 11, 15, and 16 on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1, 2, 9-11, 13, and 16 of U.S. Patent No. 10,559,306; and
- allowed claims 1-20.

AMENDMENTS TO THE CLAIMS

Claims 2, 10, and 20 have been amended for clarity and to correct for informalities. No new matter has been added.

After entry of this amendment, the pending claims are: claims 1-20.

REMARKS CONCERNING DOUBLE PATENTING REJECTIONS

I. REJECTION OF CLAIMS 1, 2, 8, 10, 11, 15, AND 16 ON THE GROUND OF NONSTATUTORY OBVIOUSNESS-TYPE DOUBLE PATENTING AS BEING UNPATENTABLE OVER CLAIMS 1, 2, 9-11, 13, AND 16 OF U.S. PATENT NO. 10,559,306

The Applicant is concurrently submitting with this Response an electronic terminal disclaimer over U.S. Patent No. 10,559,306 to address the double patenting rejection.

REMARKS CONCERNING ALLOWABLE SUBJECT MATTER

II. ALLOWANCE OF CLAIMS 1-20

The Applicant thanks the examiner for noting the allowability of claims 1-20. As noted above, Applicant is concurrently submitting with this Response an electronic terminal disclaimer over U.S. Patent No. 10,559,306 to address the double patenting rejection. Thus, Applicant respectfully requests that the Examiner regard claims 1-20 with a view toward allowance.

CONCLUDING REMARKS

By responding in the foregoing remarks only to particular positions asserted by the Examiner, the Applicants do not necessarily acquiesce in other positions that have not been explicitly addressed. In addition, the Applicants' arguments for the patentability of a claim should not be understood as implying that no other reasons for the patentability of that claim exist.

In light of the above amendments and remarks, the Applicants respectfully request that the Examiner reconsider this application with a view towards allowance. The Examiner is invited to call the undersigned attorney at (650) 843-4000, if a telephone call could help resolve any remaining items.

Respectfully submitted,

Date: November 25, 2020

/Douglas J. Crisman/

39,951

Douglas J. Crisman

(Reg. No.)

MORGAN, LEWIS & BOCKIUS LLP

1400 Page Mill Road

Palo Alto, CA 94304

(650) 843-4000

Exhibit 13



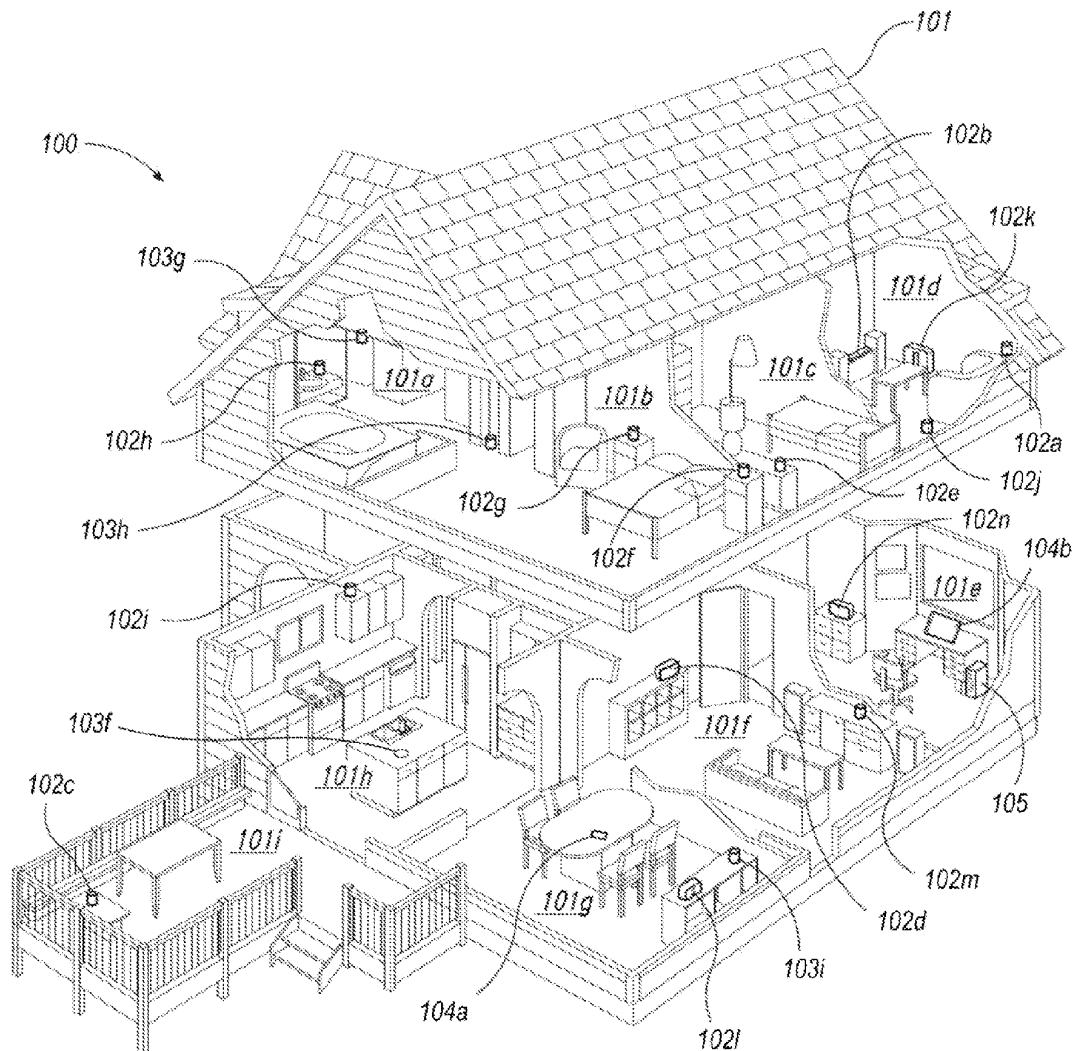
US 20220244905A1

(19) **United States**(12) **Patent Application Publication**
Kannan(10) **Pub. No.: US 2022/0244905 A1**(43) **Pub. Date: Aug. 4, 2022**(54) **PLAYBACK SESSION TRANSITIONS
ACROSS DIFFERENT PLATFORMS****G10L 25/51** (2006.01)**G05B 15/02** (2006.01)(71) Applicant: **Sonos, Inc.**, Santa Barbara, CA (US)(72) Inventor: **Dinesh Kannan**, Seattle, WA (US)(21) Appl. No.: **17/524,579**(22) Filed: **Nov. 11, 2021****Related U.S. Application Data**(63) Continuation of application No. 16/745,910, filed on
Jan. 17, 2020, now Pat. No. 11,175,883.**Publication Classification**(51) **Int. Cl.****G06F 3/16** (2006.01)**H04L 65/60** (2006.01)(52) **U.S. Cl.**CPC **G06F 3/165** (2013.01); **H04L 65/60**
(2013.01); **H04W 84/12** (2013.01); **G05B**
15/02 (2013.01); **G06F 3/162** (2013.01); **G10L**
25/51 (2013.01)

(57)

ABSTRACT

As noted above, example techniques relate to playback transitions between platforms. In an example implementation, while a first playback device is playing back audio content during a first playback session in a first domain, a mobile device detecting a first event configured as a trigger to transfer playback to a second domain on a second playback device. In response to detecting the first event, the mobile device determining playback information for the playback session and sends, via a network interface, instructions to initiate a playback session in the second domain to reproduce the first playback session in the first domain.



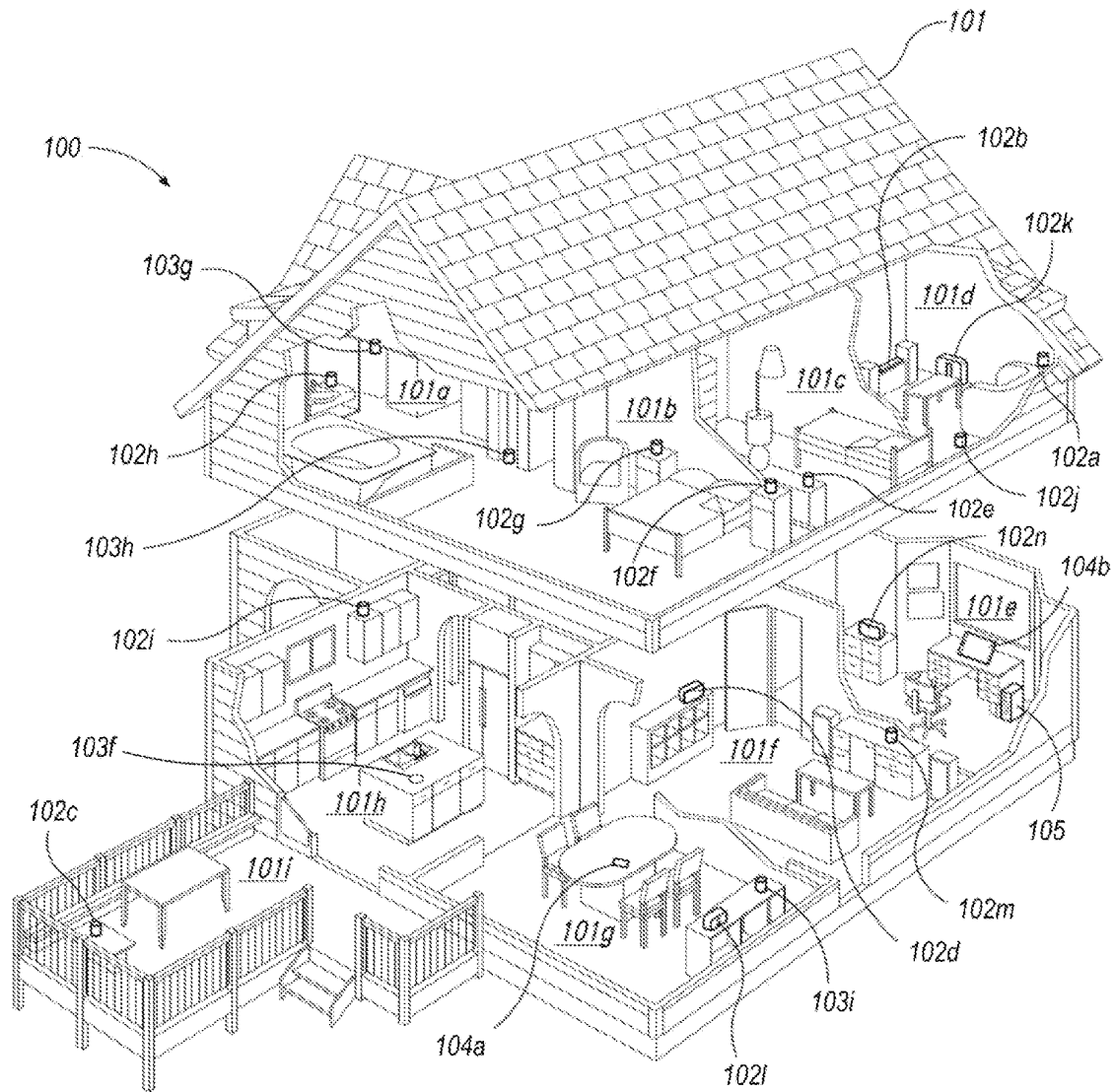


Fig. 1A

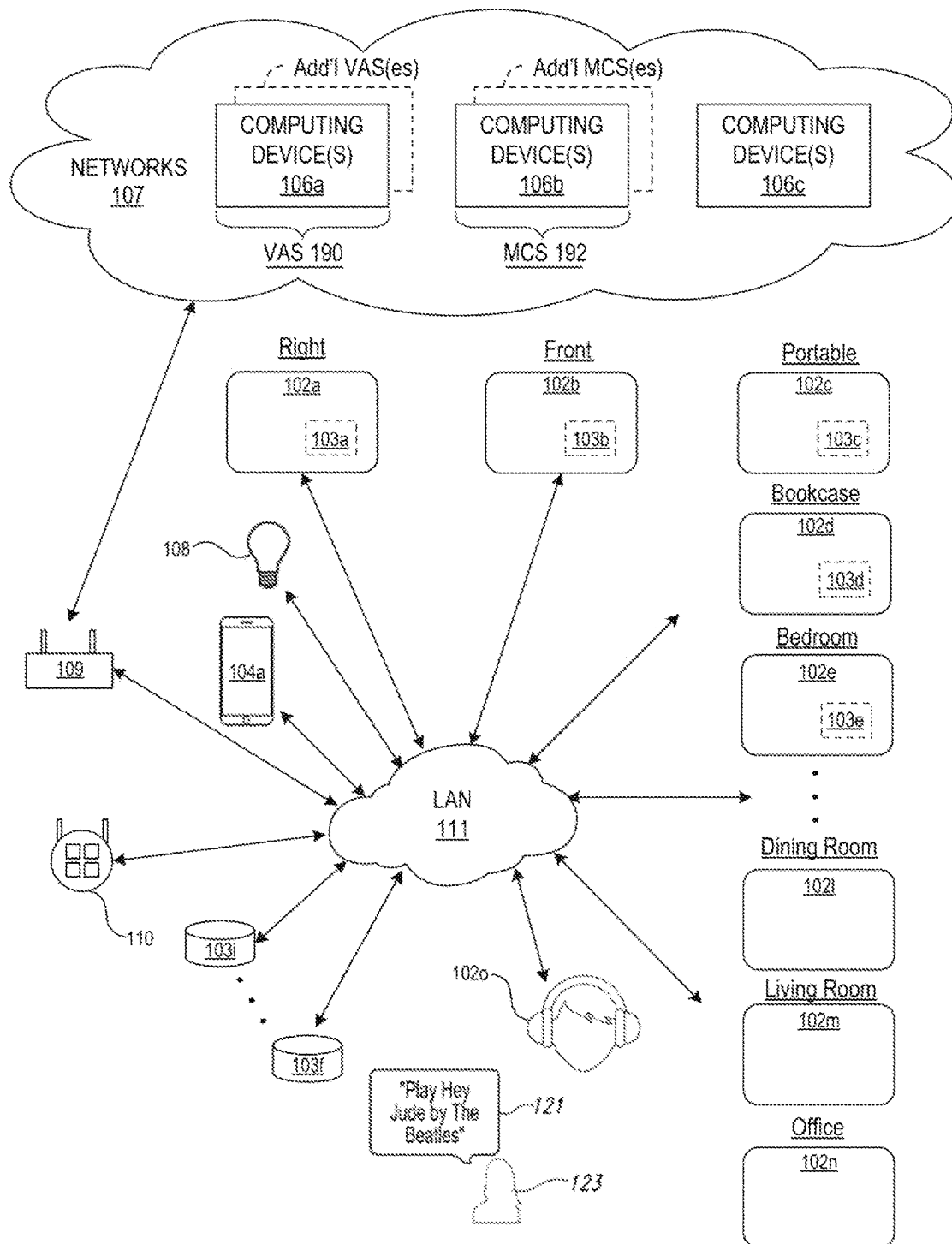


Fig. 1B

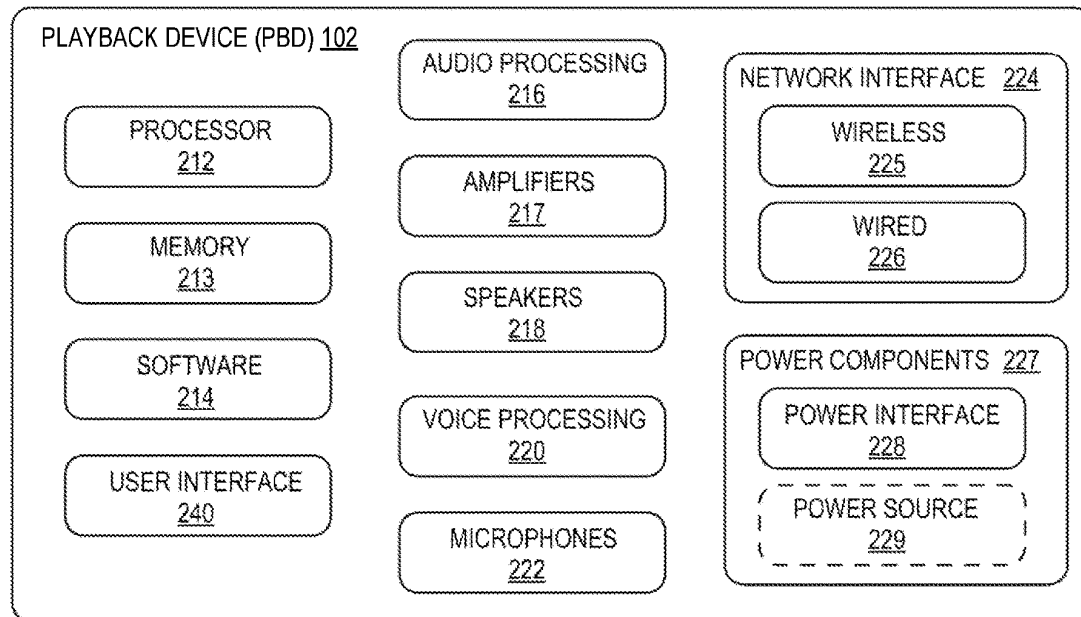


Fig. 2A

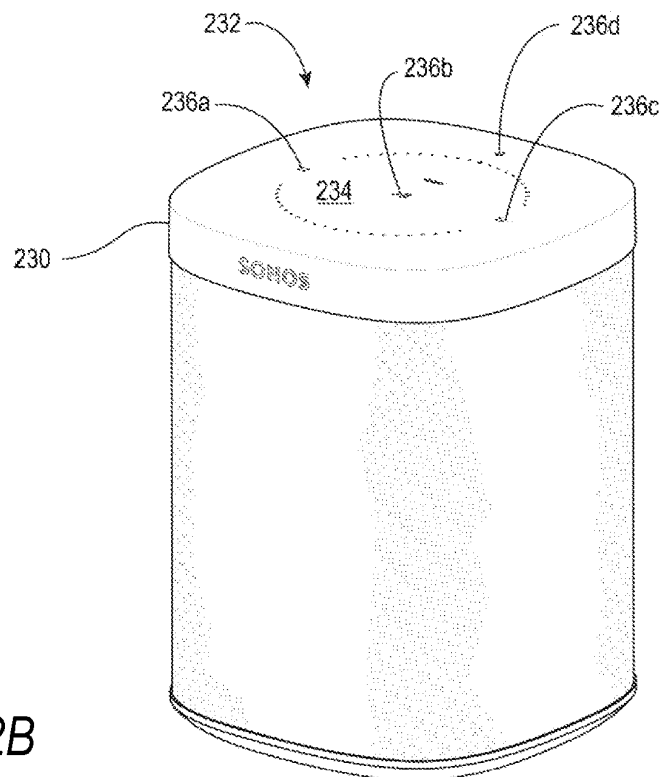


Fig. 2B

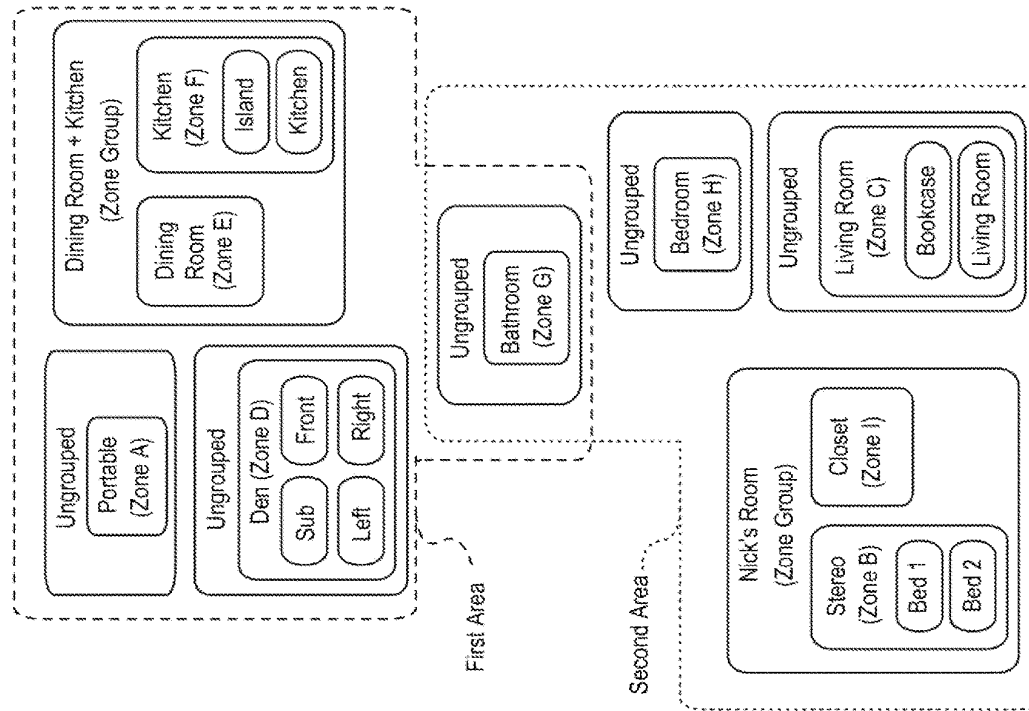


Fig. 3A

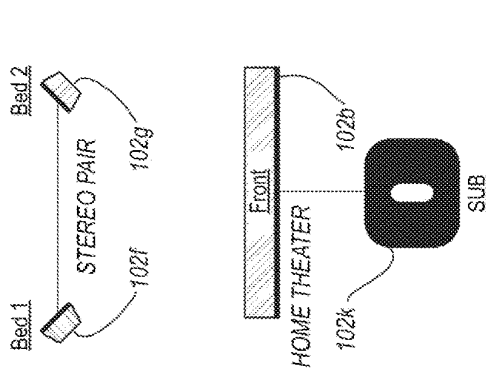


Fig. 3B

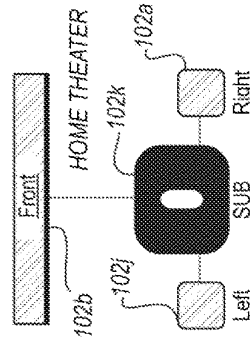


Fig. 3C

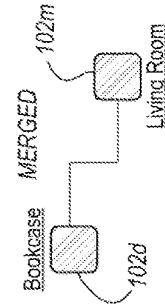


Fig. 3D

Fig. 3E

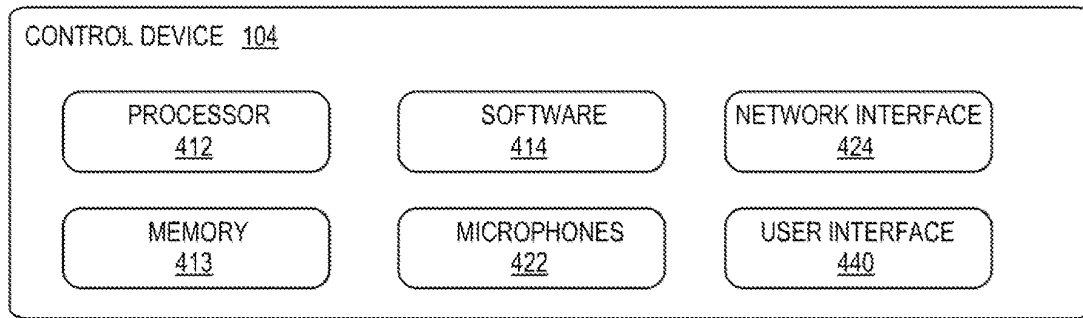


Fig. 4

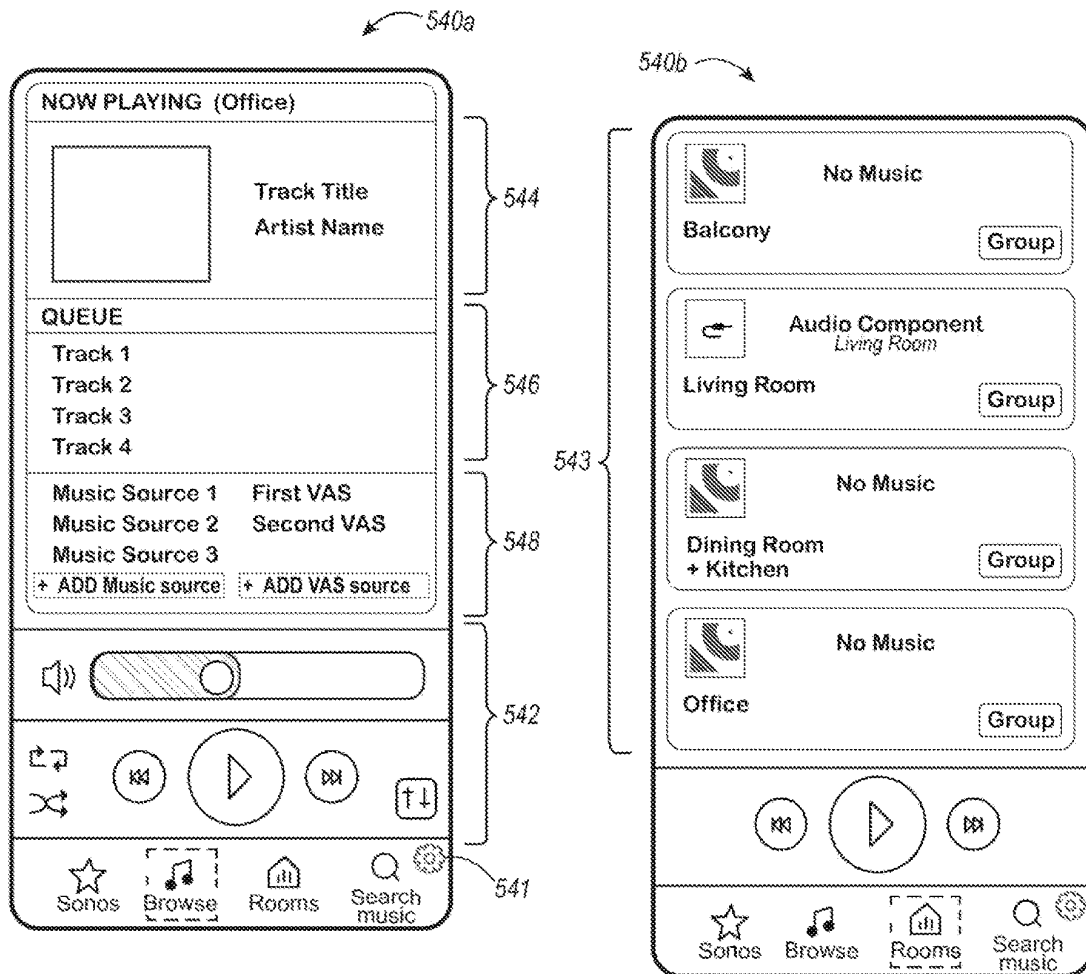


Fig. 5A

Fig. 5B

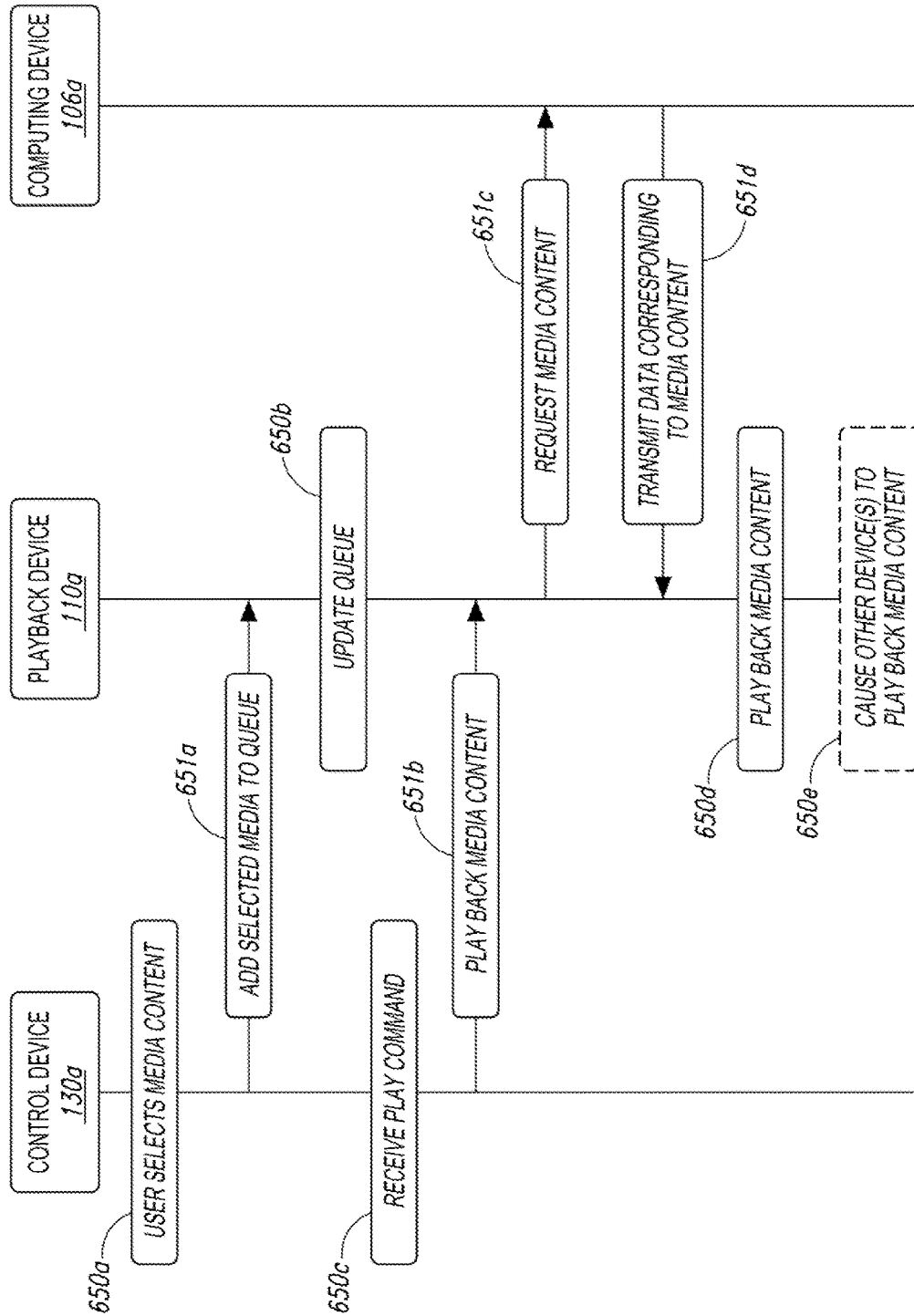


Fig. 6

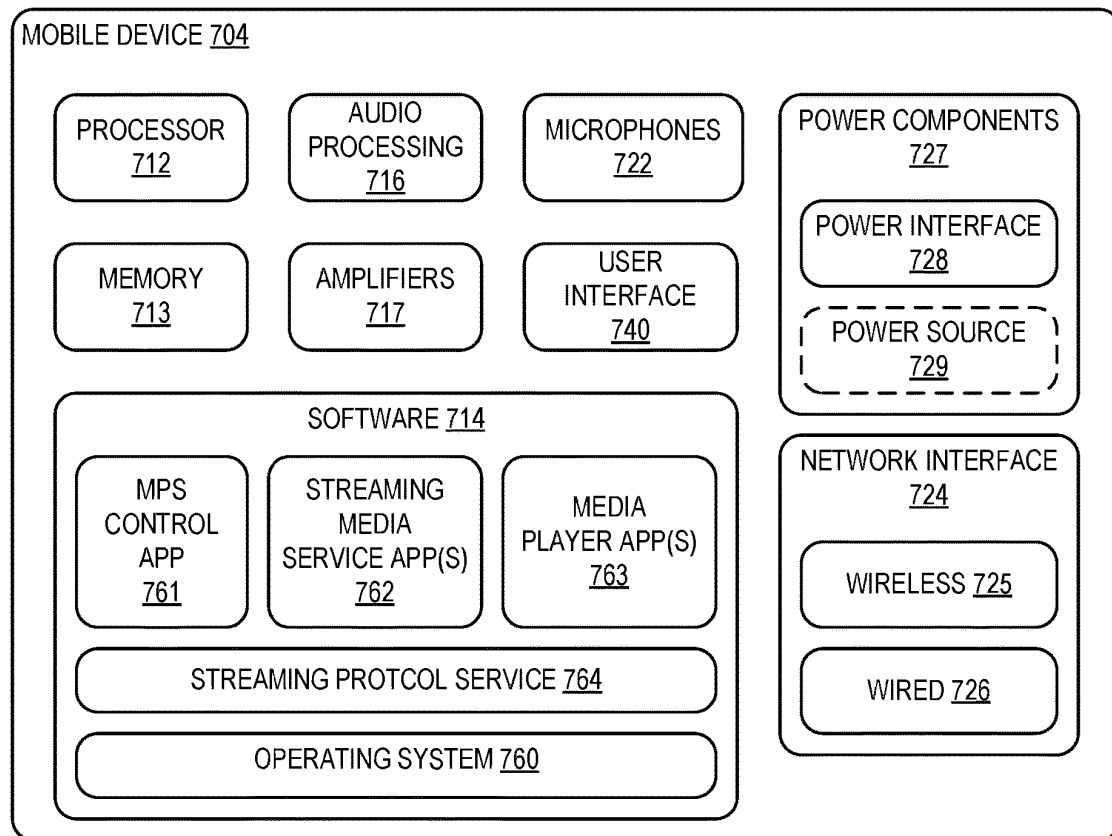


Fig. 7A

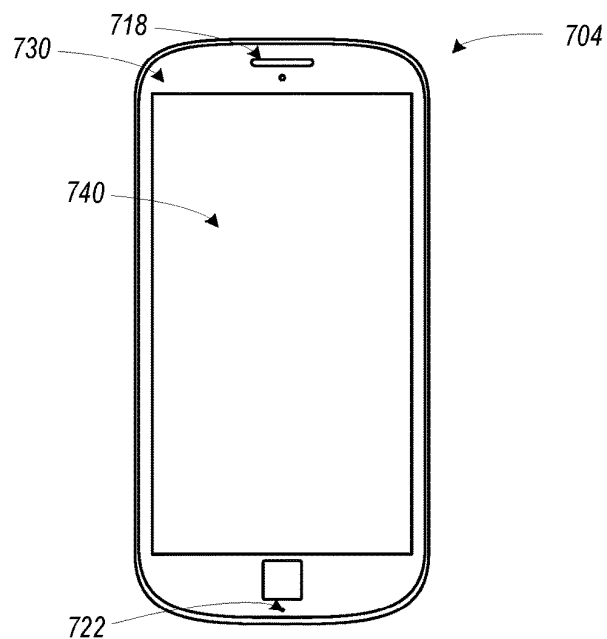


Fig. 7B

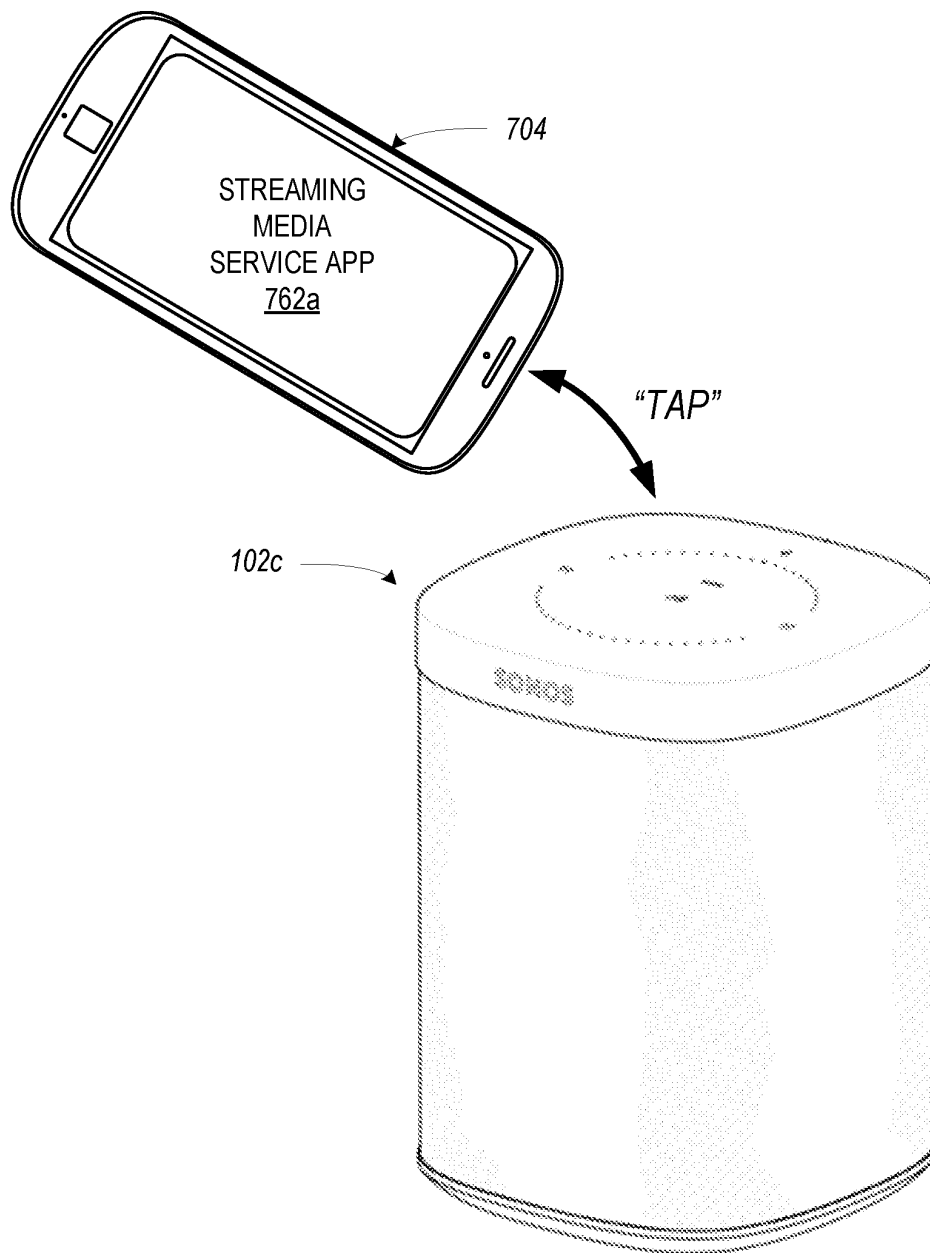


Fig. 8A

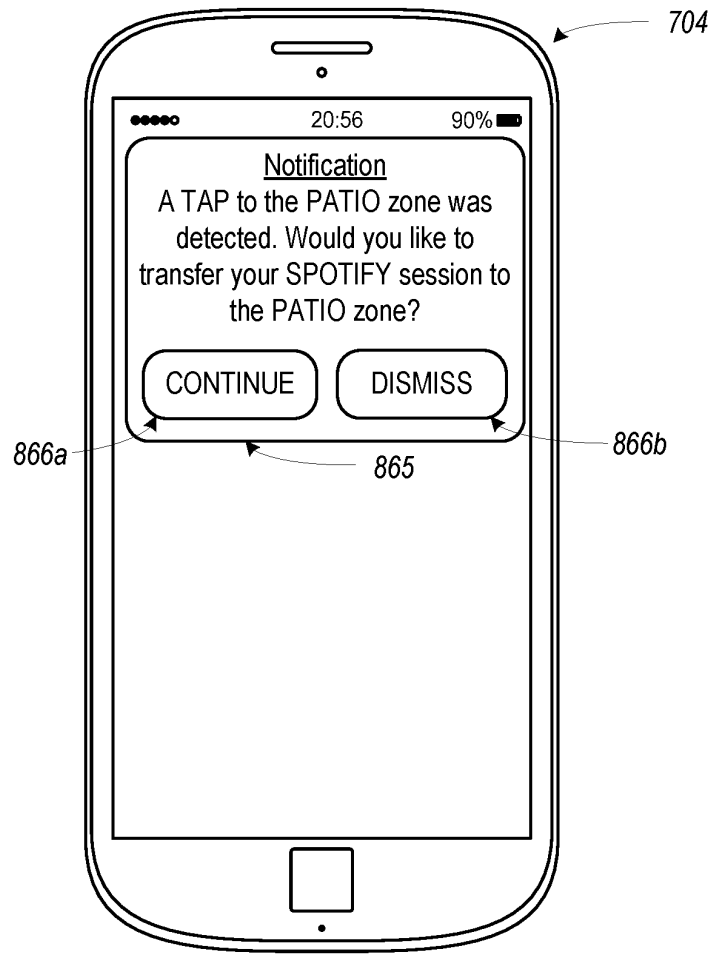


Fig. 8B

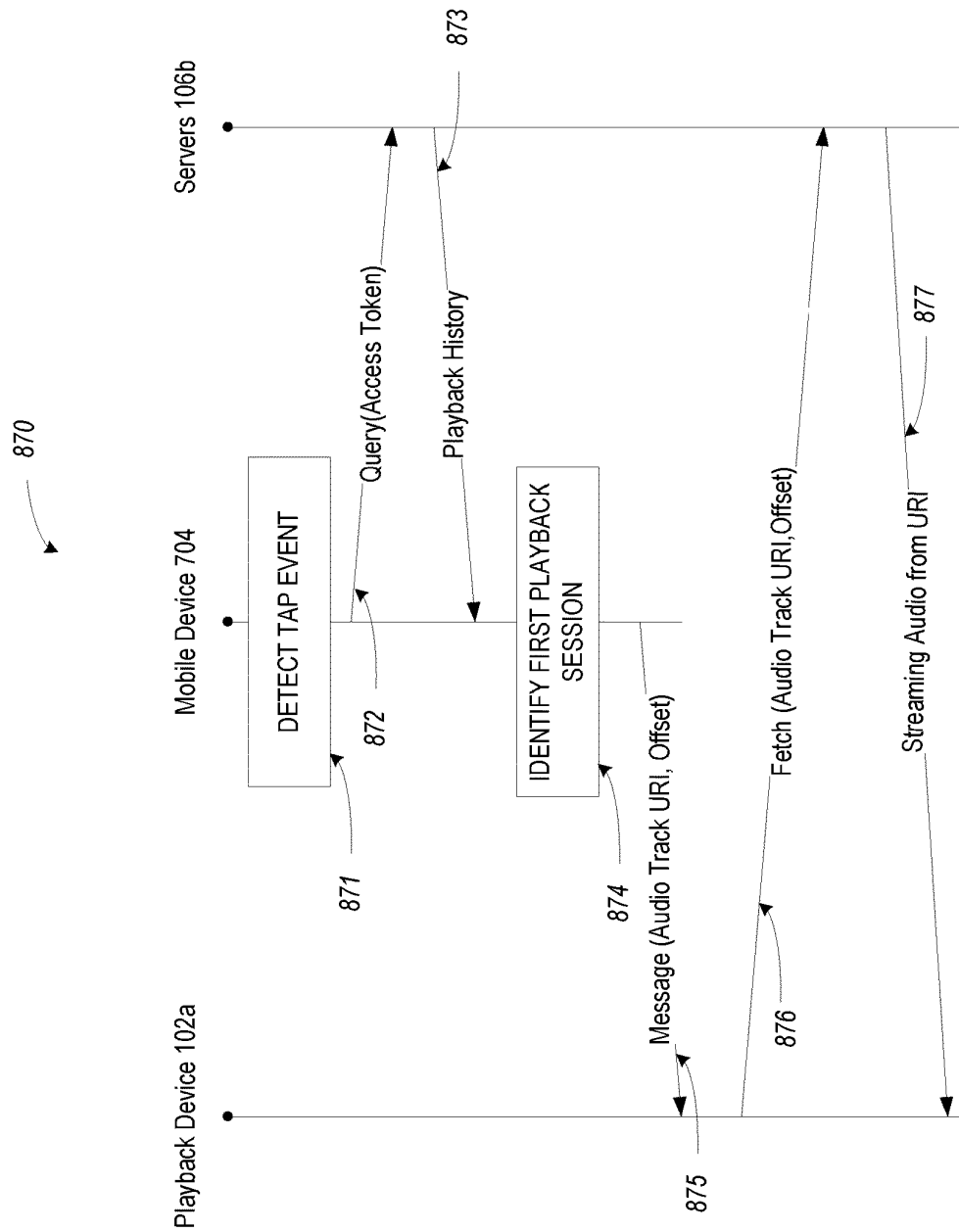
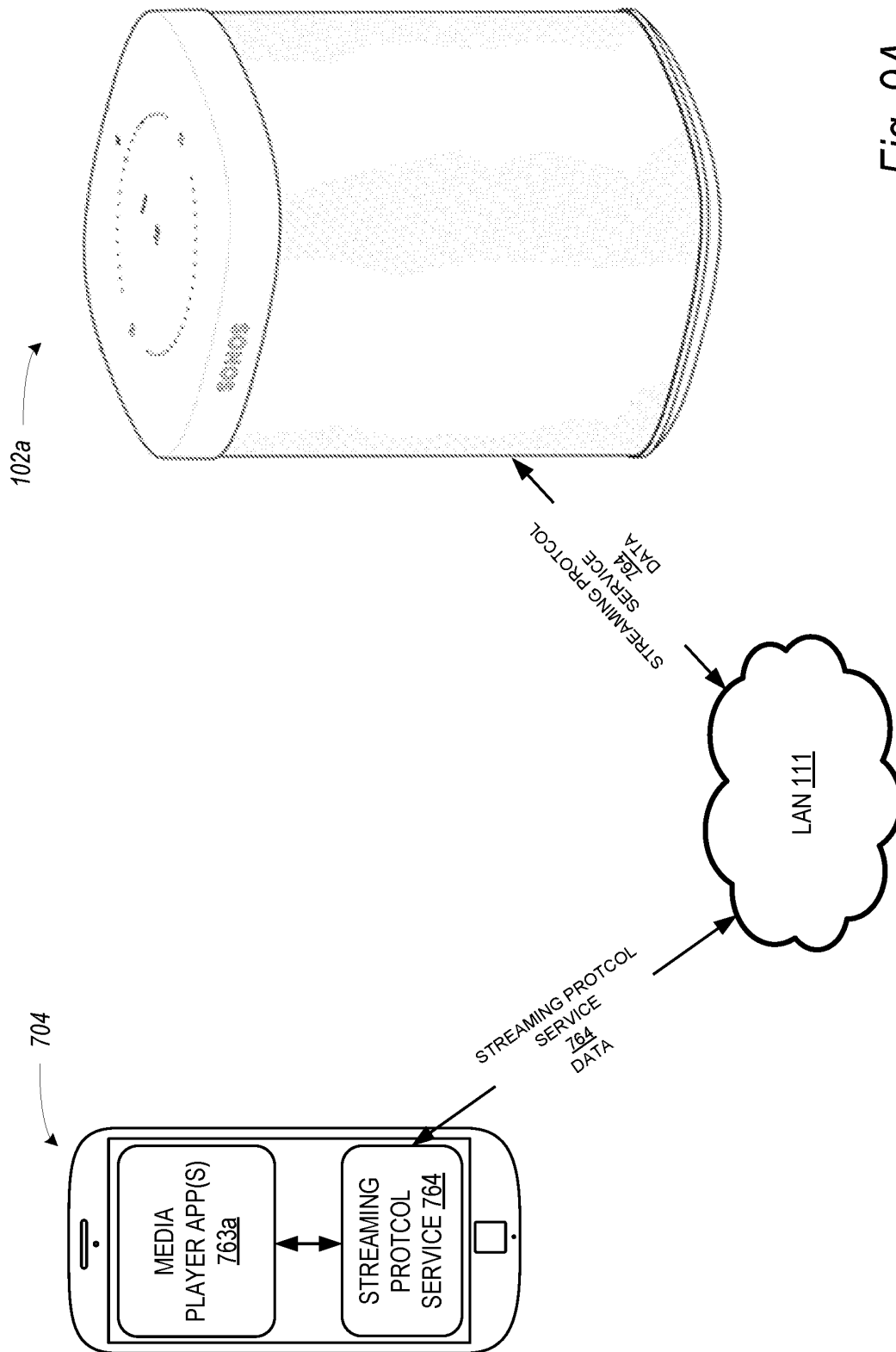


Fig. 8C



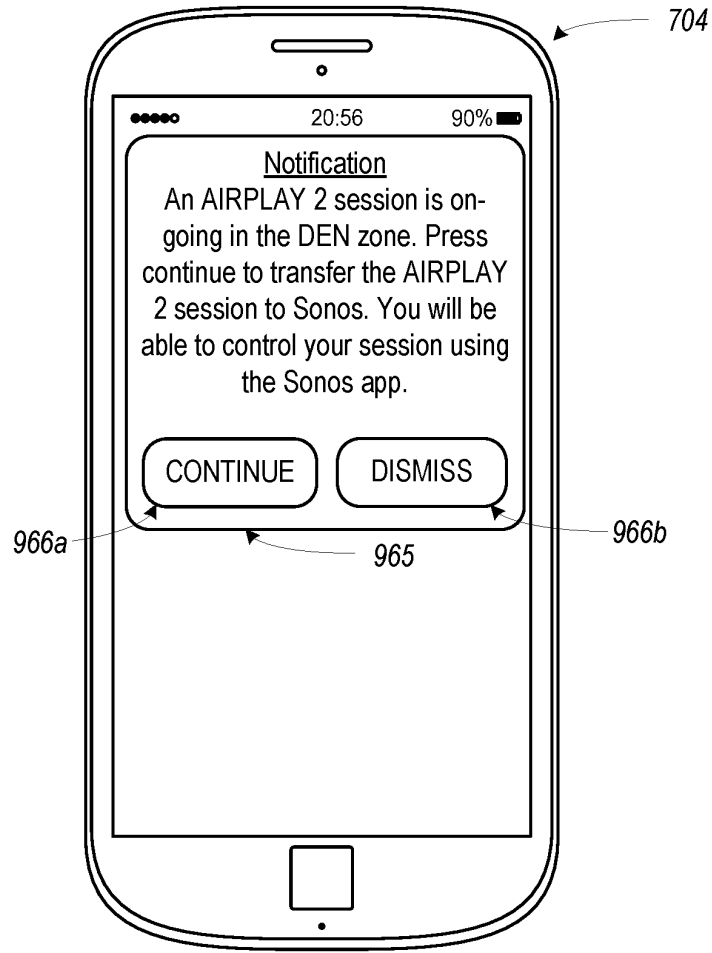


Fig. 9B

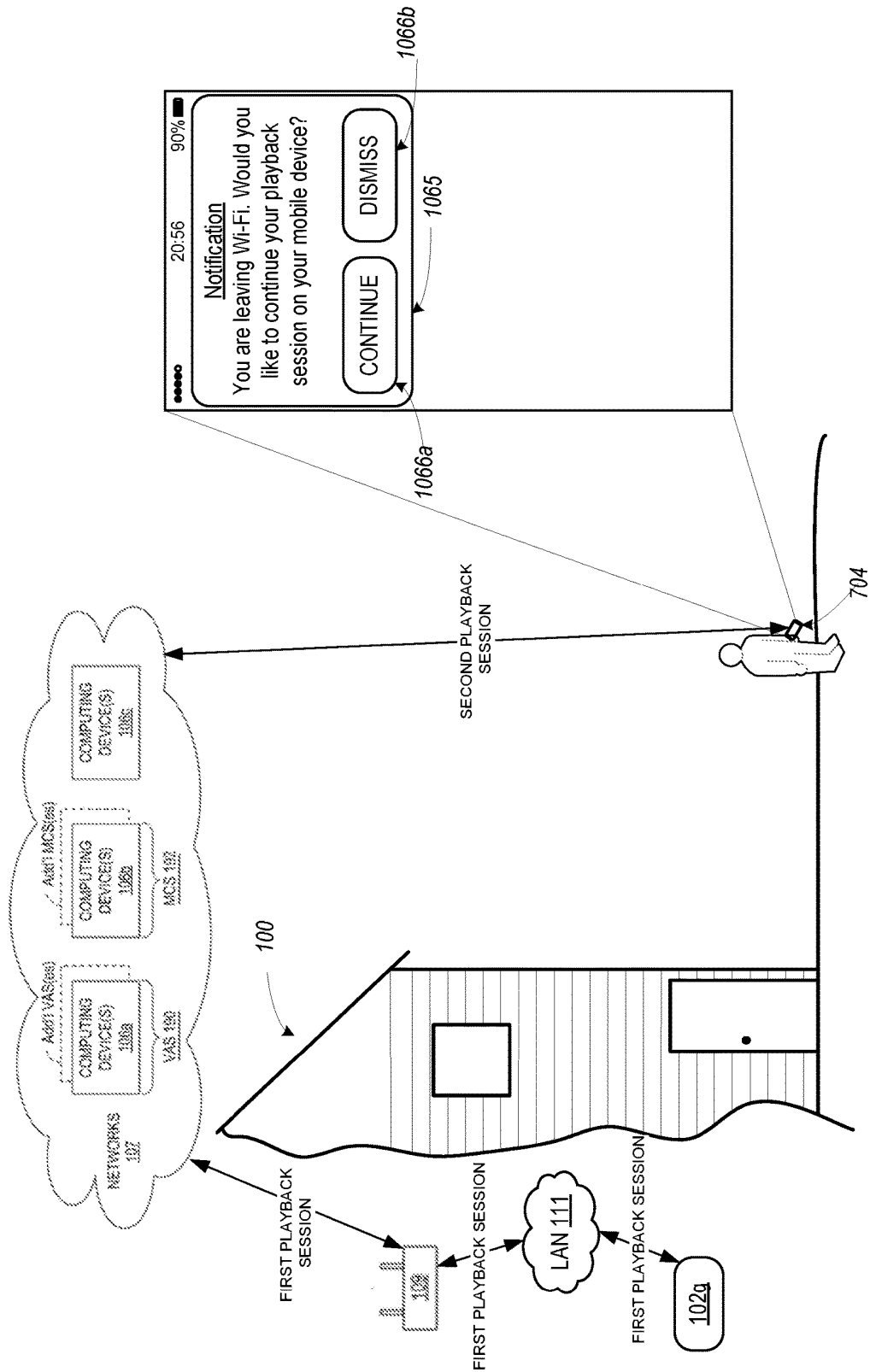


Fig. 10

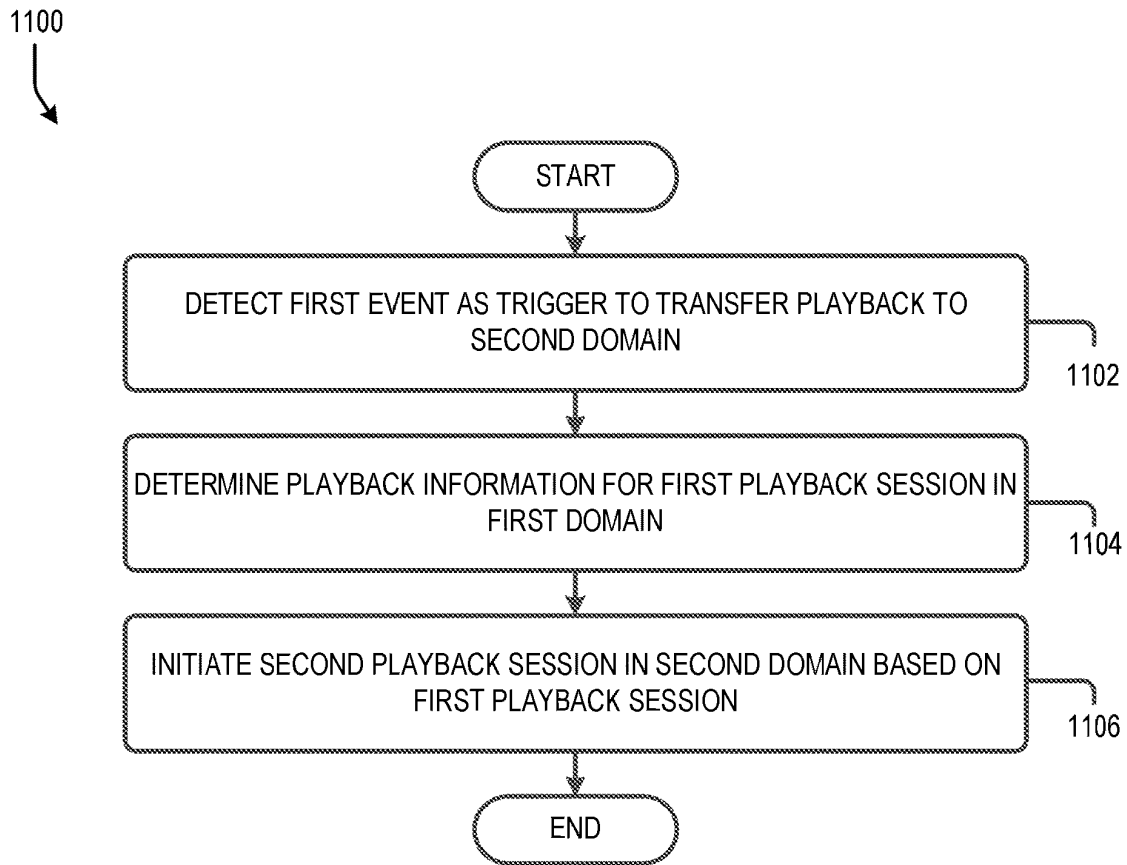


Fig. 11

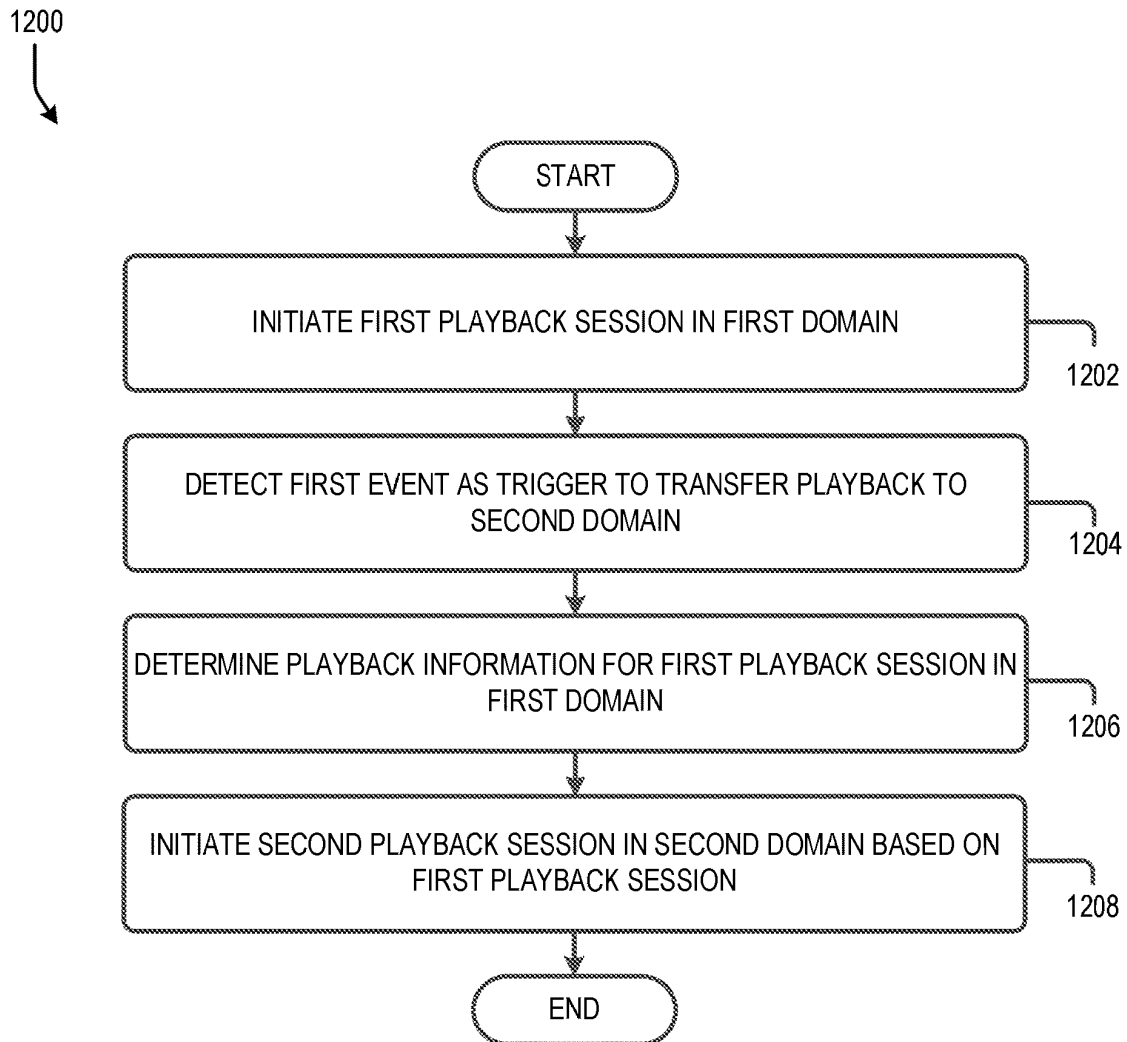


Fig. 12

PLAYBACK SESSION TRANSITIONS ACROSS DIFFERENT PLATFORMS

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority under 35 U.S.C. § 120 to, and is a continuation of, U.S. non-provisional patent application Ser. No. 16/745,910, filed on Jan. 17, 2020, and entitled “Playback Session Transitions Across Different Platforms,” which is incorporated herein by reference in its entirety.

FIELD OF THE DISCLOSURE

[0002] The present technology relates to consumer goods and, more particularly, to methods, systems, products, features, services, and other elements directed to voice-assisted control of media playback systems or some aspect thereof.

BACKGROUND

[0003] Options for accessing and listening to digital audio in an out-loud setting were limited until in 2002, when SONOS, Inc. began development of a new type of playback system. Sonos then filed one of its first patent applications in 2003, entitled “Method for Synchronizing Audio Playback between Multiple Networked Devices,” and began offering its first media playback systems for sale in 2005. The Sonos Wireless Home Sound System enables people to experience music from many sources via one or more networked playback devices. Through a software control application installed on a controller (e.g., smartphone, tablet, computer, voice input device), one can play what she wants in any room having a networked playback device. Media content (e.g., songs, podcasts, video sound) can be streamed to playback devices such that each room with a playback device can play back corresponding different media content. In addition, rooms can be grouped together for synchronous playback of the same media content, and/or the same media content can be heard in all rooms synchronously.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] Features, aspects, and advantages of the presently disclosed technology may be better understood with regard to the following description, appended claims, and accompanying drawings where:

[0005] Features, aspects, and advantages of the presently disclosed technology may be better understood with regard to the following description, appended claims, and accompanying drawings, as listed below. A person skilled in the relevant art will understand that the features shown in the drawings are for purposes of illustrations, and variations, including different and/or additional features and arrangements thereof, are possible.

[0006] FIG. 1A is a partial cutaway view of an environment having a media playback system configured in accordance with aspects of the disclosed technology.

[0007] FIG. 1B is a schematic diagram of the media playback system of FIG. 1A and one or more networks.

[0008] FIG. 2A is a functional block diagram of an example playback device.

[0009] FIG. 2B is an isometric diagram of an example housing of the playback device of FIG. 2A.

[0010] FIGS. 3A, 3B, 3C, 3D and 3E are diagrams showing example playback device configurations in accordance with aspects of the disclosure.

[0011] FIG. 4 is a functional block diagram of an example controller device in accordance with aspects of the disclosure.

[0012] FIGS. 5A and 5B are controller interfaces in accordance with aspects of the disclosure.

[0013] FIG. 6 is a message flow diagram of a media playback system.

[0014] FIG. 7A is a functional block diagram of an example mobile device, in accordance with aspects of the disclosure.

[0015] FIG. 7B is a diagram of the example mobile device, in accordance with aspects of the disclosure.

[0016] FIG. 8A is a schematic diagram illustrating a tap event between the example mobile device and the example playback device, in accordance with aspects of the disclosure.

[0017] FIG. 8B illustrates an example notification on the example mobile device, in accordance with aspects of the disclosure.

[0018] FIG. 8C is a schematic diagram illustrating an example playback session transition, in accordance with aspects of the disclosure.

[0019] FIG. 9A is a schematic diagram illustrating a streaming protocol service session, in accordance with aspects of the disclosure.

[0020] FIG. 9B illustrates an example notification on the example mobile device, in accordance with aspects of the disclosure.

[0021] FIG. 10 is a schematic diagram illustrating an example playback session transition, in accordance with aspects of the disclosure.

[0022] FIG. 11 is a flow diagram of an example method to perform a playback session transition in accordance with aspects of the disclosure.

[0023] FIG. 12 is a flow diagram of an example method to perform a playback session transition in accordance with aspects of the disclosure.

[0024] The drawings are for purposes of illustrating example embodiments, but it should be understood that the inventions are not limited to the arrangements and instrumentality shown in the drawings. In the drawings, identical reference numbers identify at least generally similar elements. To facilitate the discussion of any particular element, the most significant digit or digits of any reference number refers to the Figure in which that element is first introduced. For example, element 103a is first introduced and discussed with reference to FIG. 1A.

DETAILED DESCRIPTION

I. Overview

[0025] Example system and techniques described herein relate to transitioning a playback session between different platforms. Various entities, such as playback device manufacturers, mobile device manufacturers, and streaming audio services, may each establish respective platforms or “domains” for utilizing their products or services. To provide options to users for control, a wireless “smart” speaker, such as a SONOS playback device, may support playback control using multiple platforms. For instance, a user of a playback device may be able to control the playback device

remotely using a native control application (“app”) on their smartphone or computer, which utilizes a native protocol. The user may also be able to control their playback device using a protocol integrated into their mobile phone or computer (e.g., APPLE AIRPLAY). Further, the user may additionally be able to control the playback device using a streaming media service app utilizing a streaming media service protocol (e.g., via the SPOTIFY app using SPOTIFY CONNECT).

[0026] One benefit to these platforms is that they enable interoperability between devices that support the platform, such as the ability to transition a playback session between devices. For instance, a user may start a playback session on a first playback device (e.g., their mobile phone or a first “smart” speaker) in a first domain using a given app on their mobile device. After starting the playback session on the first playback device, the user may transition that playback session to a second playback device (e.g., a second “smart speaker”) by selecting the second playback device in the app. This allows the user to move the music or other audio content between rooms of their home (having respective “smart” speakers”) or to move the music from and to their mobile device (e.g., when leaving their home). The protocol provided by the platform supports the control messaging necessary to enable such transitions.

[0027] One issue with these platforms is that they are not generally compatible with one another. Such incompatibility may be inherent as different platforms generally implement similar features in different ways, using different communications protocols and application programming interfaces (APIs). Yet, some users may benefit from interoperability between playback devices of different domains, including the ability to transition playback sessions between domains.

[0028] Since a playback device of the native domain is not natively compatible with a playback device of an extrinsic domain, a playback session is not inherently able to be transitioned from a first domain on a first playback device (e.g., a wireless smart speaker) to a second domain on a second playback device (e.g., a mobile device). As such, exemplary techniques to transition playback sessions may involve stopping a first playback session in a source domain on the first playback device and recreating the first session in a target domain as a second session on the second playback device. In this manner, the playback session may appear from the perspective of a user to have transitioned between the first and second playback devices and the source and target domains.

[0029] While, in some cases, a user may be able to manually stop a first playback session in a source domain on a first playback device and recreate the first session in the target domain as a second session on a second playback device, such a process is generally inconvenient. The user would have to make note of the currently playing audio track and play position in the source domain, and then find that audio track and play position in the target domain to start a playback session. If the user is playing a playlist in the source domain, recreating that playlist in the target domain may also be cumbersome.

[0030] Accordingly, example techniques described herein may involve automating one or more aspects of a playback session transition. For instance, the example techniques may involve querying, determining, or otherwise obtaining the current state of the first playback session in the source domain on the first playback device, which may include a

currently playing audio track and play position within the audio track, as well as a playlist or queue that is currently being played back. The process for obtaining the current state of the first playback session in the first domain may vary based on the particular domain that is acting as the source, as different domains may make playback information available in different manners.

[0031] Further, the example techniques may further involve automating one or more aspects of a playback session transition by initiating the playback session transition based on a trigger condition. Various trigger conditions, such as a change in location (e.g., from at-home to on-the-go) or device proximity (a “tap” proximity contact between a mobile phone and wireless smart speaker) are indicative of conditions under which a user may desire to perform a playback session transition between platforms. Based on detecting one of these trigger conditions, a device may initiate the playback session transfer, perhaps after receiving input data confirming that the playback session should be transferred.

[0032] As noted above, example techniques described herein involve playback session transitions across different platforms. An example implementation includes: while a first playback device is playing back audio content during a first playback session in a first domain, detecting a first event, the first event configured as a trigger to transfer playback to a second domain on a second playback device; in response to detecting the first event, determining that the first playback device was playing particular audio content from a network location at one or more servers of a particular streaming audio service and that the first playback device was at a given playback position in the particular audio content; and sending, via a network interface, instructions to initiate a playback session in the second domain to play back the particular audio content on the second playback device, the instructions indicating the network location at the one or more servers of the particular streaming audio service and an offset in the particular audio content representing the given playback position.

[0033] While some examples described herein may refer to functions performed by given actors such as “users,” “listeners,” and/or other entities, it should be understood that this is for purposes of explanation only. The claims should not be interpreted to require action by any such example actor unless explicitly required by the language of the claims themselves.

[0034] Moreover, some functions are described herein as being performed “based on” or “in response to” another element or function. “Based on” should be understood that one element or function is related to another function or element. “In response to” should be understood that one element or function is a necessary result of another function or element. For the sake of brevity, functions are generally described as being based on another function when a functional link exists; however, such disclosure should be understood as disclosing either type of functional relationship.

[0035] In the Figures, identical reference numbers identify generally similar, and/or identical, elements. To facilitate the discussion of any particular element, the most significant digit or digits of a reference number refers to the Figure in which that element is first introduced. For example, element **110a** is first introduced and discussed with reference to FIG. 1A. Many of the details, dimensions, angles and other features shown in the Figures are merely illustrative of

particular embodiments of the disclosed technology. Accordingly, other embodiments can have other details, dimensions, angles and features without departing from the spirit or scope of the disclosure. In addition, those of ordinary skill in the art will appreciate that further embodiments of the various disclosed technologies can be practiced without several of the details described below.

II. Example Operation Environment

[0036] FIGS. 1A and 1B illustrate an example configuration of a media playback system **100** (or “MPS **100**”) in which one or more embodiments disclosed herein may be implemented. Referring first to FIG. 1A, the MPS **100** as shown is associated with an example home environment having a plurality of rooms and spaces, which may be collectively referred to as a “home environment,” “smart home,” or “environment **101**.” The environment **101** comprises a household having several rooms, spaces, and/or playback zones, including a master bathroom **101a**, a master bedroom **101b**, (referred to herein as “Nick’s Room”), a second bedroom **101c**, a family room or den **101d**, an office **101e**, a living room **101f**, a dining room **101g**, a kitchen **101h**, and an outdoor patio **101i**. While certain embodiments and examples are described below in the context of a home environment, the technologies described herein may be implemented in other types of environments. In some embodiments, for example, the MPS **100** can be implemented in one or more commercial settings (e.g., a restaurant, mall, airport, hotel, a retail or other store), one or more vehicles (e.g., a sports utility vehicle, bus, car, a ship, a boat, an airplane), multiple environments (e.g., a combination of home and vehicle environments), and/or another suitable environment where multi-zone audio may be desirable.

[0037] Within these rooms and spaces, the MPS **100** includes one or more computing devices. Referring to FIGS. 1A and 1B together, such computing devices can include playback devices **102** (identified individually as playback devices **102a-102o**), network microphone devices **103** (identified individually as “NMDs” **103a-102i**), and controller devices **104a** and **104b** (collectively “controller devices **104**”). Referring to FIG. 1B, the home environment may include additional and/or other computing devices, including local network devices, such as one or more smart illumination devices **108** (FIG. 1B), a smart thermostat **110**, and a local computing device **105** (FIG. 1A). In embodiments described below, one or more of the various playback devices **102** may be configured as portable playback devices, while others may be configured as stationary playback devices. For example, the headphones **102o** (FIG. 1B) are a portable playback device, while the playback device **102d** on the bookcase may be a stationary device. As another example, the playback device **102c** on the Patio may be a battery-powered device, which may allow it to be transported to various areas within the environment **101**, and outside of the environment **101**, when it is not plugged in to a wall outlet or the like.

[0038] With reference still to FIG. 1B, the various playback, network microphone, and controller devices **102**, **103**, and **104** and/or other network devices of the MPS **100** may be coupled to one another via point-to-point connections and/or over other connections, which may be wired and/or wireless, via a network **111**, such as a LAN including a network router **109**. For example, the playback device **102j** in the Den **101d** (FIG. 1A), which may be designated as the

“Left” device, may have a point-to-point connection with the playback device **102a**, which is also in the Den **101d** and may be designated as the “Right” device. In a related embodiment, the Left playback device **102j** may communicate with other network devices, such as the playback device **102b**, which may be designated as the “Front” device, via a point-to-point connection and/or other connections via the NETWORK **111**.

[0039] As further shown in FIG. 1B, the MPS **100** may be coupled to one or more remote computing devices **106** via a wide area network (“WAN”) **107**. In some embodiments, each remote computing device **106** may take the form of one or more cloud servers. The remote computing devices **106** may be configured to interact with computing devices in the environment **101** in various ways. For example, the remote computing devices **106** may be configured to facilitate streaming and/or controlling playback of media content, such as audio, in the home environment **101**.

[0040] In some implementations, the various playback devices, NMDs, and/or controller devices **102-104** may be communicatively coupled to at least one remote computing device associated with a VAS and at least one remote computing device associated with a media content service (“MCS”). For instance, in the illustrated example of FIG. 1B, remote computing devices **106** are associated with a VAS **190** and remote computing devices **106b** are associated with an MCS **192**. Although only a single VAS **190** and a single MCS **192** are shown in the example of FIG. 1B for purposes of clarity, the MPS **100** may be coupled to multiple, different VASes and/or MCSes. In some implementations, VASes may be operated by one or more of AMAZON, GOOGLE, APPLE, MICROSOFT, SONOS or other voice assistant providers. In some implementations, MCSes may be operated by one or more of SPOTIFY, PANDORA, AMAZON MUSIC, or other media content services.

[0041] As further shown in FIG. 1B, the remote computing devices **106** further include remote computing device **106c** configured to perform certain operations, such as remotely facilitating media playback functions, managing device and system status information, directing communications between the devices of the MPS **100** and one or multiple VASes and/or MCSes, among other operations. In one example, the remote computing devices **106c** provide cloud servers for one or more SONOS Wireless HiFi Systems.

[0042] In various implementations, one or more of the playback devices **102** may take the form of or include an on-board (e.g., integrated) network microphone device. For example, the playback devices **102a-e** include or are otherwise equipped with corresponding NMDs **103a-e**, respectively. A playback device that includes or is equipped with an NMD may be referred to herein interchangeably as a playback device or an NMD unless indicated otherwise in the description. In some cases, one or more of the NMDs **103** may be a stand-alone device. For example, the NMDs **103f** and **103g** may be stand-alone devices. A stand-alone NMD may omit components and/or functionality that is typically included in a playback device, such as a speaker or related electronics. For instance, in such cases, a stand-alone NMD may not produce audio output or may produce limited audio output (e.g., relatively low-quality audio output).

[0043] The various playback and network microphone devices **102** and **103** of the MPS **100** may each be associated with a unique name, which may be assigned to the respective

devices by a user, such as during setup of one or more of these devices. For instance, as shown in the illustrated example of FIG. 1B, a user may assign the name “Bookcase” to playback device **102d** because it is physically situated on a bookcase. Similarly, the NMD **103f** may be assigned the name “Island” because it is physically situated on an island countertop in the Kitchen **101h** (FIG. 1A). Some playback devices may be assigned names according to a zone or room, such as the playback devices **102e**, **102l**, **102m**, and **102n**, which are named “Bedroom,” “Dining Room,” “Living Room,” and “Office,” respectively. Further, certain playback devices may have functionally descriptive names. For example, the playback devices **102a** and **102b** are assigned the names “Right” and “Front,” respectively, because these two devices are configured to provide specific audio channels during media playback in the zone of the Den **101d** (FIG. 1A). The playback device **102c** in the Patio may be named portable because it is battery-powered and/or readily transportable to different areas of the environment **101**. Other naming conventions are possible.

[0044] As discussed above, an NMD may detect and process sound from its environment, such as sound that includes background noise mixed with speech spoken by a person in the NMD’s vicinity. For example, as sounds are detected by the NMD in the environment, the NMD may process the detected sound to determine if the sound includes speech that contains voice input intended for the NMD and ultimately a particular VAS. For example, the NMD may identify whether speech includes a wake word associated with a particular VAS.

[0045] In the illustrated example of FIG. 1B, the NMDs **103** are configured to interact with the VAS **190** over a network via the network **111** and the router **109**. Interactions with the VAS **190** may be initiated, for example, when an NMD identifies in the detected sound a potential wake word. The identification causes a wake-word event, which in turn causes the NMD to begin transmitting detected-sound data to the VAS **190**. In some implementations, the various local network devices **102-105** (FIG. 1A) and/or remote computing devices **106c** of the MPS **100** may exchange various feedback, information, instructions, and/or related data with the remote computing devices associated with the selected VAS. Such exchanges may be related to or independent of transmitted messages containing voice inputs. In some embodiments, the remote computing device(s) and the MPS **100** may exchange data via communication paths as described herein and/or using a metadata exchange channel as described in U.S. application Ser. No. 15/438,749 filed Feb. 21, 2017, and titled “Voice Control of a Media Playback System,” which is herein incorporated by reference in its entirety.

[0046] Upon receiving the stream of sound data, the VAS **190** determines if there is voice input in the streamed data from the NMD, and if so the VAS **190** will also determine an underlying intent in the voice input. The VAS **190** may next transmit a response back to the MPS **100**, which can include transmitting the response directly to the NMD that caused the wake-word event. The response is typically based on the intent that the VAS **190** determined was present in the voice input. As an example, in response to the VAS **190** receiving a voice input with an utterance to “Play Hey Jude by The Beatles,” the VAS **190** may determine that the underlying intent of the voice input is to initiate playback and further determine that intent of the voice input is to play

the particular song “Hey Jude.” After these determinations, the VAS **190** may transmit a command to a particular MCS **192** to retrieve content (i.e., the song “Hey Jude”), and that MCS **192**, in turn, provides (e.g., streams) this content directly to the MPS **100** or indirectly via the VAS **190**. In some implementations, the VAS **190** may transmit to the MPS **100** a command that causes the MPS **100** itself to retrieve the content from the MCS **192**.

[0047] In certain implementations, NMDs may facilitate arbitration amongst one another when voice input is identified in speech detected by two or more NMDs located within proximity of one another. For example, the NMD-equipped playback device **102d** in the environment **101** (FIG. 1A) is in relatively close proximity to the NMD-equipped Living Room playback device **102m**, and both devices **102d** and **102m** may at least sometimes detect the same sound. In such cases, this may require arbitration as to which device is ultimately responsible for providing detected-sound data to the remote VAS. Examples of arbitrating between NMDs may be found, for example, in previously referenced U.S. application Ser. No. 15/438,749.

[0048] In certain implementations, an NMD may be assigned to, or otherwise associated with, a designated or default playback device that may not include an NMD. For example, the Island NMD **103f** in the Kitchen **101h** (FIG. 1A) may be assigned to the Dining Room playback device **102l**, which is in relatively close proximity to the Island NMD **103f**. In practice, an NMD may direct an assigned playback device to play audio in response to a remote VAS receiving a voice input from the NMD to play the audio, which the NMD might have sent to the VAS in response to a user speaking a command to play a certain song, album, playlist, etc. Additional details regarding assigning NMDs and playback devices as designated or default devices may be found, for example, in previously referenced U.S. patent application Ser. No. [text missing or illegible when filed]

[0049] Further aspects relating to the different components of the example MPS **100** and how the different components may interact to provide a user with a media experience may be found in the following sections. While discussions herein may generally refer to the example MPS **100**, technologies described herein are not limited to applications within, among other things, the home environment described above. For instance, the technologies described herein may be useful in other home environment configurations comprising more or fewer of any of the playback, network microphone, and/or controller devices **102-104**. For example, the technologies herein may be utilized within an environment having a single playback device **102** and/or a single NMD **103**. In some examples of such cases, the NETWORK **111** (FIG. 1B) may be eliminated and the single playback device **102** and/or the single NMD **103** may communicate directly with the remote computing devices **106-d**. In some embodiments, a telecommunication network (e.g., an LTE network, a 5G network, etc.) may communicate with the various playback, network microphone, and/or controller devices **102-104** independent of a LAN.

a. Example Playback & Network Microphone Devices

[0050] FIG. 2A is a functional block diagram illustrating certain aspects of one of the playback devices **102** of the MPS **100** of FIGS. 1A and 1B. As shown, the playback

device **102** includes various components, each of which is discussed in further detail below, and the various components of the playback device **102** may be operably coupled to one another via a system bus, communication network, or some other connection mechanism. In the illustrated example of FIG. 2A, the playback device **102** may be referred to as an “NMD-equipped” playback device because it includes components that support the functionality of an NMD, such as one of the NMDs **103** shown in FIG. 1A.

[0051] As shown, the playback device **102** includes at least one processor **212**, which may be a clock-driven computing component configured to process input data according to instructions stored in memory **213**. The memory **213** may be a tangible, non-transitory, computer-readable medium configured to store instructions that are executable by the processor **212**. For example, the memory **213** may be data storage that can be loaded with software code **214** that is executable by the processor **212** to achieve certain functions.

[0052] In one example, these functions may involve the playback device **102** retrieving audio data from an audio source, which may be another playback device. In another example, the functions may involve the playback device **102** sending audio data, detected-sound data (e.g., corresponding to a voice input), and/or other information to another device on a network via at least one network interface **224**. In yet another example, the functions may involve the playback device **102** causing one or more other playback devices to synchronously playback audio with the playback device **102**. In yet a further example, the functions may involve the playback device **102** facilitating being paired or otherwise bonded with one or more other playback devices to create a multi-channel audio environment. Numerous other example functions are possible, some of which are discussed below.

[0053] As just mentioned, certain functions may involve the playback device **102** synchronizing playback of audio content with one or more other playback devices. During synchronous playback, a listener may not perceive time-delay differences between playback of the audio content by the synchronized playback devices. U.S. Pat. No. 8,234,395 filed on Apr. 4, 2004, and titled “System and method for synchronizing operations among a plurality of independently clocked digital data processing devices,” which is hereby incorporated by reference in its entirety, provides in more detail some examples for audio playback synchronization among playback devices.

[0054] To facilitate audio playback, the playback device **102** includes audio processing components **216** that are generally configured to process audio prior to the playback device **102** rendering the audio. In this respect, the audio processing components **216** may include one or more digital-to-analog converters (“DAC”), one or more audio pre-processing components, one or more audio enhancement components, one or more digital signal processors (“DSPs”), and so on. In some implementations, one or more of the audio processing components **216** may be a subcomponent of the processor **212**. In operation, the audio processing components **216** receive analog and/or digital audio and process and/or otherwise intentionally alter the audio to produce audio signals for playback.

[0055] The produced audio signals may then be provided to one or more audio amplifiers **217** for amplification and playback through one or more speakers **218** operably coupled to the amplifiers **217**. The audio amplifiers **217** may

include components configured to amplify audio signals to a level for driving one or more of the speakers **218**.

[0056] Each of the speakers **218** may include an individual transducer (e.g., a “driver”) or the speakers **218** may include a complete speaker system involving an enclosure with one or more drivers. A particular driver of a speaker **218** may include, for example, a subwoofer (e.g., for low frequencies), a mid-range driver (e.g., for middle frequencies), and/or a tweeter (e.g., for high frequencies). In some cases, a transducer may be driven by an individual corresponding audio amplifier of the audio amplifiers **217**. In some implementations, a playback device may not include the speakers **218**, but instead may include a speaker interface for connecting the playback device to external speakers. In certain embodiments, a playback device may include neither the speakers **218** nor the audio amplifiers **217**, but instead may include an audio interface (not shown) for connecting the playback device to an external audio amplifier or audio-visual receiver.

[0057] In addition to producing audio signals for playback by the playback device **102**, the audio processing components **216** may be configured to process audio to be sent to one or more other playback devices, via the network interface **224**, for playback. In example scenarios, audio content to be processed and/or played back by the playback device **102** may be received from an external source, such as via an audio line-in interface (e.g., an auto-detecting 3.5 mm audio line-in connection) of the playback device **102** (not shown) or via the network interface **224**, as described below.

[0058] As shown, the at least one network interface **224**, may take the form of one or more wireless interfaces **225** and/or one or more wired interfaces **226**. A wireless interface may provide network interface functions for the playback device **102** to wirelessly communicate with other devices (e.g., other playback device(s), NMD(s), and/or controller device(s)) in accordance with a communication protocol (e.g., any wireless standard including IEEE 802.11a, 802.11b, 802.11g, 802.11n, 802.11ac, 802.15, 4G mobile communication standard, and so on). A wired interface may provide network interface functions for the playback device **102** to communicate over a wired connection with other devices in accordance with a communication protocol (e.g., IEEE 802.3). While the network interface **224** shown in FIG. 2A include both wired and wireless interfaces, the playback device **102** may in some implementations include only wireless interface(s) or only wired interface(s).

[0059] In general, the network interface **224** facilitates data flow between the playback device **102** and one or more other devices on a data network. For instance, the playback device **102** may be configured to receive audio content over the data network from one or more other playback devices, network devices within a LAN, and/or audio content sources over a WAN, such as the Internet. In one example, the audio content and other signals transmitted and received by the playback device **102** may be transmitted in the form of digital packet data comprising an Internet Protocol (IP)-based source address and IP-based destination addresses. In such a case, the network interface **224** may be configured to parse the digital packet data such that the data destined for the playback device **102** is properly received and processed by the playback device **102**.

[0060] As shown in FIG. 2A, the playback device **102** also includes voice processing components **220** that are operably coupled to one or more microphones **222**. The microphones

222 are configured to detect sound (i.e., acoustic waves) in the environment of the playback device **102**, which is then provided to the voice processing components **220**. More specifically, each microphone **222** is configured to detect sound and convert the sound into a digital or analog signal representative of the detected sound, which can then cause the voice processing component **220** to perform various functions based on the detected sound, as described in greater detail below. In one implementation, the microphones **222** are arranged as an array of microphones (e.g., an array of six microphones). In some implementations, the playback device **102** includes more than six microphones (e.g., eight microphones or twelve microphones) or fewer than six microphones (e.g., four microphones, two microphones, or a single microphone).

[0061] In operation, the voice-processing components **220** are generally configured to detect and process sound received via the microphones **222**, identify potential voice input in the detected sound, and extract detected-sound data to enable a VAS, such as the VAS **190** (FIG. 1B), to process voice input identified in the detected-sound data. The voice processing components **220** may include one or more analog-to-digital converters, an acoustic echo canceller (“AEC”), a spatial processor (e.g., one or more multi-channel Wiener filters, one or more other filters, and/or one or more beam former components), one or more buffers (e.g., one or more circular buffers), one or more wake-word engines, one or more voice extractors, and/or one or more speech processing components (e.g., components configured to recognize a voice of a particular user or a particular set of users associated with a household), among other example voice processing components. In example implementations, the voice processing components **220** may include or otherwise take the form of one or more DSPs or one or more modules of a DSP. In this respect, certain voice processing components **220** may be configured with particular parameters (e.g., gain and/or spectral parameters) that may be modified or otherwise tuned to achieve particular functions. In some implementations, one or more of the voice processing components **220** may be a subcomponent of the processor **212**.

[0062] As further shown in FIG. 2A, the playback device **102** also includes power components **227**. The power components **227** include at least an external power source interface **228**, which may be coupled to a power source (not shown) via a power cable or the like that physically connects the playback device **102** to an electrical outlet or some other external power source. Other power components may include, for example, transformers, converters, and like components configured to format electrical power.

[0063] In some implementations, the power components **227** of the playback device **102** may additionally include an internal power source **229** (e.g., one or more batteries) configured to power the playback device **102** without a physical connection to an external power source. When equipped with the internal power source **229**, the playback device **102** may operate independent of an external power source. In some such implementations, the external power source interface **228** may be configured to facilitate charging the internal power source **229**. As discussed before, a playback device comprising an internal power source may be referred to herein as a “portable playback device.” On the other hand, a playback device that operates using an external power source may be referred to herein as a “stationary

playback device,” although such a device may in fact be moved around a home or other environment.

[0064] The playback device **102** further includes a user interface **240** that may facilitate user interactions independent of or in conjunction with user interactions facilitated by one or more of the controller devices **104**. In various embodiments, the user interface **240** includes one or more physical buttons and/or supports graphical interfaces provided on touch sensitive screen(s) and/or surface(s), among other possibilities, for a user to directly provide input. The user interface **240** may further include one or more of lights (e.g., LEDs) and the speakers to provide visual and/or audio feedback to a user.

[0065] As an illustrative example, FIG. 2B shows an example housing **230** of the playback device **102** that includes a user interface in the form of a control area **232** at a top portion **234** of the housing **230**. The control area **232** includes buttons **236a-c** for controlling audio playback, volume level, and other functions. The control area **232** also includes a button **236d** for toggling the microphones **222** to either an on state or an off state.

[0066] As further shown in FIG. 2B, the control area **232** is at least partially surrounded by apertures formed in the top portion **234** of the housing **230** through which the microphones **222** (not visible in FIG. 2B) receive the sound in the environment of the playback device **102**. The microphones **222** may be arranged in various positions along and/or within the top portion **234** or other areas of the housing **230** so as to detect sound from one or more directions relative to the playback device **102**.

[0067] By way of illustration, SONOS, Inc. presently offers (or has offered) for sale certain playback devices that may implement certain of the embodiments disclosed herein, including a “PLAY:1,” “PLAY:3,” “PLAY:5,” “PLAYBAR,” “CONNECT:AMP,” “PLAYBASE,” “BEAM,” “CONNECT,” and “SUB.” Any other past, present, and/or future playback devices may additionally or alternatively be used to implement the playback devices of example embodiments disclosed herein. Additionally, it should be understood that a playback device is not limited to the examples illustrated in FIGS. 2A or 2B or to the SONOS product offerings. For example, a playback device may include, or otherwise take the form of, a wired or wireless headphone set, which may operate as a part of the MPS **100** via a network interface or the like. In another example, a playback device may include or interact with a docking station for personal mobile media playback devices. In yet another example, a playback device may be integral to another device or component such as a television, a lighting fixture, or some other device for indoor or outdoor use.

b. Example Playback Device Configurations

[0068] FIGS. 3A-3E show example configurations of playback devices. Referring first to FIG. 3A, in some example instances, a single playback device may belong to a zone. For example, the playback device **102c** (FIG. 1A) on the Patio may belong to Zone A. In some implementations described below, multiple playback devices may be “bonded” to form a “bonded pair,” which together form a single zone. For example, the playback device **102f** (FIG. 1A) named “Bed 1” in FIG. 3A may be bonded to the playback device **102g** (FIG. 1A) named “Bed 2” in FIG. 3A to form Zone B. Bonded playback devices may have different playback responsibilities (e.g., channel responsibil-

ties). In another implementation described below, multiple playback devices may be merged to form a single zone. For example, the playback device **102d** named “Bookcase” may be merged with the playback device **102m** named “Living Room” to form a single Zone C. The merged playback devices **102d** and **102m** may not be specifically assigned different playback responsibilities. That is, the merged playback devices **102d** and **102m** may, aside from playing audio content in synchrony, each play audio content as they would if they were not merged.

[0069] For purposes of control, each zone in the MPS **100** may be represented as a single user interface (“UI”) entity. For example, as displayed by the controller devices **104**, Zone A may be provided as a single entity named “Portable,” Zone B may be provided as a single entity named “Stereo,” and Zone C may be provided as a single entity named “Living Room.”

[0070] In various embodiments, a zone may take on the name of one of the playback devices belonging to the zone. For example, Zone C may take on the name of the Living Room device **102m** (as shown). In another example, Zone C may instead take on the name of the Bookcase device **102d**. In a further example, Zone C may take on a name that is some combination of the Bookcase device **102d** and Living Room device **102m**. The name that is chosen may be selected by a user via inputs at a controller device **104**. In some embodiments, a zone may be given a name that is different than the device(s) belonging to the zone. For example, Zone B in FIG. 3A is named “Stereo” but none of the devices in Zone B have this name. In one aspect, Zone B is a single UI entity representing a single device named “Stereo,” composed of constituent devices “Bed 1” and “Bed 2.” In one implementation, the Bed 1 device may be playback device **102f** in the master bedroom **101h** (FIG. 1A) and the Bed 2 device may be the playback device **102g** also in the master bedroom **101h** (FIG. 1A).

[0071] As noted above, playback devices that are bonded may have different playback responsibilities, such as playback responsibilities for certain audio channels. For example, as shown in FIG. 3B, the Bed 1 and Bed 2 devices **102f** and **102g** may be bonded so as to produce or enhance a stereo effect of audio content. In this example, the Bed 1 playback device **102f** may be configured to play a left channel audio component, while the Bed 2 playback device **102g** may be configured to play a right channel audio component. In some implementations, such stereo bonding may be referred to as “pairing.”

[0072] Additionally, playback devices that are configured to be bonded may have additional and/or different respective speaker drivers. As shown in FIG. 3C, the playback device **102b** named “Front” may be bonded with the playback device **102k** named “SUB.” The Front device **102b** may render a range of mid to high frequencies, and the SUB device **102k** may render low frequencies as, for example, a subwoofer. When unbonded, the Front device **102b** may be configured to render a full range of frequencies. As another example, FIG. 3D shows the Front and SUB devices **102b** and **102k** further bonded with Right and Left playback devices **102a** and **102j**, respectively. In some implementations, the Right and Left devices **102a** and **102j** may form surround or “satellite” channels of a home theater system. The bonded playback devices **102a**, **102b**, **102j**, and **102k** may form a single Zone D (FIG. 3A).

[0073] In some implementations, playback devices may also be “merged.” In contrast to certain bonded playback devices, playback devices that are merged may not have assigned playback responsibilities, but may each render the full range of audio content that each respective playback device is capable of. Nevertheless, merged devices may be represented as a single UI entity (i.e., a zone, as discussed above). For instance, FIG. 3E shows the playback devices **102d** and **102m** in the Living Room merged, which would result in these devices being represented by the single UI entity of Zone C. In one embodiment, the playback devices **102d** and **102m** may playback audio in synchrony, during which each outputs the full range of audio content that each respective playback device **102d** and **102m** is capable of rendering.

[0074] In some embodiments, a stand-alone NMD may be in a zone by itself. For example, the NMD **103h** from FIG. 1A is named “Closet” and forms Zone I in FIG. 3A. An NMD may also be bonded or merged with another device so as to form a zone. For example, the NMD device **103f** named “Island” may be bonded with the playback device **102i** Kitchen, which together form Zone F, which is also named “Kitchen.” Additional details regarding assigning NMDs and playback devices as designated or default devices may be found, for example, in previously referenced U.S. patent application Ser. No. 15/438,749. In some embodiments, a stand-alone NMD may not be assigned to a zone.

[0075] Zones of individual, bonded, and/or merged devices may be arranged to form a set of playback devices that playback audio in synchrony. Such a set of playback devices may be referred to as a “group,” “zone group,” “synchrony group,” or “playback group.” In response to inputs provided via a controller device **104**, playback devices may be dynamically grouped and ungrouped to form new or different groups that synchronously play back audio content. For example, referring to FIG. 3A, Zone A may be grouped with Zone B to form a zone group that includes the playback devices of the two zones. As another example, Zone A may be grouped with one or more other Zones C-I. The Zones A-I may be grouped and ungrouped in numerous ways. For example, three, four, five, or more (e.g., all) of the Zones A-I may be grouped. When grouped, the zones of individual and/or bonded playback devices may play back audio in synchrony with one another, as described in previously referenced U.S. Pat. No. 8,234,395. Grouped and bonded devices are example types of associations between portable and stationary playback devices that may be caused in response to a trigger event, as discussed above and described in greater detail below.

[0076] In various implementations, the zones in an environment may be assigned a particular name, which may be the default name of a zone within a zone group or a combination of the names of the zones within a zone group, such as “Dining Room+Kitchen,” as shown in FIG. 3A. In some embodiments, a zone group may be given a unique name selected by a user, such as “Nick’s Room,” as also shown in FIG. 3A. The name “Nick’s Room” may be a name chosen by a user over a prior name for the zone group, such as the room name “Master Bedroom.”

[0077] Referring back to FIG. 2A, certain data may be stored in the memory **213** as one or more state variables that are periodically updated and used to describe the state of a playback zone, the playback device(s), and/or a zone group associated therewith. The memory **213** may also include the

data associated with the state of the other devices of the MPS 100, which may be shared from time to time among the devices so that one or more of the devices have the most recent data associated with the system.

[0078] In some embodiments, the memory 213 of the playback device 102 may store instances of various variable types associated with the states. Variables instances may be stored with identifiers (e.g., tags) corresponding to type. For example, certain identifiers may be a first type “a1” to identify playback device(s) of a zone, a second type “b1” to identify playback device(s) that may be bonded in the zone, and a third type “c1” to identify a zone group to which the zone may belong. As a related example, in FIG. 1A, identifiers associated with the Patio may indicate that the Patio is the only playback device of a particular zone and not in a zone group. Identifiers associated with the Living Room may indicate that the Living Room is not grouped with other zones but includes bonded playback devices 102a, 102b, 102j, and 102k. Identifiers associated with the Dining Room may indicate that the Dining Room is part of Dining Room+Kitchen group and that devices 103f and 102i are bonded. Identifiers associated with the Kitchen may indicate the same or similar information by virtue of the Kitchen being part of the Dining Room+Kitchen zone group. Other example zone variables and identifiers are described below.

[0079] In yet another example, the MPS 100 may include variables or identifiers representing other associations of zones and zone groups, such as identifiers associated with Areas, as shown in FIG. 3A. An Area may involve a cluster of zone groups and/or zones not within a zone group. For instance, FIG. 3A shows a first area named “First Area” and a second area named “Second Area.” The First Area includes zones and zone groups of the Patio, Den, Dining Room, Kitchen, and Bathroom. The Second Area includes zones and zone groups of the Bathroom, Nick’s Room, Bedroom, and Living Room. In one aspect, an Area may be used to invoke a cluster of zone groups and/or zones that share one or more zones and/or zone groups of another cluster. In this respect, such an Area differs from a zone group, which does not share a zone with another zone group. Further examples of techniques for implementing Areas may be found, for example, in U.S. application Ser. No. 15/682,506 filed Aug. 21, 2017 and titled “Room Association Based on Name,” and U.S. Pat. No. 8,483,853 filed Sep. 11, 2007, and titled “Controlling and manipulating groupings in a multi-zone media system.” Each of these applications is incorporated herein by reference in its entirety. In some embodiments, the MPS 100 may not implement Areas, in which case the system may not store variables associated with Areas.

[0080] The memory 213 may be further configured to store other data. Such data may pertain to audio sources accessible by the playback device 102 or a playback queue that the playback device (or some other playback device(s)) may be associated with. In embodiments described below, the memory 213 is configured to store a set of command data for selecting a particular VAS when processing voice inputs. During operation, one or more playback zones in the environment of FIG. 1A may each be playing different audio content. For instance, the user may be grilling in the Patio zone and listening to hip hop music being played by the playback device 102c, while another user may be preparing food in the Kitchen zone and listening to classical music being played by the playback device 102i. In another

example, a playback zone may play the same audio content in synchrony with another playback zone.

[0081] For instance, the user may be in the Office zone where the playback device 102n is playing the same hip-hop music that is being playing by playback device 102c in the Patio zone. In such a case, playback devices 102c and 102n may be playing the hip-hop in synchrony such that the user may seamlessly (or at least substantially seamlessly) enjoy the audio content that is being played out-loud while moving between different playback zones. Synchronization among playback zones may be achieved in a manner similar to that of synchronization among playback devices, as described in previously referenced U.S. Pat. No. 8,234,395.

[0082] As suggested above, the zone configurations of the MPS 100 may be dynamically modified. As such, the MPS 100 may support numerous configurations. For example, if a user physically moves one or more playback devices to or from a zone, the MPS 100 may be reconfigured to accommodate the change(s). For instance, if the user physically moves the playback device 102c from the Patio zone to the Office zone, the Office zone may now include both the playback devices 102c and 102n. In some cases, the user may pair or group the moved playback device 102c with the Office zone and/or rename the players in the Office zone using, for example, one of the controller devices 104 and/or voice input. As another example, if one or more playback devices 102 are moved to a particular space in the home environment that is not already a playback zone, the moved playback device(s) may be renamed or associated with a playback zone for the particular space.

[0083] Further, different playback zones of the MPS 100 may be dynamically combined into zone groups or split up into individual playback zones. For example, the Dining Room zone and the Kitchen zone may be combined into a zone group for a dinner party such that playback devices 102i and 102j may render audio content in synchrony. As another example, bonded playback devices in the Den zone may be split into (i) a television zone and (ii) a separate listening zone. The television zone may include the Front playback device 102b. The listening zone may include the Right, Left, and SUB playback devices 102a, 102j, and 102k, which may be grouped, paired, or merged, as described above. Splitting the Den zone in such a manner may allow one user to listen to music in the listening zone in one area of the living room space, and another user to watch the television in another area of the living room space. In a related example, a user may utilize either of the NMD 103a or 103b (FIG. 1B) to control the Den zone before it is separated into the television zone and the listening zone. Once separated, the listening zone may be controlled, for example, by a user in the vicinity of the NMD 103a, and the television zone may be controlled, for example, by a user in the vicinity of the NMD 103b. As described above, however, any of the NMDs 103 may be configured to control the various playback and other devices of the MPS 100.

c. Example Controller Devices

[0084] FIG. 4 is a functional block diagram illustrating certain aspects of a selected one of the controller devices 104 of the MPS 100 of FIG. 1A. Such controller devices may also be referred to herein as a “control device” or “controller.” The controller device shown in FIG. 4 may include components that are generally similar to certain components of the network devices described above, such as a processor

412, memory 413 storing program software 414, at least one network interface 424, and one or more microphones 422. In one example, a controller device may be a dedicated controller for the MPS 100. In another example, a controller device may be a network device on which media playback system controller application software may be installed, such as for example, an iPhone™, iPad™ or any other smart phone, tablet, or network device (e.g., a networked computer such as a PC or Mac™).

[0085] The memory 413 of the controller device 104 may be configured to store controller application software and other data associated with the MPS 100 and/or a user of the system 100. The memory 413 may be loaded with instructions in software 414 that are executable by the processor 412 to achieve certain functions, such as facilitating user access, control, and/or configuration of the MPS 100. The controller device 104 is configured to communicate with other network devices via the network interface 424, which may take the form of a wireless interface, as described above.

[0086] In one example, system information (e.g., such as a state variable) may be communicated between the controller device 104 and other devices via the network interface 424. For instance, the controller device 104 may receive playback zone and zone group configurations in the MPS 100 from a playback device, an NMD, or another network device. Likewise, the controller device 104 may transmit such system information to a playback device or another network device via the network interface 424. In some cases, the other network device may be another controller device.

[0087] The controller device 104 may also communicate playback device control commands, such as volume control and audio playback control, to a playback device via the network interface 424. As suggested above, changes to configurations of the MPS 100 may also be performed by a user using the controller device 104. The configuration changes may include adding/removing one or more playback devices to/from a zone, adding/removing one or more zones to/from a zone group, forming a bonded or merged player, separating one or more playback devices from a bonded or merged player, among others.

[0088] As shown in FIG. 4, the controller device 104 also includes a user interface 440 that is generally configured to facilitate user access and control of the MPS 100. The user interface 440 may include a touch-screen display or other physical interface configured to provide various graphical controller interfaces, such as the controller interfaces 540a and 540b shown in FIGS. 5A and 5B. Referring to FIGS. 5A and 5B together, the controller interfaces 540a and 540b includes a playback control region 542, a playback zone region 543, a playback status region 544, a playback queue region 546, and a sources region 548. The user interface as shown is just one example of an interface that may be provided on a network device, such as the controller device shown in FIG. 4, and accessed by users to control a media playback system, such as the MPS 100. Other user interfaces of varying formats, styles, and interactive sequences may alternatively be implemented on one or more network devices to provide comparable control access to a media playback system.

[0089] The playback control region 542 (FIG. 5A) may include selectable icons (e.g., by way of touch or by using a cursor) that, when selected, cause playback devices in a selected playback zone or zone group to play or pause, fast

forward, rewind, skip to next, skip to previous, enter/exit shuffle mode, enter/exit repeat mode, enter/exit cross fade mode, etc. The playback control region 542 may also include selectable icons that, when selected, modify equalization settings and/or playback volume, among other possibilities.

[0090] The playback zone region 543 (FIG. 5B) may include representations of playback zones within the MPS 100. The playback zones regions 543 may also include a representation of zone groups, such as the Dining Room+ Kitchen zone group, as shown.

[0091] In some embodiments, the graphical representations of playback zones may be selectable to bring up additional selectable icons to manage or configure the playback zones in the MPS 100, such as a creation of bonded zones, creation of zone groups, separation of zone groups, and renaming of zone groups, among other possibilities.

[0092] For example, as shown, a “group” icon may be provided within each of the graphical representations of playback zones. The “group” icon provided within a graphical representation of a particular zone may be selectable to bring up options to select one or more other zones in the MPS 100 to be grouped with the particular zone. Once grouped, playback devices in the zones that have been grouped with the particular zone will be configured to play audio content in synchrony with the playback device(s) in the particular zone. Analogously, a “group” icon may be provided within a graphical representation of a zone group. In this case, the “group” icon may be selectable to bring up options to deselect one or more zones in the zone group to be removed from the zone group. Other interactions and implementations for grouping and ungrouping zones via a user interface are also possible. The representations of playback zones in the playback zone region 543 (FIG. 5B) may be dynamically updated as playback zone or zone group configurations are modified.

[0093] The playback status region 544 (FIG. 5A) may include graphical representations of audio content that is presently being played, previously played, or scheduled to play next in the selected playback zone or zone group. The selected playback zone or zone group may be visually distinguished on a controller interface, such as within the playback zone region 543 and/or the playback status region 544. The graphical representations may include track title, artist name, album name, album year, track length, and/or other relevant information that may be useful for the user to know when controlling the MPS 100 via a controller interface.

[0094] The playback queue region 546 may include graphical representations of audio content in a playback queue associated with the selected playback zone or zone group. In some embodiments, each playback zone or zone group may be associated with a playback queue comprising information corresponding to zero or more audio items for playback by the playback zone or zone group. For instance, each audio item in the playback queue may comprise a uniform resource identifier (URI), a uniform resource locator (URL), or some other identifier that may be used by a playback device in the playback zone or zone group to find and/or retrieve the audio item from a local audio content source or a networked audio content source, which may then be played back by the playback device.

[0095] In one example, a playlist may be added to a playback queue, in which case information corresponding to each audio item in the playlist may be added to the playback

queue. In another example, audio items in a playback queue may be saved as a playlist. In a further example, a playback queue may be empty, or populated but “not in use” when the playback zone or zone group is playing continuously streamed audio content, such as Internet radio that may continue to play until otherwise stopped, rather than discrete audio items that have playback durations. In an alternative embodiment, a playback queue can include Internet radio and/or other streaming audio content items and be “in use” when the playback zone or zone group is playing those items. Other examples are also possible.

[0096] When playback zones or zone groups are “grouped” or “ungrouped,” playback queues associated with the affected playback zones or zone groups may be cleared or re-associated. For example, if a first playback zone including a first playback queue is grouped with a second playback zone including a second playback queue, the established zone group may have an associated playback queue that is initially empty, that contains audio items from the first playback queue (such as if the second playback zone was added to the first playback zone), that contains audio items from the second playback queue (such as if the first playback zone was added to the second playback zone), or a combination of audio items from both the first and second playback queues.

[0097] Subsequently, if the established zone group is ungrouped, the resulting first playback zone may be re-associated with the previous first playback queue or may be associated with a new playback queue that is empty or contains audio items from the playback queue associated with the established zone group before the established zone group was ungrouped. Similarly, the resulting second playback zone may be re-associated with the previous second playback queue or may be associated with a new playback queue that is empty or contains audio items from the playback queue associated with the established zone group before the established zone group was ungrouped. Other examples are also possible.

[0098] With reference still to FIGS. 5A and 5B, the graphical representations of audio content in the playback queue region 646 (FIG. 5A) may include track titles, artist names, track lengths, and/or other relevant information associated with the audio content in the playback queue. In one example, graphical representations of audio content may be selectable to bring up additional selectable icons to manage and/or manipulate the playback queue and/or audio content represented in the playback queue. For instance, a represented audio content may be removed from the playback queue, moved to a different position within the playback queue, or selected to be played immediately, or after any currently playing audio content, among other possibilities. A playback queue associated with a playback zone or zone group may be stored in a memory on one or more playback devices in the playback zone or zone group, on a playback device that is not in the playback zone or zone group, and/or some other designated device. Playback of such a playback queue may involve one or more playback devices playing back media items of the queue, perhaps in sequential or random order.

[0099] The sources region 548 may include graphical representations of selectable audio content sources and/or selectable voice assistants associated with a corresponding VAS. The VASes may be selectively assigned. In some examples, multiple VASes, such as AMAZON’s Alexa,

MICROSOFT’s Cortana, etc., may be invocable by the same NMD. In some embodiments, a user may assign a VAS exclusively to one or more NMDs. For example, a user may assign a first VAS to one or both of the NMDs 102a and 102b in the Living Room shown in FIG. 1A, and a second VAS to the NMD 103f in the Kitchen. Other examples are possible.

d. Example Audio Content Sources

[0100] The audio sources in the sources region 548 may be audio content sources from which audio content may be retrieved and played by the selected playback zone or zone group. One or more playback devices in a zone or zone group may be configured to retrieve for playback audio content (e.g., according to a corresponding URI or URL for the audio content) from a variety of available audio content sources. In one example, audio content may be retrieved by a playback device directly from a corresponding audio content source (e.g., via a line-in connection). In another example, audio content may be provided to a playback device over a network via one or more other playback devices or network devices. As described in greater detail below, in some embodiments audio content may be provided by one or more media content services.

[0101] Example audio content sources may include a memory of one or more playback devices in a media playback system such as the MPS 100 of FIG. 1, local music libraries on one or more network devices (e.g., a controller device, a network-enabled personal computer, or a networked-attached storage (“NAS”)), streaming audio services providing audio content via the Internet (e.g., cloud-based music services), or audio sources connected to the media playback system via a line-in input connection on a playback device or network device, among other possibilities.

[0102] In some embodiments, audio content sources may be added or removed from a media playback system such as the MPS 100 of FIG. 1A. In one example, an indexing of audio items may be performed whenever one or more audio content sources are added, removed, or updated. Indexing of audio items may involve scanning for identifiable audio items in all folders/directories shared over a network accessible by playback devices in the media playback system and generating or updating an audio content database comprising metadata (e.g., title, artist, album, track length, among others) and other associated information, such as a URI or URL for each identifiable audio item found. Other examples for managing and maintaining audio content sources may also be possible.

[0103] FIG. 6 is a message flow diagram illustrating data exchanges between devices of the MPS 100. At step 650a, the MPS 100 receives an indication of selected media content (e.g., one or more songs, albums, playlists, podcasts, videos, stations) via the control device 104. The selected media content can comprise, for example, media items stored locally on or more devices (e.g., the audio source 105 of FIG. 1C) connected to the media playback system and/or media items stored on one or more media service servers (one or more of the remote computing devices 106 of FIG. 1B). In response to receiving the indication of the selected media content, the control device 104 transmits a message 651a to the playback device 102 (FIGS. 1A-1C) to add the selected media content to a playback queue on the playback device 102.

[0104] At step 650b, the playback device 102 receives the message 651a and adds the selected media content to the playback queue for play back.

[0105] At step 650c, the control device 104 receives input corresponding to a command to play back the selected media content. In response to receiving the input corresponding to the command to play back the selected media content, the control device 104 transmits a message 651b to the playback device 102 causing the playback device 102 to play back the selected media content. In response to receiving the message 651b, the playback device 102 transmits a message 651c to the computing device 106 requesting the selected media content. The computing device 106, in response to receiving the message 651c, transmits a message 651d comprising data (e.g., audio data, video data, a URL, a URI) corresponding to the requested media content.

[0106] At step 650d, the playback device 102 receives the message 651d with the data corresponding to the requested media content and plays back the associated media content.

[0107] At step 650e, the playback device 102 optionally causes one or more other devices to play back the selected media content. In one example, the playback device 102 is one of a bonded zone of two or more players (FIG. 1M). The playback device 102 can receive the selected media content and transmit all or a portion of the media content to other devices in the bonded zone. In another example, the playback device 102 is a coordinator of a group and is configured to transmit and receive timing information from one or more other devices in the group. The other one or more devices in the group can receive the selected media content from the computing device 106, and begin playback of the selected media content in response to a message from the playback device 102 such that all of the devices in the group play back the selected media content in synchrony.

III. Example Mobile Device to Facilitate Playback Session Transitions

[0108] In some example implementations, a mobile device, such as a smartphone or tablet, may facilitate playback session transitions across different platforms. FIG. 7A is a functional block diagram illustrating certain aspects of an example mobile device 704. As shown, the mobile device 704 includes various components, each of which is discussed in further detail below. The various components of the mobile device 104 may be operably coupled to one another via a system bus, communication network, or some other connection mechanism.

[0109] As shown, the mobile device 704 includes at least one processor 712, which may be a clock-driven computing component configured to process input data according to instructions stored in memory 713. The memory 713 may be a tangible, non-transitory, computer-readable medium configured to store instructions that are executable by the processor 712. For example, the memory 713 may be data storage that can be loaded with software code 714 that is executable by the processor 712 to achieve certain functions.

[0110] The mobile device 704 also includes audio processing components 216, amplifiers 217, and/or speakers 218. These audio components may be used for telephony functions (e.g., voice and/or video calls) as well as audio playback. Generally, in an effort to be commercially viable as a mobile device, the mobile device 704 includes relatively less capable versions of these audio components as compared with the playback devices 102, which may include

relatively more capable versions of the audio processing components 216, amplifiers 217, and/or speakers 218 specifically intended for audio playback.

[0111] The at least one network interface 724 may take the form of one or more wireless interfaces 725 and/or one or more wired interfaces 726. The wireless interface 725 may provide network interface functions for the NMD 703 to wirelessly communicate with other devices (e.g., playback device(s) 102, other NMD(s) 103, and/or controller device(s) 104) in accordance with a communication protocol (e.g., any wireless standard including IEEE 802.11a, 802.11b, 802.11g, 802.11n, 802.11ac, 802.15, 4G mobile communication standard, and so on). The wired interface 726 may provide network interface functions for the NMD 703 to communicate over a wired connection with other devices in accordance with a communication protocol (e.g., IEEE 802.3). While the network interface 724 shown in FIG. 7A includes both wired and wireless interfaces, the playback device 102 may in various implementations include only wireless interface(s) or only wired interface(s).

[0112] The software code 714 may include an operating system 760, such as Apple iOS or ANDROID. The operating system 760 is software that includes basic features like process management, interrupts, memory management, a file system, device drivers, networking, security, and input/output. For hardware functions such as input and output and memory allocation, the operating system acts as an intermediary between applications (apps) and the computer hardware, allowing the apps to use hardware such as the network interface(s) 724 and the user interface 740.

[0113] The software code 714 further includes various applications ("apps"). By way of illustration, the mobile device 704 includes a media playback system (MPS) control app 761, one or more streaming media service apps 762, and one or more media player apps 763. General purpose mobile devices 704 will generally include additional applications.

[0114] The mobile device 104 further includes a user interface 740 that may facilitate user interactions independent of or in conjunction with user interactions facilitated by one or more of the controller devices 104. In various embodiments, the user interface 740 includes a touch-sensitive graphical display, which can display various control interfaces provided by the operating system 760 and/or the MPS control app 761, the streaming media service app(s) 762, and/or the media player app(s) 763, as well as other apps that may be installed on the mobile device 704.

[0115] By executing the MPS control app 761, the mobile device 704 may function as the control device 104. The MPS control app 761 provides control of the media playback system 100 (FIGS. 1A and 1B) in a first domain (referred to herein as a native domain). In particular, the MPS control app 761 may cause the user interface 740 to display controller interface(s) with playback controls to facilitate control of the playback devices 102 (e.g., the controller interfaces 540a and/or 540b shown in FIGS. 5A and 5B). A commercial embodiment of the MPS control app 761 is the SONOS app, currently available for APPLE iOS, ANDROID, WINDOWS, and MAC.

[0116] A user may start a playback session in the native domain by using the controller interface(s) of the MPS control app 761. In particular, a user may use the controller interface(s) of the MPS control app 761 to select the Kitchen zone (including the playback device 102i) and to select one or more media items to play back in the Kitchen zone.

Further, the user may start playback by selecting a play/pause control, or by selecting a media item in the playback queue, among other possible inputs.

[0117] As discussed above, a user of the MPS 100 may register the MPS 100 with one or more streaming media services by providing user account information. For instance, a user may provide their user accounts and passwords for one or more streaming media services via a controller interface of the MPS control app 761. After registering a given streaming media service, the user may browse or search for media provided by that service using the MPS control app 761. Further, the user may select media with the MPS control app 761 and play back that media on the playback devices 102a-o (FIG. 1A).

[0118] When executing, each streaming media service app 762 provide a respective interface to access streaming media provided by a respective streaming media service (also referred to as media content services). For instance, the SPOTIFY app provides an interface to search for and browse audio tracks available for streaming from the SPOTIFY streaming audio service. By selecting an audio track, a stream is initiated to the mobile device 704 via the network interface 724 (e.g., from the remote computing devices 106b (FIG. 1b)), which can be played back by the mobile device. Other commercially-available streaming media services include PANDORA, AMAZON MUSIC, APPLE MUSIC, TIDAL, PLAY MUSIC, DEEZER, and others. Before using the streaming media service app 762, a user will typically be required to log-in to the service using their user account information for the corresponding streaming media service.

[0119] The mobile device 704 can play back the audio track using the audio processing components 216, amplifiers 217, and/or speakers 218 or an externally-connected playback device, such as a speaker, headphones, or earbuds (e.g., connected via a 3.5 mm line-in or BLUETOOTH). In contrast to the playback devices 102 of the MPS 100, such externally-connected playback devices do not stream content directly from the remote computing devices 106b but instead receive audio from the mobile device 704. Some playback devices 102 may have a certain “portable” mode in which they are connected to the mobile device 704 via BLUETOOTH rather than to the LAN 111.

[0120] Further, some streaming media service app(s) 762 provide a protocol to allow a user to select a playback device (e.g., one of the playback devices 102) as the active playback device for playback. For instance, SPOTIFY CONNECT allows a user of the SPOTIFY streaming media service app 762 to select a playback device that supports SPOTIFY CONNECT (e.g., the playback devices 102). When such a playback device is selected, instead of streaming audio to the mobile device 704, the remote services 106b of the SPOTIFY streaming audio service stream the audio to the selected playback device for playback. The SPOTIFY streaming media service app 762 then acts as a remote to the active playback device.

[0121] If a SPOTIFY playback session is on-going on the mobile device 704, selecting another playback device allows the user to transition that session to that playback device. However, such transitions are internal to the SPOTIFY CONNECT platform and the devices that support that platform. Features supported by the native domain that are not part of the SPOTIFY CONNECT domain may be unavailable when working in the SPOTIFY connect domain.

[0122] The media player app(s) 763 provide playback features for local content in the memory 713 of the mobile device 704. Some media player app(s) 763 allow playback of content in network-attached storage (e.g., attached to the LAN 111 (FIG. 1B)). Example media player app(s) 763 include APPLE iTunes and VLC media player.

[0123] The software 714 further includes one or more streaming protocol services 764. Each streaming protocol service 764 is software that supports a streaming protocol, such as AIRPLAY 2 or GOOGLE CAST, which allows users to stream media from the mobile device 704 to a supported playback device. In some examples, the operating system 760 includes media playback controls, which include a streaming protocol control to select a target playback device. For instance, the iOS Control Center includes an AIRPLAY 2 control to select among compatible AIRPLAY 2 devices on the LAN 111 as streaming targets. As another example, the CHROME browser includes a GOOGLE CAST control select among compatible GOOGLE CAST devices on the LAN 111 as streaming targets. In other examples, certain apps, such as the streaming media service app(s) 762 and/or the media player apps 763 include streaming protocol support. For instance, the iTunes app includes an AIRPLAY 2 control to select among compatible AIRPLAY 2 devices on the LAN 111 as streaming targets.

[0124] As further shown in FIG. 7A, the mobile device 704 also includes power components 727. The power components 727 include at least an external power source interface 728 and an internal power source 729 (e.g., one or more batteries) configured to power the mobile device 704 without a physical connection to an external power source. An external power source interface 728 may be coupled to a power source (not shown) via a USB cable or the like that physically connects the mobile device 704 to a charger and electrical outlet or some other external power source. The external power source interface 728 may be configured to facilitate charging the internal power source 729.

[0125] As an illustrative example, FIG. 7B shows a front view of the mobile device 704. As shown in FIG. 7B, the NMD 730 includes a housing 730. The housing 730 may carry one or more components shown in FIG. 7A. For instance, the user interface 740 includes a touch-sensitive graphical display carried on a front portion of the housing 730, as shown. Further, the housing 730 is also carrying a speaker 718 and a microphone 722 on its front portion.

IV. Example Playback Transition Scenarios

[0126] FIG. 8A illustrates an exemplary first scenario in which a playback session transition between platforms is triggered via a “tap” event between the mobile device 704 and the playback device 102c (FIG. 1A). In the first scenario, a user may start a first playback session using a particular streaming media service app 762a corresponding to a particular streaming media service. In this first playback session, the streaming media service app 762a streams data representing audio content from one or more servers of the particular streaming media service (e.g., the computing devices 106b (FIG. 1B)) and plays back that audio content on the mobile device 704 (e.g., using the internal speakers 718 or an externally-connected speaker, earbuds, or headphones). In this example, the streaming media service app 762a is a first domain.

[0127] While the first playback session is on-going on the mobile device 704, the user may desire to transition play-

back from the mobile device **704** to the playback device **102c**. For instance, the user was previously listening while “on-the-go” with their mobile device **704**, and desires to transition to listening out-loud using the playback device **102c**. If the streaming media service app **762a** supports a protocol (such as SPOTIFY CONNECT) that allows changing the active playback device within the first domain, the user may opt to transition playback from the mobile device **704** to the playback device **102c** using this protocol by selecting the playback device **102c** as the active playback device within the streaming media service app **762a**, which maintains the first playback session in the first domain, albeit on a different playback device.

[0128] However, since the first domain is non-native to the playback device **102c**, the first domain might not support all features of the playback device **102c**. For instance, the streaming media service app **762a** might not support grouping multiple playback devices **102a-102o** into synchrony groups. As such, the user may instead desire to transition playback from the first platform to a second platform native to the playback device **102c**. As noted above in connection with FIG. 7A, the MPS control app **761** controls the playback device **102c** in its native domain.

[0129] Accordingly, the user may bring the mobile device **704** into proximity of the playback device **102c** to generate a tap event between the mobile device **704** and the playback device **102c**, which causes a playback transition between a source platform (i.e., the streaming media service app **762**) and a target platform (i.e., the MPS control app **761**). In particular, while the first playback session is on-going via the streaming media service app **762a**, the MPS control app **761** monitors for tap events. When the user brings the mobile device **704** into proximity of the playback device **102c** to generate the tap event between the mobile device **704** and the playback device **102c**, the MPS control app **761** detects this event and initiates the playback session transition from the source domain to the target domain (i.e., the native domain for the MPS **100**).

[0130] In some implementations, the tap event is generated using a short-range wireless protocol such as BLUETOOTH LOW ENERGY (BLE). For example, the playback device **102c** may emit a BLE beacon using a BLUETOOTH (802.15-compatible) interface of the wireless network interface **225** (FIG. 2A). This BLE beacon is detectable using a BLUETOOTH (802.15-compatible) interface of the wireless network interface **725** in the mobile device **704** (FIG. 7A). Detection of the low-energy BLE beacon at a certain threshold signal strength by the mobile device **704** indicates that the mobile device **704** is in close proximity to the playback device **102c** (i.e., that a tap event is occurring).

[0131] Alternatively, the tap event is generated using near-field communication (NFC). The playback device **102c** may have a NFC tag on its housing **230** (FIG. 2B) and the mobile device **704** may have an NFC reader. When the mobile device **704** is placed in proximity to the NFC tag on the playback device **102c**, the NFC tag is read and a tap event is generated.

[0132] Since the MPS control app **761** is a different platform from the streaming media service app **762a**, a playback session transition is effected by reproducing the first playback session in the source domain using a second playback session in the target domain. This reproduction results in an apparent playback session transition from the

perspective of the user. To be able to reproduce the first playback session using a second playback session in the target domain, the MPS control app **761** needs to determine the target playback device(s) to start the second playback session on, as well as the currently playing audio content and play position within that content in the first domain.

[0133] Since a tap event involves two devices, the tap event itself facilitates determining the source and target of the playback session transition. That is, when a tap event is generated, one of the two devices involved in the tap event is the source (i.e., the device with the first playback session, which is the mobile device **704** in the first scenario). The other device involved in the tap event is the target (i.e., the device without an on-going playback session, which is the playback device **102c**). If both devices have active playback sessions, the mobile device **704** may determine the source and target based on a pre-determined setting or default.

[0134] In the first scenario, to identify the target device (i.e., the playback device **102c**) to the source device (e.g., the mobile phone **704a**), the tap event may involve transmitting an identifier that distinguishes the playback device **102c** from the playback devices **102b-o**. For instance, in the BLE example above, the BLE beacon emitted by the playback device **102c** may include an identifier that distinguishes the playback device **102c** from the playback devices **102a-b** and **102d-o**. Similarly, the NFC tag on the housing **230** of the playback device **102c** may store such an identifier, which is read by the mobile device **704** during the tap event.

[0135] The MPS control app **761** may store or have access to data correlating this identifier to the playback device **102c**. For instance, the memory **713** of the mobile phone **704** may include a table or other data structure correlating the identifier with configuration information of the playback device **102c**, such as its internet protocol (IP) address, media access control (MAC) address, zone name, and the like. When the mobile device **704** receives the identifier of the playback device **102c**, the MPS control app **761** may use this identifier to reference this configuration information.

[0136] As noted above, to be able to reproduce the first playback session using a second playback session in the target domain, the MPS control app **761** needs to determine the currently playing audio content and play position within that content in the first domain in addition to the target playback device(s). One possible source of this information is the servers of the particular streaming media service corresponding to the streaming media service app **762a** (e.g., the computing devices **106b** (FIG. 1B)). Another possible source is the streaming media service app **762a** itself.

[0137] In some examples, the particular streaming media service may make playback session information available using an application programming interface (API). In such examples, the MPS control app **761** may query the playback session information using one or more function calls to the API. Then, playback session data may be transferred between the streaming media service app **762a** and the MPS control app **761** using an inter-process communication mechanism implemented by the operating system **760**. A possible issue with this approach is that it requires the streaming media service to make such an API available, and further the MPS control app **761** may need to be updated to maintain compatibility with the API.

[0138] Accordingly, the MPS control app **761** may instead access the playback session information from the servers of the particular streaming media service corresponding to the

streaming media service app 762a. As noted above, to facilitate playback of streamed audio on the playback devices 102a-o, a user may register their MPS 100 (FIG. 1A) with their user account information for one or more streaming media services, including the particular streaming media service corresponding to the streaming media service app 762a. The MPS control app 761 may be authenticated with the streaming media services using this user account information, so that users of the MPS control app 761 can browse and/or search for media information from the streaming media services to play on the playback devices 102a-o. Further, streaming media services generally maintain a playback history of media items streamed by a given user account. As such, because the user has already provided the MPS control app 761 with the user account information for their subscribed streaming media services, the streaming media service app 762a may query the subscribed streaming media services for the audio track(s) that the user has streamed most recently. In other words, by virtue of users having registered their subscribed services with the MPS 100 (and thereby providing the MPS 100 with access to the user accounts of the various subscribed services), the MPS 100 and its associated devices (e.g., the mobile device executing the MPS control app 761 to function as a control device 104) are positioned to transfer playback sessions across platforms.

[0139] Since the first playback session is happening in a different domain, the MPS control app 761 may not know which streaming media service is involved in the first playback session. As such, to determine the playback session information, the MPS control app 761 may query multiple streaming audio services for playback history data associated with respective user accounts of the streaming audio services. Then, based on the queried playback history data, the MPS control app 761 may determine which streaming media service had an active playback session. For instance, the MPS control app 761 may look for the most recent time stamp associated with a streamed audio track. Various streaming media services may make available APIs to facilitate querying for playback history data.

[0140] Generally, APIs to interact with streaming media service servers are relatively available (as compared with APIs to access data via IPC) to facilitate access by third party playback devices to streaming media services. For example, SPOTIFY offers a Web API that includes functions allowing authorized third parties (i.e., those having a valid authorization token, such as the MPS control app 761) to query for user information. In particular, the Web API includes a function call to get a user's current playback information (curl-X GET "https://api.spotify.com/v1/me/player"-H "Authorization: Bearer {your access token }"), which returns structured data indicating which device is active (i.e., the source device), the currently playing track, the progress within the track, as well as other related information, such as an associated playlist that the currently playing track that the currently playing track is in (if any). Other streaming media services may make available APIs with equivalent or similar functions.

[0141] After getting the playback session information, the MPS control app 761 creates a second session in the target (native) domain. For instance, the MPS control app 761 may send instructions via the network interface 724 of the mobile device 704 to cause the playback device 102c to queue the currently playing audio track of the first playback session

(and possibly additional audio tracks, such as those in a playlist with the currently playing audio track) in a queue in the memory 213 of the playback device 102c. As another example, the MPS control app 761 may send instructions to the remote computing devices 106c (e.g., one or more SONOS cloud servers) to cause the remote computing devices 106c to queue the currently playing audio track of the first playback session in a cloud queue associated with the playback device 102c. Additional details regarding cloud queues are described in U.S. patent application Ser. No. 14/616,310, filed Feb. 6, 2015, titled "Cloud Queue Access Control," and U.S. patent application Ser. No. 14/616,332, filed Feb. 6, 2015, titled "Cloud Queue Synchronization Protocol," which are incorporated herein by reference in their entirety. Other examples are possible as well.

[0142] Using the streaming media service APIs, the MPS control app 761 may finish the playback session transition by stopping the first playback session in the first domain. For instance, the MPS control app 761 may send instructions representing a function call to stop the session or pause playback, among other examples. Alternatively, the first playback session may stop automatically by the particular streaming media service when the second playback session is started in the target domain, since many streaming media services permit each user account to have only a single active stream.

[0143] In some cases, the MPS control app 761 may facilitate a fade-in to the second session and/or a fade-out from the first session. If the first session or the second session is in the native domain of the MPS control app 761, the MPS control app 761 may control the target playback device(s) directly to fade-in or fade-out. For playback sessions in a non-native domain, the MPS control app 761 may cause fade-in or fade-out using an API. Alternatively, such domains may automatically fade-in or fade-out, possibly based on user preferences configured in that domain.

[0144] For some types of content, the MPS control app 761 might not start the second playback session in the second domain at the same playback position as the first playback session in the first domain. For instance, for streaming radio and other types of live content, the MPS control app 761 may start playback in the second playback domain at the "live position." On the other hand, for offline content like podcasts or audio books, the MPS control app 761 will start the second playback session in the second domain at the same playback position as the first playback session in the first domain.

[0145] In some implementations, after detecting a tap event, the MPS control app 761 may obtain confirmation that a playback session transition should be performed. FIG. 8B shows the mobile device 704 displaying a graphical notification 865. In some examples, the graphical notification 865 may be a feature of the operating system 760 (FIG. 7A) and may be displayed based on a function call to an API of the operating system 760. As such, the MPS control app 761 need not be active and may instead be executing in a background or suspend mode.

[0146] The graphical notification 865 includes a textual prompt indicating that the TAP event between the playback device 102c (in the patio 101i (FIG. 1A)) and the mobile device 704 was detected (FIG. 8A) and asking if the user would like to transfer the SPOTIFY session (e.g., the first playback session, initiated via the particular streaming media service application 762a. The graphical notification

865 further includes two selectable prompts **865a** and **865b**. The selectable prompt **865a**, when selected, causes the MPS control app **761** to perform the playback session transition described in the first scenario. In contrast, when the selectable prompt **865b** is selected, the MPS control app **761** foregoes performing the playback session transition.

[**0147**] FIG. 8C is a message flow diagram illustrating instructions exchanged between the mobile device **704**, the playback device **102c**, and the remote computing devices **106b** (i.e., servers of one or more streaming audio services) during the playback session transition of the first scenario illustrated in FIGS. 8A and 8B. Such messages are representative and may include additional or fewer messages.

[**0148**] At **871**, the MPS control app **761** executing on the mobile device **704** detects a tap event between the mobile device **704** and the playback device **702**. To obtain playback session information, at **872**, the MPS control app **761** queries the remote computing devices **106b** of one or more streaming media services to obtain playback history information for respective user accounts. Each query may conform to an API provided by a respective streaming audio service to get access to playback history information. The queries may identify the user accounts of the various streaming media services that are registered with the MPS **100** (e.g., using respective authorization tokens, user names, etc.).

[**0149**] At **873**, the remote computing devices **106b** of the one or more streaming media services provide respective responses to the queries sent by the mobile device **704** indicating respective playback history information associated with the registered user account of each streaming media service. At **874**, the MPS control app **761** identifies the first playback session based on the received playback history information.

[**0150**] In identifying the first playback session, the MPS control app **761** extracts or otherwise determines the currently playing audio track and offset within that track from the playback history information. If the mobile device **704** playback history information from multiple streaming audio services, the MPS control app **761** compares the playback history information to determine which user account was most recently active (i.e., that is currently active) to determine which streaming media service the user is currently using in the first playback session. Since the first playback session is occurring via the streaming media service app **762a** of the particular streaming media service, the comparison of the playback history information indicates that the user's session is via the particular streaming media service.

[**0151**] At **875**, to initiate a second session in the native domain, the MPS control app **761** sends a message to the playback device **102c** that includes a command to start playing back the currently playing audio track (in the first session). The message also includes a uniform resource identifier identifying a source of the currently playing audio track (in the first session) at the servers **106b** and an offset indicating the current play position within the currently playing audio track. If the currently playing audio track is part of a playlist, the MPS control app **761** may also send instructions to queue audio tracks of the playlist in a queue on the playback device **102c** (or in a cloud queue associated with the playback device **102c**).

[**0152**] At **876**, based on receiving the message from the mobile device **704** with the audio track URI and the offset, the playback device **102c** sends a fetch message to the

remote computing device **106b** to request a stream of the audio track from the remote computing device **106b**. Based on this fetch message, at **877**, the remote computing device **106b** streams the audio track beginning at the offset to the playback device **102c** for playback. By starting the stream to the playback device **102c**, the remote computing device **106b** automatically stop the first session using the streaming media service app **762a** because the user account of the particular streaming media service is permitted a single concurrent stream.

[**0153**] In a variation of the first scenario, the user may start a playback session using a particular streaming media service app **762a** on the mobile device **704** but select another playback device (e.g., the playback device **102b**) as the active playback device using a protocol such as SPOTIFY CONNECT. As noted above, in such cases, the particular streaming media service app **762a** functions as a remote control for the playback device **102b**. In example implementations, a tap event between the mobile device **704** and the playback device **102c** may similarly trigger a playback session transition. Since the user account registered with the streaming media service app **762a** is still being utilized in the playback session on the playback device **102b**, if the MPS control app **761** queries the particular streaming media service for currently playing media, the query will return playback session information associated with the playback session on the playback device **102b**. The MPS control app **761** can then create a second session in the native domain on the playback device **102c**.

[**0154**] In another variation of the first scenario, the user may start a first playback session in the native domain on the playback device **102c** using the MPS control app **761** and then decide to transition to a second domain using the tap event illustrated in FIG. 8A. Since the MPS control app **761** already has the first playback session info (currently playing track, offset, playlist, etc.), the MPS control app **761** can transition to a second playback session in a second domain by instructing an entity in the second domain to create the second playback session using the first playback session info (and then end the first playback session). For instance, to start a second playback session in the SPOTIFY domain, the MPS control app **761** may call into the SPOTIFY Web API with one or more functions to create a new session with the currently playing track and to seek to the current offset within that track.

[**0155**] In an exemplary second scenario, a user may start a first playback session that utilizes the streaming protocol service to stream audio content from a first device to a second device. FIG. 9A illustrates a first playback session that utilizes the streaming protocol service **764** (FIG. 7A) to stream audio content between the mobile device **704** and the playback device **702c**. In FIG. 9A, the audio content is streamed using the LAN **111** (FIG. 1B).

[**0156**] The streaming protocol service **764** defines a "sender" that streams audio content and a "receiver" that receives the audio content from the sender over the LAN **111** (FIG. 1B) and plays the audio content back. These types of streams may be streams provided using a real-time streaming protocol (e.g., RTSP). To support the streaming protocol service **764**, each of the sender and the receiver include hardware and/or software configured to send and/or receive data according to the streaming protocol service **764**. As noted above, examples of the streaming protocol service include APPLE AIRPLAY2 and GOOGLE CAST. In the

FIG. 9A example, the mobile device **704** and the playback device **102a** are operating as the sender and the receiver, respectively, of the streaming protocol service **764**.

[0157] Generally, the streaming protocol service **764** allows a user to stream any audio from the sender to the receiver. For instance, in the FIG. 9A example, the audio source is a media player app **763a**, which may play local audio content from the memory **713** (FIG. 7A), or from a network source on the LAN **111** or the networks **107** (FIG. 1B). In further examples, the audio source may be one of the streaming media service app(s) **762** (FIG. 7A), which may stream audio from the remote computing devices **106b** (FIG. 1B).

[0158] Notably, the native domain of the MPS **100** may provide similar or equivalent features as the streaming protocol service **764**, albeit on a different platform than the streaming protocol service **764**. In particular, the native domain may similarly provide a native domain streaming protocol. Using the native domain streaming protocol, the MPS control app **761** may stream audio to the playback devices **102** of the MPS **100**. Alternatively, as described above, the MPS control app **761** may control the playback devices **102** to stream audio directly from an audio source, rather than having the mobile device **704** function as an intermediary. While the playback device **102a** is part of the first playback session in the domain of the streaming protocol service **764**, the streaming protocol service **764** may functionally “step-in” or replace the native domain features, which are then not in use.

[0159] For various reasons, a user may desire to transition the first playback session in the domain of the streaming protocol service **764** to the native domain of the MPS **100**. For instance, as noted above, since the streaming protocol service **764** is controlling the playback device **102a** during the first playback session, certain native domain features may be unavailable. To regain access to those native domain features (without disrupting playback), a user may desire to transition the first playback session in the domain of the streaming protocol service **764** to the native domain of the MPS **100**.

[0160] In some examples, a user may trigger a playback session transition from the domain of the streaming protocol service **764** using user input. FIG. 9B shows the mobile device **704** displaying a graphical notification **965**. Similar to the graphical notification **865**, the graphical notification **965** may be a feature of the operating system **760** (FIG. 7A) and may be displayed based on a function call to an API of the operating system **760**. Other triggers, such as the tap event, may be implemented as well.

[0161] The graphical notification **965** includes a textual prompt indicating that the first playback session in the domain of the streaming protocol service **764** (e.g., AIRPLAY 2 session) is on-going. The textual prompt of the graphical notification **965** further asks if the user would like to transfer the AIRPLAY 2 session to the native domain of the MPS **100** (e.g., SONOS). The graphical notification **965** further includes two selectable prompts **865a** and **865b**. The selectable prompt **965a**, when selected, triggers the MPS control app **761** to transition first playback session to the native domain of the MPS **100**. In contrast, when selected, the selectable prompt **965b** foregoes performing the playback session transition.

[0162] Since the first playback session in the domain of the streaming protocol service **764** is taking place on the

playback device **102a**, which is part of the MPS **100**, the MPS control app **761** may get playback session information for the first playback session from the playback device **102a**. As discussed in connection with FIG. 2A, the memory **213** of each playback devices **102** may maintain state variables indicating the current state of the playback device(s) **102**, including current playing audio tracks, offsets, and the like. These state variables kept synchronized across the MPS **100**, including in the control devices **104** (i.e., the mobile phone **704** executing the MPS control app **761**). Using this data, the MPS control app **761** may start a second playback session.

[0163] To complete the playback session transition, the first playback session is stopped. In some implementations, the streaming protocol service **764** may automatically stop the first playback session when the MPS control app **761** takes over control of the playback device **102a** by starting the second playback session. Alternatively or additionally, if the audio source is a streaming audio service, the streaming audio service may automatically stop the first playback session when the playback device **102a** starts streaming in the second playback session due to limitations in the number of concurrent audio streams. Yet further, the MPS control app **761** may send instructions to the streaming protocol service **764** to stop the first playback session, perhaps using a function call of an API of the streaming protocol service **764**.

[0164] In a variation of the second scenario, a user may desire to transition a native domain playback session to the domain of the streaming protocol service **764**. For instance, the user may have devices from multiple manufactures, which each support the streaming protocol service **764**, but have respective native domains. As such, to transition a playback session from one or more playback devices **102** of the MPS **100** to a third-party playback device, the user may desire to transition a native domain playback session to the domain of the streaming protocol service **764**.

[0165] In this variation, since the first playback session in the native domain of the MPS **100**, the MPS control app **761** already maintains or has access to playback session information for the first playback session from the playback device **102a**. To start a second playback session in the domain of the streaming protocol service **764**, the MPS control app **761** may send instructions to the streaming protocol service **764**. The instructions may be internal to the mobile device **704** (e.g., via an IPC mechanism) or via a server of the streaming protocol service **764** and/or the MPS **100** (e.g., the computing device **106c** (FIG. 1B)).

[0166] In another variation of the second scenario, a user may desire to transition a playback session in the domain of the streaming protocol service **764** to a domain of a streaming media service. For instance, a user may be playing a particular audio track from a network-attached storage on the LAN **111** on the playback device **102a** in a first playback session in the domain of the streaming protocol service **764**. To take this playback session on-the-go, the user may desire to transition the first playback session to play back on the mobile device **704** using the streaming media service app **762a**.

[0167] Since this playback session in the domain of the streaming protocol service **764** is taking place on the playback device **102a**, which is part of the MPS **100**, the MPS control app **761** may get playback session information for the first playback session from the playback device **102a**. The MPS control app **761** may then start a playback session

in the domain of the particular streaming media service using an API of the streaming media service app 762a. For instance, the SPOTIFY Web API includes a function called find-and-play, which allows searching for and playing an audio track based on a uniform resource identifier and/or metadata.

[0168] In some instances, a user may desire to take a native domain playback session or a streaming protocol service 764 occurring on a playback device 102 “on-the-go.” When these domains utilize wireless local area networks to stream audio content, such domains are generally unavailable when the user is outside of the range of the LAN 111. Accordingly, the user may desire to transition a native domain playback session or a streaming protocol service 764 occurring on a playback device 102 to a playback session on the mobile device 704, which the user can take with them and listen using speakers 718 and/or an externally-connected speaker, headphones, or earbuds connected via BLUETOOTH or line-in.

[0169] FIG. 10 illustrates an exemplary third scenario in which a playback session transition between platforms is triggered via a network disconnection event on the mobile device 704. In particular, a user may start a native domain playback session on the playback device 102g that involves playing back a particular audio track from a particular streaming audio service. To play the audio track, the playback device 102g streams data representing the particular audio track from the computing devices 106b of the particular streaming audio service via the router 109 and the LAN 111.

[0170] When the user carries the mobile device 704 away from the home environment including the MPS 100 (FIGS. 1A and 1B), the mobile device 704 disconnects from the LAN 111 when the LAN 111 becomes out-of-range. The MPS control app 761 (or other software, such as a background process or agent) may detect this network disconnection event. Since this network disconnection event indicates that the user is leaving the home environment, the MPS control app 761 may facilitate a playback session transition from the playback device 102g and to the mobile device 704.

[0171] When the mobile device 704 detects the network disconnection event, the MPS control app 761 may obtain confirmation that a playback session transition should be performed. FIG. 10 further shows the mobile device 704 displaying a graphical notification 1065. The graphical notification 1065 includes a textual prompt indicating that the user is leaving the LAN 111 and asking if the user would like to transfer the native domain playback session to the mobile device 704. The graphical notification 1065 further includes two selectable prompts 1066a and 1066b. The selectable prompt 1066a, when selected, causes the MPS control app 761 to perform a playback session transition from the native domain to a domain available locally on the mobile device 704. In contrast, when the selectable prompt 1065b is selected, the MPS control app 761 foregoes performing the playback session transition.

[0172] Since, in the third scenario, the playback device 102g was playing back the particular audio track from a particular streaming audio service, the MPS control app 761 may create a second playback session in the domain of the particular streaming audio service. As described above in connection with other examples, exemplary streaming audio services provide respective APIs allowing a third-party, such as the MPS control app 761, to create playback session in

their domain. For instance, a function call to SPOTIFY API can create a SPOTIFY domain playback session in the SPOTIFY app (known as a SPOTIFY Deep Link). Further, since the first playback session is in the native domain, the MPS control app 761 maintains or has access to data indicating that the playback device 102g was playing back the particular audio track from a particular streaming audio service. After creating the second playback session, the mobile device 704 streams the audio content via the networks 107 from the remote computing device 106b, as shown in FIG. 10.

[0173] In a variation of the third scenario in FIG. 10, a playback session transition may be triggered by a network connection event. For instance, when the user returns home with the mobile device 704 and reconnects to the LAN 111, the mobile device 704 may detect this network connection event and trigger a playback session transition from a domain local to the mobile device 704 (e.g., the domain of the particular streaming service) to a domain that can be used to stream audio content to one or more of the playback devices 102 (e.g., the native domain or the streaming protocol service 764 domain). Then the user can select the playback device 102 to continue the playback session on (e.g., via user input or a TAP event (FIG. 8A)).

[0174] The MPS control app 761 may detect a network connection event using various techniques. In some examples, the MPS control app 761 stores the service set ID (SSID) of the LAN 111 in local data storage or in the cloud as the network interconnecting the media playback system 100. Then, the MPS control app 761 may query or otherwise obtain the current network status from the operating system 760. In further examples, when the mobile device 704 connects to the LAN 111, the MPS control app 761 may send out a broadcast discovery message on the LAN 111, which may cause the playback devices 102 to respond with one or more messages that indicate to the MPS control app 761 that mobile device 704 is connected to the LAN 111. Alternatively, the playback devices 102 may periodically transmit broadcast discovery messages on the LAN 111, which the mobile device 704 may receive once connected to the LAN 111. Other network connection techniques involve local discovery using BLUETOOTH (e.g., a BLE connection) or one or more audio tones (e.g., an ultrasonic tone). Further examples may use a combination of such techniques, or any other suitable technique.

[0175] In another variation of the third scenario in FIG. 10, as an alternative to triggering a playback session transition based on a network disconnection event when the user leaves the range of the LAN 111, the playback session transition may be triggered based on a network connection event. For instance, when the mobile device 704 connects via BLUETOOTH to a car stereo receiver in the user's automobile, the mobile device 704 may trigger the playback session transition of the third scenario. Alternatively, some cars include an infotainment system that provides a Wi-Fi hotspot using a cellular modem. Connecting to such a Wi-Fi hotspot may trigger a playback session transition.

[0176] In various examples, the MPS control app 761 may detect a network connection event or disconnection event based on data from the network interface 724. For instance, when the network interface 724 connects to or disconnects from a network, the network interface 724 may generate an interrupt that notifies the operating system 760 of network connection or disconnection. The MPS control app 761

(directly or via a background process) may monitor network connection status information maintained by the operating system **760**.

[0177] In practice, multiple playback session transitions across domains using various triggers may facilitate a user maintaining a playback session on various devices as they go about their day. For instance, a user may start a first playback session in the native domain on the playback device **102h** in the master bathroom **101a** while getting ready. After the user leaves the house on foot for work, a network disconnection event triggers a playback session transition (FIG. 10), which creates a second playback session in a domain of a particular streaming media service that reproduces the first playback session in the native domain on the playback device **102h** in the master bathroom **101a**. From the perspective of the user, the playback session has transitioned seamlessly to the mobile device **704** to facilitate listening via earbuds or headphones during the morning commute.

[0178] After work, the user returns home and triggers a playback session transition back to the native domain via a tap event on the playback device **102i** in the Kitchen **101h**. This transition creates a third playback session, reproducing the second playback session in the domain of the particular streaming media service in the native domain. After dinner, the user receives an invite from a friend to visit. Upon turning on his car, a network connection event triggers another playback session transition, which creates a fourth playback session that reproduces the third playback session in the domain of the particular streaming media service.

[0179] After arriving at the friend's house, the user desires to share with the friend some of the audio tracks that the user has been enjoying throughout the day. Since the friend has a third-party playback device that supports the streaming protocol service **764**, the user connects to the friend's Wi-Fi network using the mobile device **704** and provides user input to trigger a playback session transition to the domain of the streaming protocol service **764** (FIG. 9B). This playback session transition creates a fifth playback session, which reproduces the fourth playback session in the domain of the streaming protocol service **764**. The user then selects the third-party playback device as the active playback device (i.e., the receiver) and the mobile device **704** becomes the sender.

[0180] Upon returning home, the user triggers a playback session transition back to the native domain via a tap event on the playback device **102g** in the Master Bedroom **101b**. This transition creates a sixth playback session, reproducing the fifth playback session in the domain of the streaming protocol service **764** in the native domain.

[0181] As described in connection with FIGS. 1A and 1B, the MPS **100** may include one or more portable playback devices, such as the headphones **102o**. Such portable devices may operate in one of two modes. In a first mode, the headphones **102o** connect to the LAN **111** and operate similarly to the other playback devices **102**. In a second mode, the headphones **102o** connect directly to a computing device (e.g., the mobile device **704**) using BLUETOOTH.

[0182] A portable playback device, such as the headphones **102o**, may be a source or target for a playback session transition. For instance, a user may use a tap event between the mobile phone **704** and the headphones **102o** to trigger a playback session transition. Moreover, since the headphones **102o** are portable, the headphones **102o** may

trigger playback session transitions in a similar manner as the mobile device **704**. For instance, a network connection or disconnection event on the headphones **102o** may trigger a playback session transition. Further, a tap event between the headphones **102o** and another one of the portable playback devices **102** may trigger a playback session transition between those devices.

V. Example Playback Session Transition Techniques

[0183] FIG. 11 is a flow diagram showing an example method **1100** to facilitate a playback session transition between domains. The method **1100** may be performed by a networked microphone device, such as the mobile device **704** (FIG. 7A). Alternatively, the method **1100** may be performed by any suitable device or by a system of devices, such as the playback devices **102**, NMDs **103**, control devices **104**, computing devices **105**, computing devices **106**, and/or mobile device **704**.

[0184] At block **1102**, the method **1100** includes detecting a first event as a trigger to transfer playback to a second domain. For example, while a first playback device is playing back audio content during a first playback session in a first domain, the mobile device **704** (FIG. 7A) may detect a first event which is configured as a trigger to transfer playback to a second domain on a second playback device. Example trigger events include a tap event (FIG. 8A), a particular user input (FIG. 9B), and a network connection or disconnection event (FIG. 10), as well as other suitable events. The first playback device may be one of the playback device(s) **102** of the MPS **100** or a third-party playback device.

[0185] At block **1104**, the method **1100** includes determining playback information for a first playback session in a first domain. For instance, in response to detecting the first event, the MPS control app **761** on the mobile device **704** may determine that the first playback device was playing particular audio content in the first playback session and that the first playback device was at a given playback position in the particular audio content.

[0186] In some instances, the first playback device may be playing audio content from a network location at one or more servers of a particular streaming audio service. If the first domain is a native domain, the MPS control app **761** may already have the playback information (e.g., from the servers or the playback device). If the first domain is a non-native domain, the MPS control app **761** may query multiple streaming audio services for playback history data associated with respective user accounts of the streaming audio services. The MPS control app **761** may be authenticated with the multiple streaming audio services as these user accounts of the streaming audio services are registered with the MPS **100**.

[0187] Based on the queried playback history data, the MPS control app **761** may determine that the particular audio content was being played back when the first event was detected. In some cases, the particular audio content is associated with a particular identifier identifying the network location at the one or more servers of the particular streaming audio service. For instance, exemplary streaming audio services may assign each track a uniform resource identifier (URI) identifying the network location at the one or more servers of the particular streaming audio service.

[0188] At block 1106, the method 1100 includes initiating a second playback session in a second domain based on the first playback session in the first domain. For example, the mobile device 704 may send, via a network interface, instructions to initiate a second playback session in the second domain to play back the particular audio content on the second playback device. The second playback device may be one of the playback device(s) 102 of the MPS 100 or a third-party playback device.

[0189] The instructions may indicate the network location at the one or more servers of the particular streaming audio service and an offset in the particular audio content representing the given playback position, which allows the second playback device to reproduce the first playback session in the second domain. If the particular audio content is associated with a particular identifier identifying the network location at the one or more servers of the particular streaming audio service, the instructions may further indicate the particular identifier.

[0190] In some examples, the first domain is a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), such as the streaming protocol service 764 (FIG. 7A). In such examples, the mobile device 704 may be configured as the sender to configure the first playback device as the receiver to stream the particular audio content from the network location at the one or more servers during the playback session in the first domain. Alternatively, the second domain may be the wireless streaming protocol.

[0191] Further, in such examples, the second domain may be a native control protocol of the MPS 100. In such examples, sending the instructions to initiate the playback session in the second domain includes a native control application executing on the mobile device 704 (i.e., the MPS control app 761) sending instructions according to the native control protocol to (i) populate a queue associated with the second playback device with the particular audio content and (ii) start playback at the offset. Alternatively, the first domain may be the native control protocol of the MPS 100.

[0192] Alternatively, the second domain may be a streaming audio service control protocol and a streaming audio service application (e.g., the streaming media service app(s) 762a) (FIG. 8A). In such examples, sending the instructions to initiate the playback session in the second domain may include the streaming audio service control application 762a executing on the mobile device 704 sending instructions according to the streaming audio service control protocol to the second playback device to play back the particular audio content starting at the offset. The mobile device 704 may send the instructions directly (e.g., over the LAN 111) or via one or more servers (e.g., the remote computing devices 106bA) over the LAN 111 and the networks 107.

[0193] Yet further, the second playback device may be an 802.15-compatible (BLUETOOTH) portable playback device, headphones, earbuds, car stereo receiver, or other playback device. In such examples, sending the instructions to initiate the playback session in the second domain may include the native control application (i.e., the MPS control app 761) executing on the mobile device initiating the playback session in the second domain locally on the mobile device 704. The MPS control app 761 may send, via an 802.15-compatible network interface, a data stream representing the particular audio content. The data stream may

begin at the offset in the particular audio content. Then, receiving the data stream causes the second playback device to begin playback of the particular audio content at the offset.

[0194] In some examples, a playback session transition may involve the first event and a second event. For example, detecting the first event may include detecting that the mobile device has left the proximity of the first playback device (e.g., via a network disconnection event (FIG. 10)). The mobile device 704 may then detect a second event corresponding to a trigger to initiate the playback session in the second domain. Detecting the second event may involve detecting that the mobile device has entered the proximity of the second playback device (e.g., via a network connection event, such as forming an 802.15 connection to a car stereo receiver or a wearable device). In such examples, sending the instructions to initiate the playback session in the second domain may involve sending the instructions to initiate the playback session in the second domain in response to detecting the second event.

[0195] In other examples, the first domain includes a streaming audio service control protocol and a streaming audio service application (e.g., the streaming media service app(s) 762a) (FIG. 8A). To initiate the first playback session on the first playback device, the streaming media service app 762a may send instructions according to the streaming audio service control protocol to initiate the first playback session in the first domain on the first playback device.

[0196] In such an example, second domain may be the native control protocol of the MPS 100. Then, sending the instructions to initiate the playback session in the second domain may include a native control application (i.e., the MPS control app 761) executing on the mobile device 704 sending instructions according to the native control protocol to (i) populate a queue associated with the second playback device with the particular audio content and (ii) start playback at the offset. As another example, sending the instructions to initiate the playback session in the second domain may include the native control application associated with a particular user account of a native cloud service executing on the mobile device sending, to one or more servers of the native cloud service (e.g., the remote computing devices 106c), instructions according to the native control protocol (i) populate a cloud queue associated with the second playback device with the particular audio content and (ii) start playback on the second playback device at the offset.

[0197] FIG. 12 is a flow diagram showing an example method 1200 to facilitate a playback session transition between domains. The method 1200 may be performed by a networked microphone device, such as the mobile device 704 (FIG. 7A). Alternatively, the method 1200 may be performed by any suitable device or by a system of devices, such as the playback devices 102, NMDs 103, control devices 104, computing devices 105, computing devices 106, and/or mobile device 704.

[0198] At block 1202, the method 1200 includes initiating a first playback session in a first domain. For instance, the MPS control app 761 on the mobile device 704 may send, via a network interface, instructions to initiate a playback session in a native domain of the MPS 100 to play back particular audio content on a first playback device 102 of the MPS 100. Alternatively, the first playback session may be

initiated in the streaming media service domain or the streaming protocol service domain, among other possible examples.

[0199] At block 1204, the method 1200 includes detecting a first event as a trigger to transfer playback to a second domain. For example, while the first playback device is playing back audio content during the first playback session in a first domain, the mobile device 704 (FIG. 7A) may detect a first event which is configured as a trigger to transfer playback to a second domain on a second playback device. Example trigger events include a tap event (FIG. 8A), a particular user input (FIG. 9B), and a network connection or disconnection event (FIG. 10), as well as other suitable events.

[0200] At block 1206, the method 1200 includes determining playback information for the first playback session in the first domain. For instance, in response to detecting the first event, the MPS control app 761 on the mobile device 704 may determine that the first playback device was playing particular audio content in the first playback session and that the first playback device was at a given playback position in the particular audio content. As noted above, if the first domain is a native domain, the MPS control app 761 may already have the playback information (e.g., from the servers or the playback device). Such playback information may be stored or maintained in one or more state variables in memory of the playback devices 102 and/or the mobile device 704.

[0201] At block 1208, the method 1200 includes initiating a second playback session in a second domain based on the first playback session in the first domain. For example, the mobile device 704 may send, via a network interface, instructions to initiate a second playback session in the second domain to play back the particular audio content on the second playback device. The second playback device may be one of the playback device(s) 102 of the MPS 100 or a third-party playback device.

[0202] When the first domain is a native domain, to transfer out of the native domain, the MPS control app 761 may send the instructions according to an API conforming to the second domain. In particular, the MPS control app 761 on the mobile device 704 may send, via a network interface to the second playback device, instructions according to an application programming interface (API) of the second domain, the instructions causing the second playback device to initiate the playback session in the second domain to play back the particular audio content on the second playback device. As another example, the MPS control app 761 on the mobile device 704 may send, via a network interface to one or more cloud servers associated with the second domain, instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device. The cloud servers may be servers of a streaming audio service (e.g., the remote computing device(s) 106b), among other examples.

[0203] Other example techniques include sending, via an inter-process communication interface to an application associated with the second domain, instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device.

Conclusion

[0204] The description above discloses, among other things, various example systems, methods, apparatus, and

articles of manufacture including, among other components, firmware and/or software executed on hardware. It is understood that such examples are merely illustrative and should not be considered as limiting. For example, it is contemplated that any or all of the firmware, hardware, and/or software aspects or components can be embodied exclusively in hardware, exclusively in software, exclusively in firmware, or in any combination of hardware, software, and/or firmware. Accordingly, the examples provided are not the only way(s) to implement such systems, methods, apparatus, and/or articles of manufacture.

[0205] The specification is presented largely in terms of illustrative environments, systems, procedures, steps, logic blocks, processing, and other symbolic representations that directly or indirectly resemble the operations of data processing devices coupled to networks. These process descriptions and representations are typically used by those skilled in the art to most effectively convey the substance of their work to others skilled in the art. Numerous specific details are set forth to provide a thorough understanding of the present disclosure. However, it is understood to those skilled in the art that certain embodiments of the present disclosure can be practiced without certain, specific details. In other instances, well known methods, procedures, components, and circuitry have not been described in detail to avoid unnecessarily obscuring aspects of the embodiments. Accordingly, the scope of the present disclosure is defined by the appended claims rather than the foregoing description of embodiments.

[0206] When any of the appended claims are read to cover a purely software and/or firmware implementation, at least one of the elements in at least one example is hereby expressly defined to include a tangible, non-transitory medium such as a memory, DVD, CD, Blu-ray, and so on, storing the software and/or firmware.

[0207] The present technology is illustrated, for example, according to various aspects described below. Various examples of aspects of the present technology are described as numbered examples (1, 2, 3, etc.) for convenience. These are provided as examples and do not limit the present technology. It is noted that any of the dependent examples may be combined in any combination, and placed into a respective independent example. The other examples can be presented in a similar manner.

[0208] Example 1: A method comprising: while a first playback device is playing back audio content during a first playback session in a first domain, detecting a first event, the first event configured as a trigger to transfer playback to a second domain on a second playback device; in response to detecting the first event, determining that the first playback device was playing particular audio content from a network location at one or more servers of a particular streaming audio service in the first playback session and that the first playback device was at a given playback position in the particular audio content; and sending, via a network interface, instructions to initiate a second playback session in the second domain to play back the particular audio content on the second playback device, the instructions indicating the network location at the one or more servers of the particular streaming audio service and an offset in the particular audio content representing the given playback position.

[0209] Example 2: The method of Example 1, wherein determining that the first playback device was playing particular audio content from one or more servers of the

particular streaming audio service comprises: querying multiple streaming audio services for playback history data associated with respective user accounts of the streaming audio services, wherein the user accounts of the streaming audio services are registered with a media playback system comprising the first playback device; and determining, based on the queried playback history data, that the particular audio content was being played back when the first event was detected, wherein the particular audio content is associated with a particular identifier identifying the network location at the one or more servers of the particular streaming audio service, wherein sending the instructions indicating the network location at the one or more servers of the particular streaming audio service comprises sending instructions indicating the particular identifier.

[0210] Example 3: The method of any of Examples 1 and 2, wherein the first domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), wherein the mobile device is configured as the sender to configure the first playback device as the receiver to stream the particular audio content from the network location at the one or more servers during the playback session in the first domain, and wherein the second domain comprises a native control protocol, and wherein sending the instructions to initiate the playback session in the second domain comprises a native control application executing on the mobile device sending instructions according to the native control protocol to (i) populate a queue associated with the second playback device with the particular audio content and (ii) start playback at the offset.

[0211] Example 4: The method of any of Examples 1-3, wherein the first domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), wherein the mobile device is configured as the sender to configure the first playback device as the receiver to stream the particular audio content from the network location at the one or more servers during the playback session in the first domain, and wherein the second domain comprises a native control protocol, and wherein sending the instructions to initiate the playback session in the second domain comprises: a native control application executing on the mobile device initiating the playback session in the second domain; and the native control application executing on the mobile device sending, via an 802.15-compatible network interface, a data stream representing the particular audio content, the data stream beginning at the offset in the particular audio content, wherein receiving the data stream causes the second playback device to begin playback of the particular audio content at the offset.

[0212] Example 5: The method of Example 4, wherein detecting the first event comprises detecting that the mobile device has left the proximity of the first playback device, wherein the second playback device comprises a car stereo receiver, and wherein the method further comprises: after detecting the first event, detecting a second event, the second event corresponding to a trigger to initiate the playback session in the second domain, wherein sending the instructions to initiate the playback session in the second domain comprises sending the instructions to initiate the playback session in the second domain in response to detecting the second event, and wherein detecting the second event comprises detecting that the mobile device has formed a 802.15 connection with the car stereo receiver.

[0213] Example 6: The method of Example 4, wherein detecting the first event comprises detecting that the mobile device has left the proximity of the first playback device, wherein the second playback device comprises 802.15-compatible headphones, and wherein the method further comprises: after detecting the first event, detecting a second event, the second event corresponding to a trigger to initiate the playback session in the second domain, wherein sending the instructions to initiate the playback session in the second domain comprises sending the instructions to initiate the playback session in the second domain in response to detecting the second event, and wherein detecting the second event comprises detecting that the mobile device has formed a 802.15 connection with the 802.15-compatible headphones.

[0214] Example 7: The method of any of Examples 1-6, wherein the first domain comprises a streaming audio service control protocol and a streaming audio service application, and wherein the streaming audio service application sent instructions according to the streaming audio service control protocol to initiate the first playback session in the first domain on the first playback device, and wherein sending the instructions to initiate the playback session in the second domain comprises a native control application executing on the mobile device sending instructions according to the native control protocol to (i) populate a queue associated with the second playback device with the particular audio content and (ii) start playback at the offset.

[0215] Example 8: The method of any of Examples 1-7, wherein the first domain comprises a wireless streaming protocol defining a sender device and a receiver device connected via a local area network (LAN), wherein the mobile device is configured as the sender to configure the first playback device as the receiver to stream the particular audio content from the network location at the one or more servers during the playback session in the first domain, and wherein the second domain comprises a streaming audio service control protocol and a streaming audio service application, and wherein sending the instructions to initiate the playback session in the second domain comprises the streaming audio service control application executing on the mobile device sending instructions according to the streaming audio service control protocol to the second playback device to play back the particular audio content starting at the offset.

[0216] Example 9: The method of any of Examples 1-8, wherein the first domain comprises a streaming audio service control protocol and a streaming audio service application associated with a particular user account of the streaming media service, and wherein the streaming audio service application sent instructions according to the streaming audio service control protocol to initiate the first playback session in the first domain on the first playback device, and wherein sending the instructions to initiate the playback session in the second domain comprises a native control application associated with a particular user account of a native cloud service executing on the mobile device sending, to one or more servers of the native cloud service, instructions according to the native control protocol (i) populate a cloud queue associated with the second playback device with the particular audio content and (ii) start playback on the second playback device at the offset.

[0217] Example 10: The method of any of Examples 1-9, wherein detecting the first event comprises detecting a proximity tap between the mobile device and the second playback device.

[0218] Example 11: The method of any of Examples 1-10, wherein sending the instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device comprises: sending, via a network interface to one or more cloud servers associated with the second domain, instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device.

[0219] Example 12: The method of any of Examples 1-11, wherein sending the instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device comprises: sending, via an inter-process communication interface to an application associated with the second domain, instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device.

[0220] Example 13: The method of any of Examples 1-12, wherein sending the instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device comprises: sending, via an inter-process communication interface to an application associated with the second domain, instructions to initiate the playback session in the second domain to play back the particular audio content on the second playback device.

[0221] Example 14: A tangible, non-transitory, computer-readable medium having instructions stored thereon that are executable by one or more processors to cause a mobile device to perform the method of any one of Examples 1-13.

[0222] Example 15: A mobile device comprising a network interface, one or more processors, and a data storage having instructions stored thereon executable by the one or more processors to cause the playback device to perform the method of any of Examples 1-13.

[0223] Example 16: A system configured to perform the method of any of Examples 1-13.

1. A method to be performed by a mobile device, the method comprising:

while a playback device is playing back audio content during a first playback session in a first domain, detecting an event, the event configured as a trigger to transfer playback to a second domain on the mobile device;

in response to detecting the event, determining that the playback device was playing particular audio content from a network location at one or more servers of a particular streaming audio service in the first playback session and that the playback device was at a given playback position in the particular audio content; and initiating a second playback session in the second domain to play back the particular audio content on the mobile device from the network location at the one or more servers of the particular streaming audio service and beginning at an offset in the particular audio content representing the given playback position.

2. The method of claim 1, wherein determining that the playback device was playing particular audio content from one or more servers of the particular streaming audio service comprises:

querying multiple streaming audio services for playback history data associated with respective user accounts of the streaming audio services, wherein the user accounts of the streaming audio services are registered with a media playback system comprising the playback device; and

determining, based on the queried playback history data, that the particular audio content was being played back when the event was detected, wherein the particular audio content is associated with a particular identifier identifying the network location at the one or more servers of the particular streaming audio service, wherein initiating the second playback session in the second domain comprises sending instructions indicating the network location at the one or more servers of the particular streaming audio service via the particular identifier.

3. The method of claim 1, wherein the first domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), wherein the mobile device is configured as the sender to configure the playback device as the receiver to stream the particular audio content from the network location at the one or more servers during the first playback session in the first domain, and wherein the second domain comprises a streaming audio service control protocol, and wherein initiating the second playback session in the second domain comprises a streaming audio service application executing on the mobile device sending instructions according to the streaming audio service control protocol to (i) stream data representing the particular audio content and (ii) start playback at the offset.

4. The method of claim 1, wherein the first domain comprises a streaming audio service control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), and wherein initiating the second playback session in the second domain comprises defining the mobile device as at least one of (a) the sender and (b) the receiver of the second playback session.

5. The method of claim 1, wherein the first domain comprises a native control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), wherein the second domain comprises a streaming audio service control protocol, and wherein initiating the second playback session in the second domain comprises a streaming audio service application executing on the mobile device sending instructions according to the streaming audio service control protocol to (i) stream data representing the particular audio content and (ii) start playback at the offset.

6. The method of claim 1, wherein the first domain comprises a native control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), and wherein initiating the second playback session in the second domain comprises defining the mobile device as at least one of (a) the sender and (b) the receiver of the second playback session.

7. The method of claim 1, wherein detecting the event comprises detecting that the mobile device has left proximity of the playback device, wherein the playback device comprises a car stereo receiver, and wherein the method further comprises:

after detecting the event, detecting an additional event, the additional event corresponding to a trigger to initiate a third playback session in a third domain;

determining that the playback device was playing the particular audio content from the network location at the one or more servers of the particular streaming audio service in the first playback session and that the playback device was at a subsequent playback position in the particular audio content; and

after detecting the additional event, initiating the third playback session in the third domain to play back the particular audio content on an additional device from the network location at the one or more servers of the particular streaming audio service and beginning at an additional offset in the particular audio content representing the subsequent playback position.

8. The method of claim 1, wherein detecting the event comprises detecting a proximity tap between the mobile device and the playback device.

9. A tangible, non-transitory computer-readable medium having stored thereon instructions executable by at least one processor to cause a mobile device to perform functions comprising:

while a playback device is playing back audio content during a first playback session in a first domain, detecting an event, the event configured as a trigger to transfer playback to a second domain on the mobile device;

in response to detecting the event, determining that the playback device was playing particular audio content from a network location at one or more servers of a particular streaming audio service in the first playback session and that the playback device was at a given playback position in the particular audio content; and initiating a second playback session in the second domain to play back the particular audio content on the mobile device from the network location at the one or more servers of the particular streaming audio service and beginning at an offset in the particular audio content representing the given playback position.

10. The tangible, non-transitory computer-readable medium of claim 9, wherein determining that the playback device was playing particular audio content from one or more servers of the particular streaming audio service comprises:

querying multiple streaming audio services for playback history data associated with respective user accounts of the streaming audio services, wherein the user accounts of the streaming audio services are registered with a media playback system comprising the playback device; and

determining, based on the queried playback history data, that the particular audio content was being played back when the event was detected, wherein the particular audio content is associated with a particular identifier identifying the network location at the one or more servers of the particular streaming audio service, wherein sending the instructions indicating the network location at the one or more servers of the particular streaming audio service comprises sending instructions indicating the particular identifier.

11. The tangible, non-transitory computer-readable medium of claim 9, wherein the first domain comprises a wireless streaming protocol defining a sender and a receiver

connected via a local area network (LAN), wherein the mobile device is configured as the sender to configure the playback device as the receiver to stream the particular audio content from the network location at the one or more servers during the first playback session in the first domain, and wherein the second domain comprises a streaming audio service control protocol, and wherein initiating the second playback session in the second domain comprises a streaming audio service application executing on the mobile device sending instructions according to the streaming audio service control protocol to (i) stream data representing the particular audio content and (ii) start playback at the offset.

12. The tangible, non-transitory computer-readable medium of claim 9, wherein the first domain comprises a streaming audio service control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), and wherein initiating the second playback session in the second domain comprises defining the mobile device as at least one of (a) the sender and (b) the receiver of the second playback session.

13. The tangible, non-transitory computer-readable medium of claim 9, wherein the first domain comprises a native control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), wherein the second domain comprises a streaming audio service control protocol, and wherein initiating the second playback session in the second domain comprises a streaming audio service application executing on the mobile device sending instructions according to the streaming audio service control protocol to (i) stream data representing the particular audio content and (ii) start playback at the offset.

14. The tangible, non-transitory computer-readable medium of claim 9, wherein the first domain comprises a native control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), and wherein initiating the second playback session in the second domain comprises defining the mobile device as at least one of (a) the sender and (b) the receiver of the second playback session.

15. The tangible, non-transitory computer-readable medium of claim 9, wherein detecting the event comprises detecting that the mobile device has left proximity of the playback device, wherein the playback device comprises a car stereo receiver, and wherein the functions further comprises:

after detecting the event, detecting an additional event, the additional event corresponding to a trigger to initiate a third playback session in a third domain;

determining that the playback device was playing the particular audio content from the network location at the one or more servers of the particular streaming audio service in the first playback session and that the playback device was at a subsequent playback position in the particular audio content; and

after detecting the additional event, initiating the third playback session in the third domain to play back the particular audio content on an additional device from the network location at the one or more servers of the particular streaming audio service and beginning at an additional offset in the particular audio content representing the subsequent playback position.

16. The tangible, non-transitory computer-readable medium of claim 9, wherein detecting the event comprises detecting a proximity tap between the mobile device and the playback device.

17. A system comprising a mobile device and a playback device, the system configured to perform functions comprising:

while the playback device is playing back audio content during a first playback session in a first domain, detecting an event, the event configured as a trigger to transfer playback to a second domain on the mobile device;

in response to detecting the event, determining that the playback device was playing particular audio content from a network location at one or more servers of a particular streaming audio service in the first playback session and that the playback device was at a given playback position in the particular audio content; and initiating a second playback session in the second domain to play back the particular audio content on the mobile device from the network location at the one or more servers of the particular streaming audio service and beginning at an offset in the particular audio content representing the given playback position.

18. The system of claim 17, wherein the first domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), wherein the mobile device is configured as the sender to configure the playback device as the receiver to stream the particular audio content from the network location at the one

or more servers during the first playback session in the first domain, and wherein the second domain comprises a streaming audio service control protocol, and wherein initiating the second playback session in the second domain comprises a streaming audio service application executing on the mobile device sending instructions according to the streaming audio service control protocol to (i) stream data representing the particular audio content and (ii) start playback at the offset.

19. The system of claim 17, wherein the first domain comprises a native control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), wherein the second domain comprises a streaming audio service control protocol, and wherein initiating the second playback session in the second domain comprises a streaming audio service application executing on the mobile device sending instructions according to the streaming audio service control protocol to (i) stream data representing the particular audio content and (ii) start playback at the offset.

20. The system of claim 17, wherein the first domain comprises a native control protocol, and wherein the second domain comprises a wireless streaming protocol defining a sender and a receiver connected via a local area network (LAN), and wherein initiating the second playback session in the second domain comprises defining the mobile device as at least one of (a) the sender and (b) the receiver of the second playback session.

* * * * *

Exhibit 14



US 20220224971A1

(19) **United States**

(12) **Patent Application Publication**
Coburn, IV et al.

(10) **Pub. No.: US 2022/0224971 A1**

(43) **Pub. Date: Jul. 14, 2022**

(54) **SYSTEMS AND METHODS FOR
NETWORKED MUSIC PLAYBACK**

(71) Applicant: **Sonos, Inc.**, Santa Barbara, CA (US)

(72) Inventors: **Arthur Coburn, IV**, Lexington, MA
(US); **Joni Rafalski Hoadley**, Santa
Barbara, CA (US)

(21) Appl. No.: **17/710,425**

(22) Filed: **Mar. 31, 2022**

Related U.S. Application Data

(63) Continuation of application No. 17/145,268, filed on Jan. 8, 2021, which is a continuation of application No. 17/019,174, filed on Sep. 11, 2020, now Pat. No. 10,945,027, which is a continuation of application No. 16/389,906, filed on Apr. 19, 2019, now Pat. No. 10,779,033, which is a continuation of application No. 15/872,500, filed on Jan. 16, 2018, now Pat. No. 10,567,831, which is a continuation of application No. 14/520,578, filed on Oct. 22, 2014, now Pat. No. 9,883,234, which is a continuation of application No. 13/341,237, filed on Dec. 30, 2011, now Pat. No. 9,654,821.

Publication Classification

(51) **Int. Cl.**

H04N 21/436 (2006.01)
G06F 3/0481 (2006.01)
G06F 3/04842 (2006.01)
G06F 3/04847 (2006.01)
G06F 3/16 (2006.01)
G11B 19/02 (2006.01)
H04L 65/1101 (2006.01)
H04L 65/612 (2006.01)
H04N 21/43 (2006.01)
H04N 21/433 (2006.01)
H04N 21/439 (2006.01)
H04N 21/472 (2006.01)
H04N 21/485 (2006.01)

H04N 21/61 (2006.01)
H04N 21/643 (2006.01)
H04N 21/658 (2006.01)
H04N 21/6587 (2006.01)
H04N 21/81 (2006.01)
H04N 21/858 (2006.01)
H04R 3/12 (2006.01)

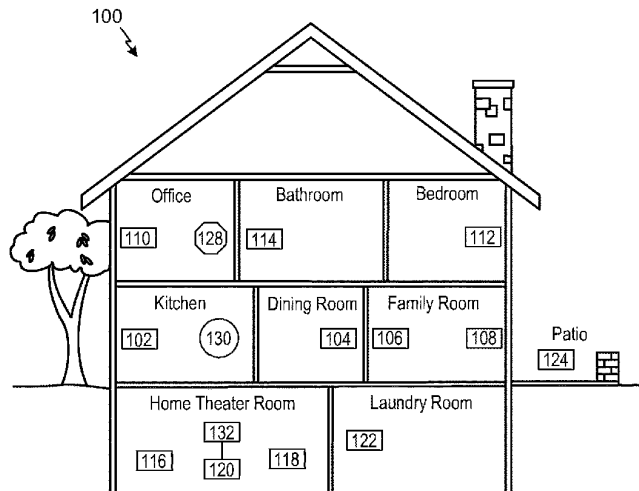
(52) **U.S. Cl.**

CPC ... **H04N 21/43615** (2013.01); **G06F 3/04817**
(2013.01); **G06F 3/04842** (2013.01); **G06F**
3/04847 (2013.01); **G06F 3/165** (2013.01);
G11B 19/025 (2013.01); **H04L 29/06027**
(2013.01); **H04L 65/4084** (2013.01); **H04N**
21/4307 (2013.01); **H04N 21/4333** (2013.01);
H04N 21/439 (2013.01); **H04N 21/47202**
(2013.01); **H04N 21/4852** (2013.01); **H04N**
21/6125 (2013.01); **H04N 21/64322** (2013.01);
H04N 21/6581 (2013.01); **H04N 21/6587**
(2013.01); **H04N 21/8113** (2013.01); **H04N**
21/8586 (2013.01); **H04R 3/12** (2013.01);
H04R 2420/07 (2013.01); **G06F 3/0481**
(2013.01)

(57)

ABSTRACT

A network device is configured to perform functions including (i) operating in a first playback-control mode including retrieving, via a first network interface, first media in a playback queue and streaming, via a second network interface to a headphone, the first media (ii) while streaming the first media to the headphone, displaying a representation of one or more playback devices in a media playback system, (iii) receiving input indicating a selection of a given playback device, (iv) based on the input, causing the given playback device to obtain data indicating the second media in the playback queue, use the obtained data to retrieve the second media, and play back the retrieved second media, and (v) after receiving the input, stopping streaming the retrieved first media and transitioning from operating in the first playback-control mode to operating in a second playback-control mode to control the given playback device's playback of the second media.



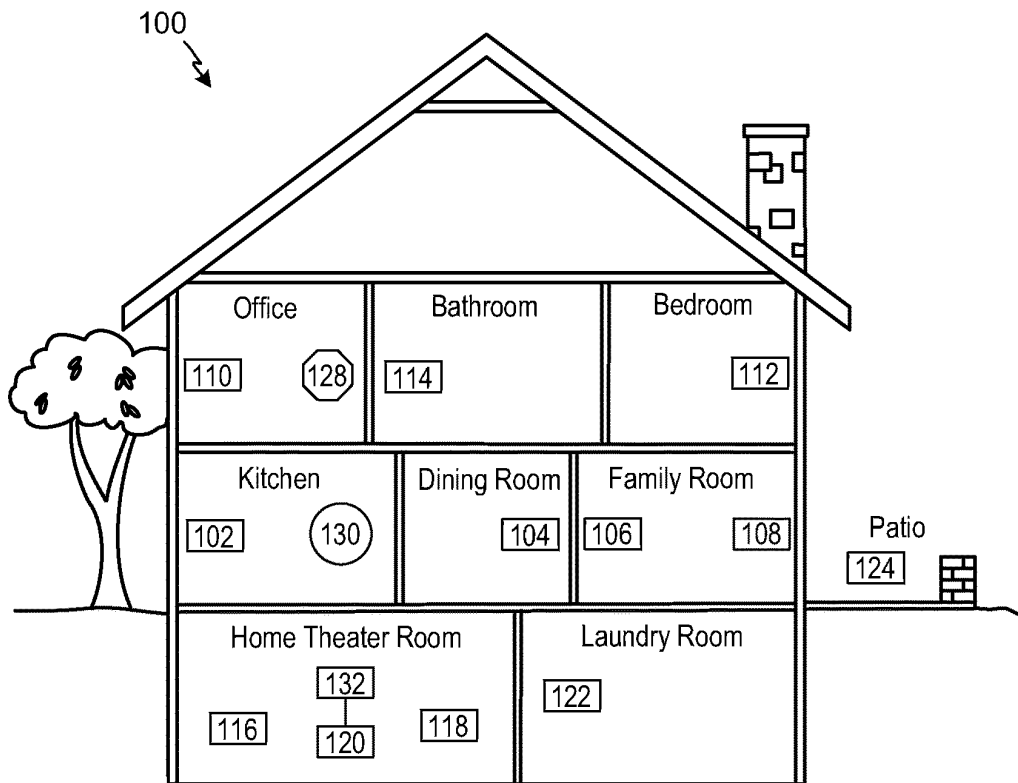


FIGURE 1

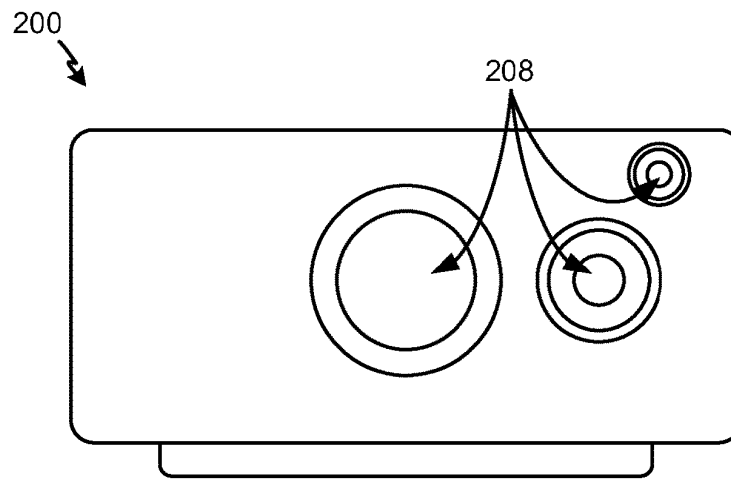


FIGURE 2A

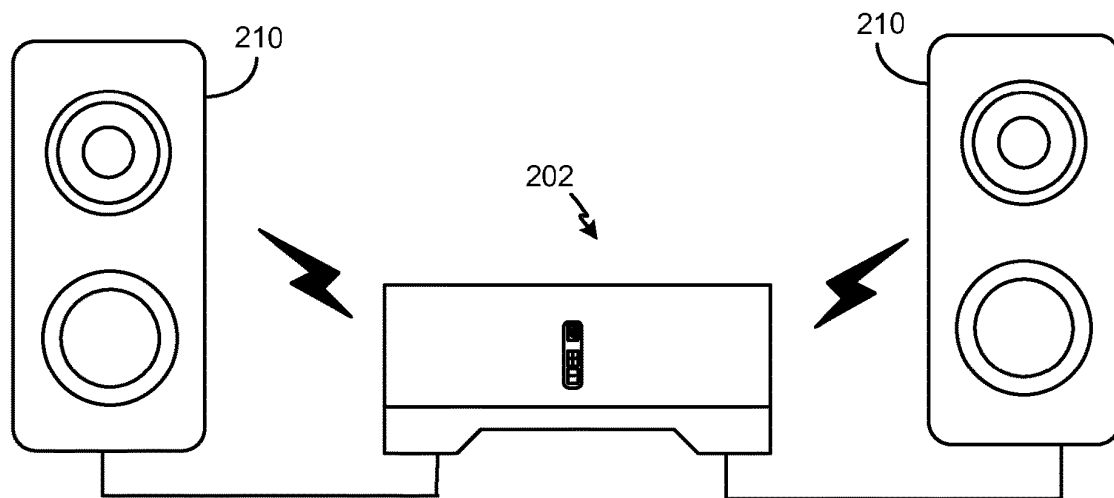


FIGURE 2B

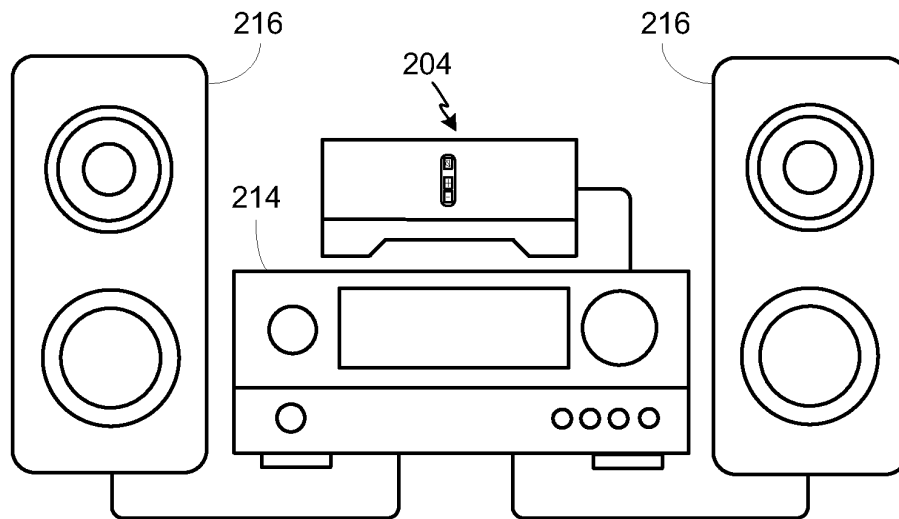


FIGURE 2C

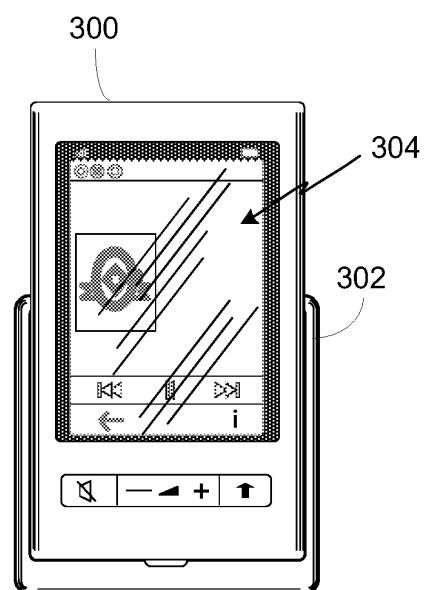


FIGURE 3

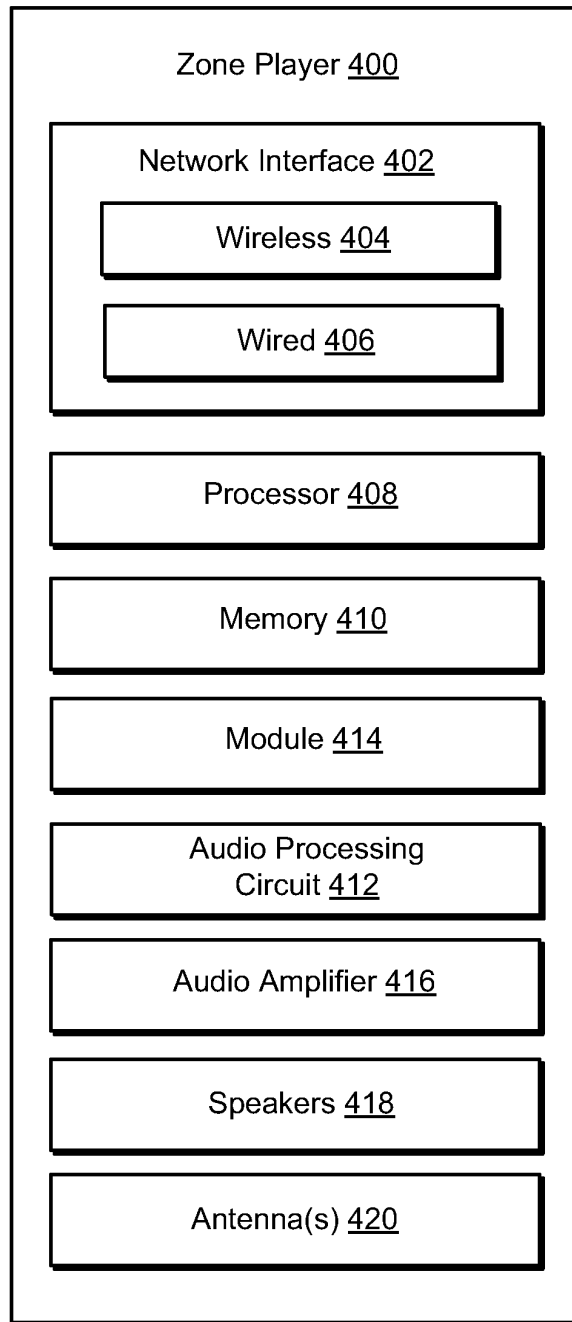


FIGURE 4

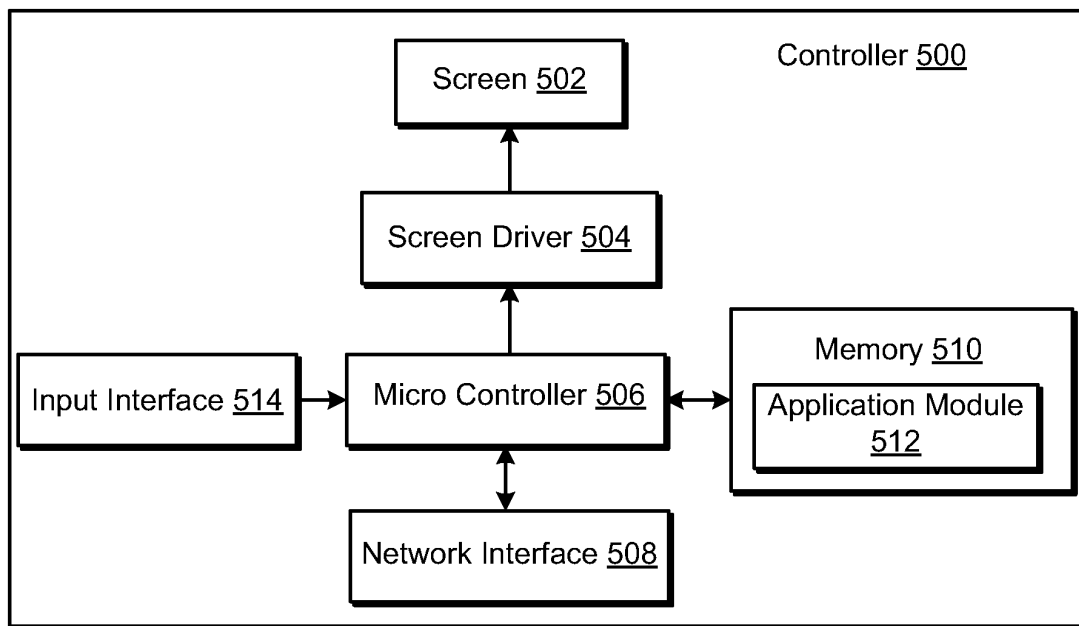


FIGURE 5

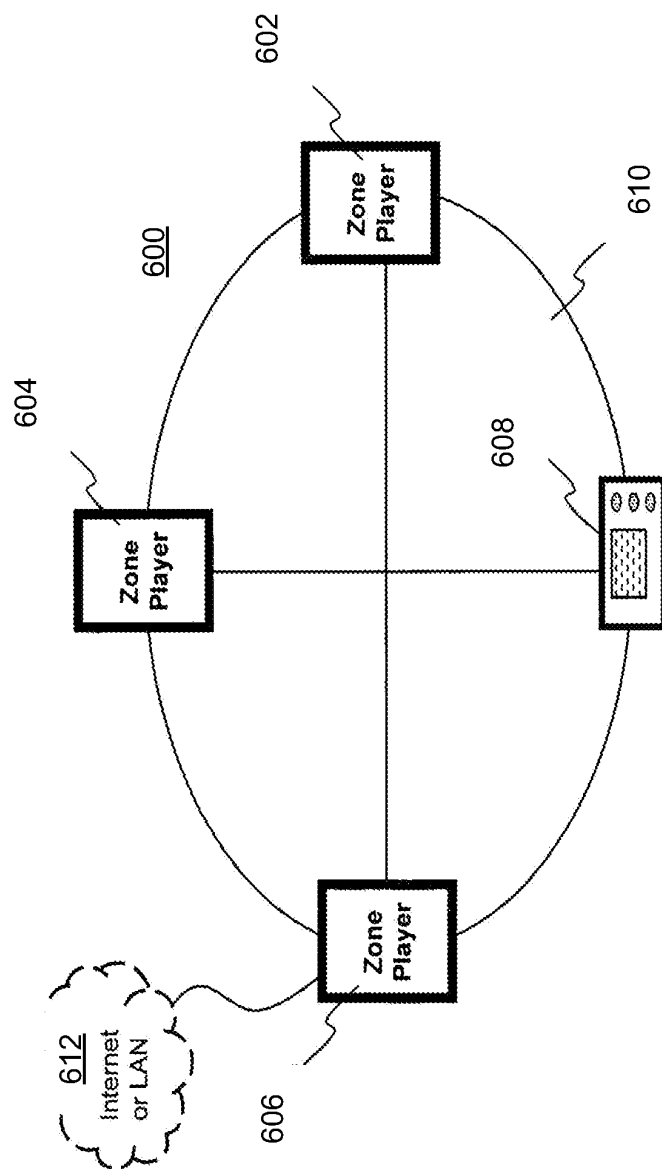


FIGURE 6

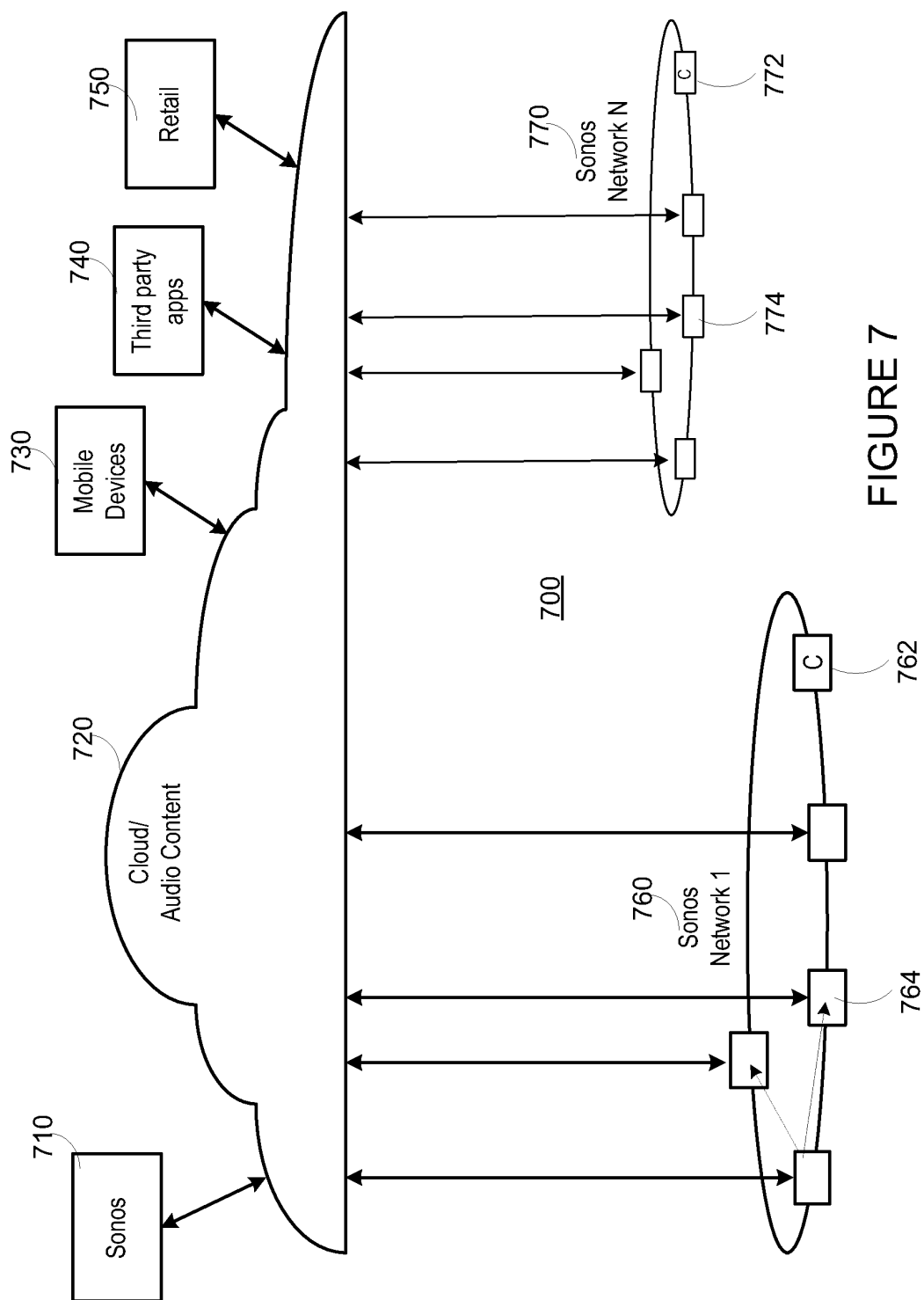


FIGURE 7

800 →

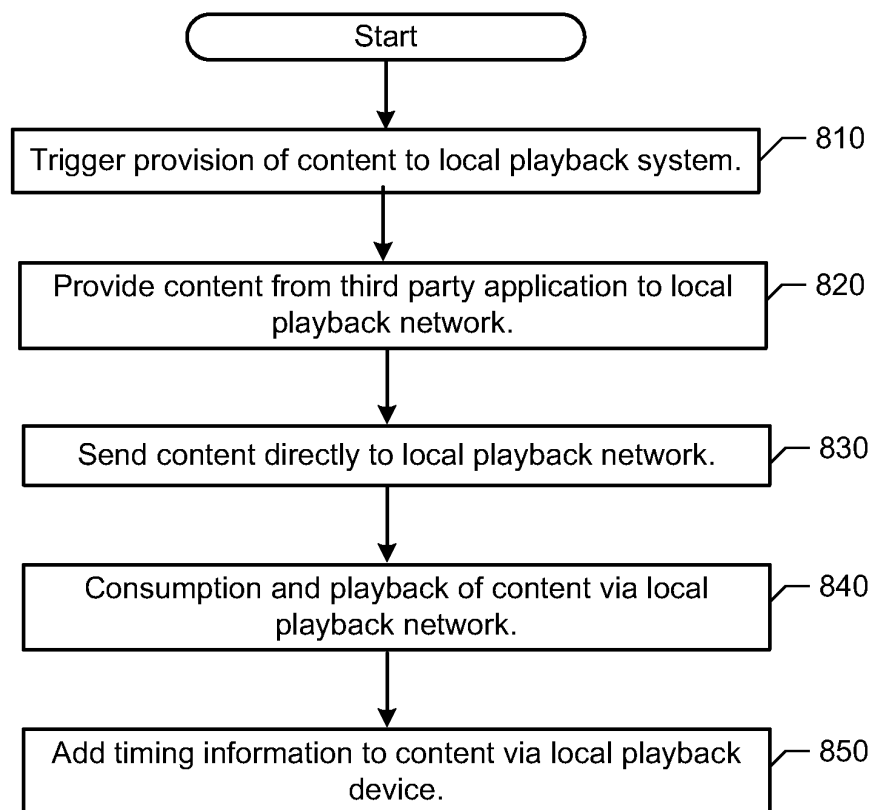


FIGURE 8

900 →

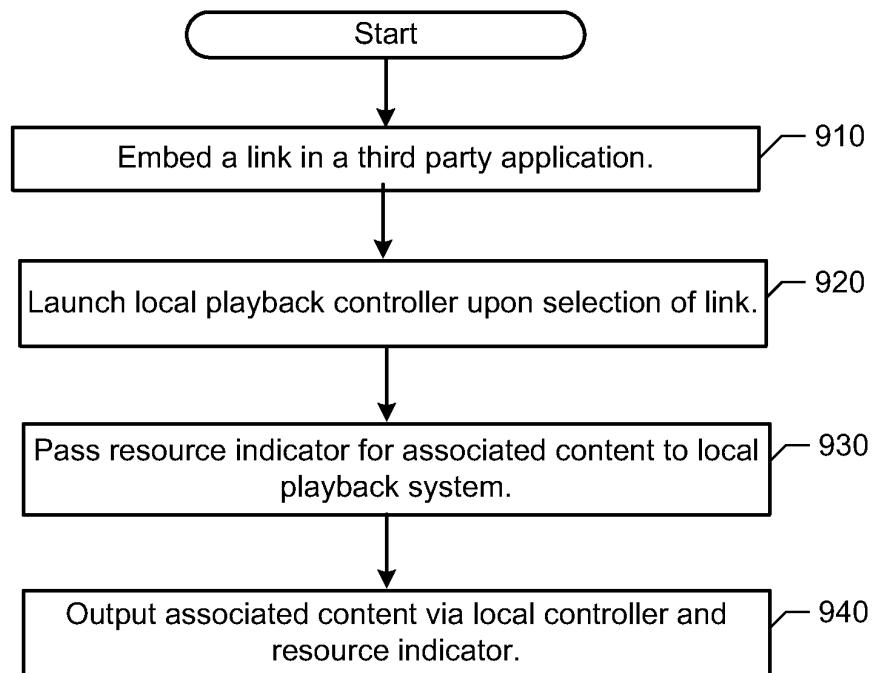


FIGURE 9

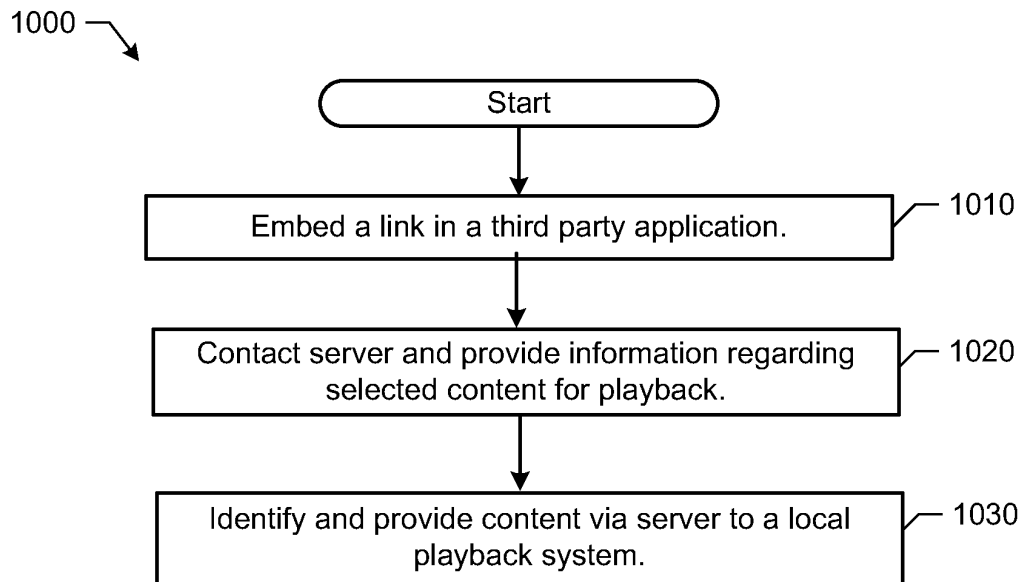


FIGURE 10

1100 →

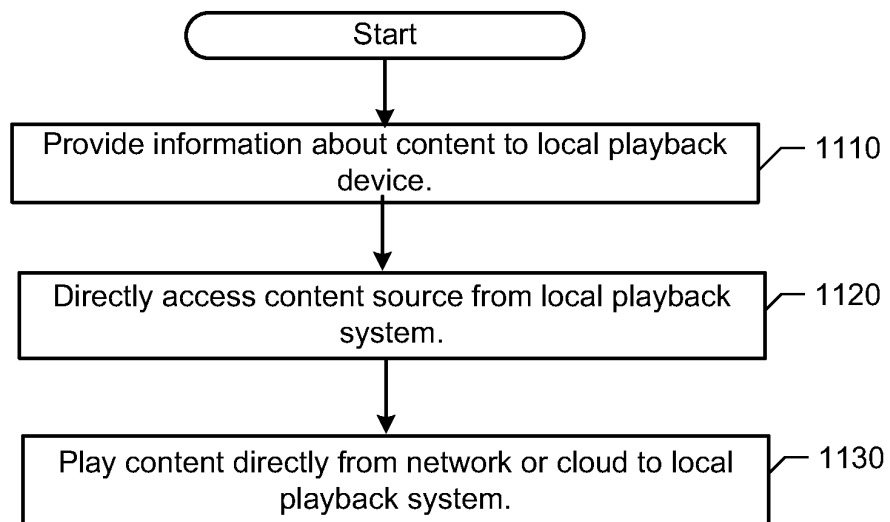


FIGURE 11

SYSTEMS AND METHODS FOR NETWORKED MUSIC PLAYBACK

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of U.S. patent application Ser. No. 17/145,268, filed on Jan. 8, 2021, which is a continuation of U.S. patent application Ser. No. 17/019,174, filed on Sep. 11, 2020 and issued on Mar. 9, 2021 as U.S. Pat. No. 10,945,027, which is a continuation of U.S. patent application Ser. No. 16/389,906, filed on Apr. 19, 2019 and issued on Sep. 15, 2020 as U.S. Pat. No. 10,779,033, which is a continuation of U.S. patent application Ser. No. 15/872,500, filed on Jan. 16, 2018 and issued on Feb. 18, 2020 as U.S. Pat. No. 10,567,831, which is a continuation of U.S. patent application Ser. No. 14/520,578, filed on Oct. 22, 2014 and issued on Jan. 30, 2018 as U.S. Pat. No. 9,883,234, which is a continuation of U.S. patent application Ser. No. 13/341,237, filed on Dec. 30, 2011 and issued on May 15, 2017 as U.S. Pat. No. 9,654,821, all of which are incorporated herein by reference in their entirety.

FIELD OF THE DISCLOSURE

[0002] The disclosure is related to consumer electronics and, more particularly, to providing music for playback via one or more devices on a playback data network.

BACKGROUND

[0003] Technological advancements have increased the accessibility of music content, as well as other types of media, such as television content, movies, and interactive content. For example, a user can access audio, video, or both audio and video content over the Internet through an online store, an Internet radio station, an online music service, an online movie service, and the like, in addition to the more traditional avenues of accessing audio and video content. Demand for such audio and video content continues to surge. Given the high demand, technology used to access and play such content has likewise improved.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] Features, aspects, and advantages of the presently disclosed technology are better understood with regard to the following description, appended claims, and accompanying drawings where:

[0005] FIG. 1 shows an illustration of an example system in which embodiments of the methods and apparatus disclosed herein can be implemented;

[0006] FIG. 2A shows an illustration of an example zone player having a built-in amplifier and speakers;

[0007] FIG. 2B shows an illustration of an example zone player having a built-in amplifier and connected to external speakers;

[0008] FIG. 2C shows an illustration of an example zone player connected to an A/V receiver and speakers;

[0009] FIG. 3 shows an illustration of an example controller;

[0010] FIG. 4 shows an internal functional block diagram of an example zone player;

[0011] FIG. 5 shows an internal functional block diagram of an example controller;

[0012] FIG. 6 shows an example ad-hoc playback network;

[0013] FIG. 7 shows a system including a plurality of networks including a cloud-based network and at least one local playback network; and

[0014] FIGS. 8-11 show flow diagrams for methods to provide audio content to a local playback system.

[0015] In addition, the drawings are for the purpose of illustrating example embodiments, but it is understood that the present disclosure is not limited to the arrangements and instrumentality shown in the drawings.

DETAILED DESCRIPTION

I. Overview

[0016] Wired or wireless networks can be used to connect one or more multimedia playback devices for a home or other location playback network (e.g., a home music system). Certain examples provide automatic configuration of parameters of a playback device to be coupled to a network with reduced or minimum human intervention. For example, a wired and/or wireless ad-hoc network is established to facilitate communications among a group of devices. Music and/or other multimedia content can be shared among devices and/or groups of devices (also referred to herein as zones) associated with a playback network.

[0017] Certain embodiments facilitate streaming or otherwise providing music from a music-playing application (e.g., browser-based application, native music player, other multimedia application, and so on) to a multimedia content playback (e.g., Sonos™) system. Certain embodiments provide simple, easy-to-use and secure systems and methods for multimedia content playback across a plurality of systems and locations. Certain embodiments facilitate integration between content partners and a playback system as well as supporting maintenance of such content and system.

[0018] Although the following discloses example systems, methods, apparatus, and articles of manufacture including, among other components, firmware and/or software executed on hardware, it should be noted that such systems, methods, apparatus, and/or articles of manufacture are merely illustrative and should not be considered as limiting. For example, it is contemplated that any or all of these firmware, hardware, and/or software components could be embodied exclusively in hardware, exclusively in software, exclusively in firmware, or in any combination of hardware, software, and/or firmware. Accordingly, while the following describes example systems, methods, apparatus, and/or articles of manufacture, the examples provided are not the only way(s) to implement such systems, methods, apparatus, and/or articles of manufacture.

[0019] When any of the appended claims are read to cover a purely software and/or firmware implementation, at least one of the elements in at least one example is hereby expressly defined to include a tangible medium such as a memory, DVD, CD, Blu-ray, and so on, storing the software and/or firmware.

[0020] Reference herein to “embodiment” means that a particular feature, structure, or characteristic described in connection with the embodiment can be included in at least one example embodiment of the invention. The appearances of this phrase in various places in the specification are not necessarily all referring to the same embodiment, nor are separate or alternative embodiments mutually exclusive of other embodiments. As such, the embodiments described

herein, explicitly and implicitly understood by one skilled in the art, can be combined with other embodiments.

[0021] Certain embodiments provide a method to provide content to a local playback network. The example method includes identifying multimedia content from a content provider. The example method includes passing information regarding the multimedia content to a local playback system including one or more multimedia playback devices in response to a trigger. The example method includes facilitating play of the multimedia content via a local playback network associated with the local playback system.

[0022] Certain embodiments provide a computer readable storage medium including instructions for execution by a processor, the instructions, when executed, cause the processor to implement a method to provide content to a local playback network. The example method includes identifying multimedia content from a content provider. The example method includes passing information regarding the multimedia content to a local playback system including one or more multimedia playback devices in response to a trigger. The example method includes facilitating play of the multimedia content via a local playback network associated with the local playback system.

[0023] Certain embodiments provide a multimedia playback device including a wireless communication interface to communicate with a local playback network and a multimedia content source and a processor. The process is to identify multimedia content from the multimedia content source; pass information regarding the multimedia content to device on the local playback network in response to a trigger; and facilitate play of the multimedia content via the devices on the local playback network.

II. Example Environment

[0024] Referring now to the drawings, in which like numerals can refer to like parts throughout the figures, FIG. 1 shows an example system configuration 100 in which one or more of the method and/or apparatus disclosed herein can be practiced or implemented. By way of illustration, the system configuration 100 represents a home with multiple zones. Each zone, for example, represents a different room or space, such as an office, bathroom, bedroom, kitchen, dining room, family room, home theater room, utility or laundry room, and patio. While not shown here, a single zone can cover more than one room or space. One or more of zone players 102-124 are shown in each respective zone. A zone player 102-124, also referred to as a playback device, multimedia unit, speaker, and so on, provides audio, video, and/or audiovisual output. A controller 130 (e.g., shown in the kitchen for purposes of illustration) provides control to the system configuration 100. The system configuration 100 illustrates an example whole house audio system, though it is understood that the technology described herein is not limited to its particular place of application or to an expansive system like a whole house audio system 100 of FIG. 1.

[0025] FIGS. 2A, 2B, and 2C show example illustrations of zone players 200-204. The zone players 200-204 of FIGS. 2A, 2B, and 2C, respectively, can correspond to any of the zone players 102-124 of FIG. 1. While certain embodiments provide multiple zone players, an audio output can be generated using only a single zone player. FIG. 2A illustrates a zone player 200 including sound producing equipment 208 capable of generating sound or an audio output corresponding to a signal received (e.g., wirelessly and/or via a wired

interface). The sound producing equipment 208 of the zone player 200 of FIG. 2A includes a built-in amplifier (not shown in this illustration) and speakers (e.g., a tweeter, a mid-range driver, and/or a subwoofer). In certain embodiments, the zone player 200 of FIG. 2A can be configured to play stereophonic audio or monaural audio. In some embodiments, the zone player 200 of FIG. 2A can be configured as a component in a combination of zone players to play stereophonic audio, monaural audio, and/or surround audio. As described in greater detail below, in some embodiments, the example zone player 200 of FIG. 2A can also transmit a second signal to, for example, other zone player(s) in the same or different zone(s), speaker(s), receiver(s), and so on. Transmission of the second signal can be part of, for example, a system in which multiple zone players, speakers, receivers, and so on, form a network to, for example, present media content in a synchronization or distributed manner.

[0026] The example zone player 202 of FIG. 2B includes a built-in amplifier (not shown in this illustration) to power a set of detached speakers 210. The speakers 210 of FIG. 2B can include, for example, any type of loudspeaker. The zone player 202 of FIG. 2B can communicate a signal corresponding to audio content to the detached speakers 210 via wired and/or wireless channels. Instead of receiving and generating audio content as in FIG. 2A, the zone player 202 of FIG. 2B receives the audio content and transmits the same (e.g., after processing the received signal) to the detached speakers 210. Similar to the example zone player 200 of FIG. 2A, in some embodiments the zone player 202 can transmit a second signal to, for example, other zone player(s) in the same or different zone(s), speaker(s), receiver(s), and so on.

[0027] The example zone player 204 of FIG. 2C does not include an amplifier, but allows a receiver 214, or another audio and/or video type device with built-in amplification, to connect to a data network 128 of FIG. 1 and to play audio received over the data network 128 via the receiver 214 and a set of detached speakers 216. In addition to the wired couplings shown in FIG. 2C, the detached speakers 216 can receive audio content via a wireless communication channel between the detached speakers 216 and, for example, the zone player 204 and/or the receiver 214. In some embodiments the zone player 202 can transmit a second signal to, for example, other zone player(s) in the same or different zone(s), speaker(s), receiver(s), and so on.

[0028] Example zone players include a “Sonos® S5,” “Sonos Play:5,” “Sonos Play:3,” “ZonePlayer 120,” and “ZonePlayer 90,” which are offered by Sonos, Inc. of Santa Barbara, Calif. Any other past, present, and/or future zone players can additionally or alternatively be used to implement the zone players of example embodiments disclosed herein. A zone player can also be referred to herein as a playback device, and a zone player is not limited to the particular examples illustrated in FIGS. 2A, 2B, and 2C. For example, a zone player can include a wired or wireless headphone. In other examples, a zone player might include a subwoofer. In yet other examples, a zone player can include a sound bar. In an example, a zone player can include or interact with a docking station for an Apple iPod™ or similar device. In some embodiments, a zone player can relay one or more signals received from, for example, a first zone player to another playback device. In some embodiments, a zone player can receive a first signal

and generate an output corresponding to the first signal and, simultaneously or separately, can receive a second signal and transmit or relay the second signal to another zone player(s), speaker(s), receiver(s), and so on. Thus, an example zone player described herein can act as a playback device and, at the same time, operate as a hub in a network of zone players. In such instances, media content corresponding to the first signal can be different from the media content corresponding to the second signal.

[0029] FIG. 3 shows an example illustration of a wireless controller **300** in a docking station **302**. The controller **300** can correspond to the controlling device **130** of FIG. 1. The controller **300** is provided with a touch screen **304** that allows a user to interact with the controller **300**, for example, to retrieve and navigate a playlist of audio items, control operations of one or more zone players, and provide overall control of the system configuration **100**. In certain embodiments, any number of controllers can be used to control the system configuration **100**. In certain embodiments, there can be a limit on the number of controllers that can control the system configuration **100**. The controllers might be wireless like wireless controller **300** or wired to the data network **128**. Furthermore, an application running on any network-enabled portable devices, such as an iPhone™, iPad™, Android™ powered phone, or any other smart phone or network-enabled device can be used as a controller by connecting to the data network **128**. An application running on a laptop or desktop PC or Mac can also be used as a controller. Example controllers include a “Sonos® Controller **200**,” “Sonos® Controller for iPhone,” “Sonos® Controller for iPad,” “Sonos® Controller for Android,” “Sonos® Controller for Mac or PC,” which are offered by Sonos, Inc. of Santa Barbara, Calif. The flexibility of such an application and its ability to be ported to a new type of portable device is advantageous.

[0030] Referring back to the system configuration **100** of FIG. 1, a particular zone can contain one or more zone players. For example, the family room of FIG. 1 contains two zone players **106** and **108**, while the kitchen is shown with one zone player **102**. Zones can be dynamically configured by positioning a zone player in a room or space and assigning via the controller **130** the zone player to a new or existing zone. As such, zones can be created, combined with another zone, removed, and given a specific name (e.g., “Kitchen”), if so programmed. The zone players **102** to **124** are coupled directly or indirectly to a data network, such as the data network **128** shown in FIG. 1. The data network **128** is represented by an octagon in the figure to stand out from other components shown in the figure. While the data network **128** is shown in a single location, it is understood that such a network can be distributed in and around the system configuration **100**.

[0031] Particularly, the data network **128** can be a wired network, a wireless network, or a combination of both. In some embodiments, one or more of the zone players **102-124** are wirelessly coupled to the data network **128** based on a proprietary mesh network. In some embodiments, one or more of the zone players **102-124** are wirelessly coupled to the data network **128** using a non-mesh topology. In some embodiments, one or more of the zone players **102-124** are coupled via a wire to the data network **128** using Ethernet or similar technology. In addition to the one or more zone

players **102-124** connecting to the data network **128**, the data network **128** can further allow access to a wide area network, such as the Internet.

[0032] In certain embodiments, the data network **128** can be created by connecting any of the zone players **102-124**, or some other connecting device, to a broadband router. Other zone players **102-124** can then be added wired or wirelessly to the data network **128**. For example, a zone player (e.g., any of zone players **102-124**) can be added to the system configuration **100** by simply pressing a button on the zone player itself, which enables a connection to be made to the data network **128**. The broadband router can be connected to an Internet Service Provider (ISP), for example. The broadband router can be used to form another data network within the system configuration **100**, which can be used in other applications (e.g., web surfing). The data network **128** can also be used in other applications, if so programmed. Further, in certain embodiments, the data network **128** is the same network used for other applications in the household.

[0033] In certain embodiments, each zone can play from the same audio source as another zone or each zone can play from a different audio source. For example, someone can be grilling on the patio and listening to jazz music via zone player **124**, while someone is preparing food in the kitchen and listening to classical music via zone player **102**. Further, someone can be in the office listening to the same jazz music via zone player **110** that is playing on the patio via zone player **124**. In some embodiments, the jazz music played via zone players **110** and **124** is played in synchrony. Synchronizing playback amongst zones allows for someone to pass through zones while seamlessly listening to the audio. Further, zones can be put into a “party mode” such that all associated zones will play audio in synchrony.

[0034] In certain embodiments, a zone contains two or more zone players. For example, the family room contains two zone players **106** and **108**, and the home theater room contains at least zone players **116**, **118**, and **120**. A zone can be configured to contain as many zone players as desired, and for example, the home theater room might contain additional zone players to play audio from a 5.1 channel or greater audio source (e.g., a movie encoded with 5.1 or greater audio channels). If a zone contains two or more zone players, such as the two zone players **106** and **108** in the family room, then the two zone players **106** and **108** can be configured to play the same audio source in synchrony, or the two zone players **106** and **108** can be paired to play two separate sounds in left and right channels, for example. In other words, the stereo effects of a sound can be reproduced or enhanced through the two zone players **106** and **108**, one for the left sound and the other for the right sound. In certain embodiments, paired zone players can play audio in synchrony with other zone players.

[0035] In certain embodiments, three or more zone players can be configured to play various channels of audio that is encoded with three channels or more sound. For example, the home theater room shows zone players **116**, **118**, and **120**. If the sound is encoded as 2.1 channel audio, then the zone player **116** can be configured to play left channel audio, the zone player **118** can be configured to play right channel audio, and the zone player **120** can be configured to play bass frequencies. Other configurations are possible and depend on the number of zone players and the type of audio. Further, a particular zone can be configured to play a 5.1

channel audio in one instance, such as when playing audio from a movie, and then dynamically switch to play stereo, such as when playing audio from a two channel source.

[0036] In certain embodiments, two or more zone players can be sonically consolidated to form a single, consolidated zone player. A consolidated zone player (though made up of multiple, separate devices) can be configured to process and reproduce sound differently than an unconsolidated zone player or zone players that are paired, because a consolidated zone player will have additional speaker drivers from which sound can be passed. The consolidated zone player can further be paired with a single zone player or yet another consolidated zone player. Each playback device of a consolidated playback device is preferably set in a consolidated mode.

[0037] According to some embodiments, one can continue to do any of: group, consolidate, and pair zone players, for example, until a desired configuration is complete. The actions of grouping, consolidation, and pairing are preferably performed through a control interface, such as using controller 130, and not by physically connecting and re-connecting speaker wire, for example, to individual, discrete speakers to create different configurations. As such, certain embodiments described herein provide a more flexible and dynamic platform through which sound reproduction can be offered to the end-user.

[0038] Sources of audio content to be played by zone players 102-124 are numerous. Music from a personal library stored on a computer or networked-attached storage (NAS) can be accessed via the data network 128 and played. Internet radio stations, shows, and podcasts can be accessed via the data network 128. Music services that let a user stream and download music and audio content can be accessed via the data network 128. Further, music can be obtained from traditional sources, such as a turntable or CD player, via a line-in connection to a zone player, for example. Audio content can also be accessed through AirPlay™ wireless technology by Apple, Inc., for example. Audio content received from one or more sources can be shared amongst the zone players 102 to 124 via the data network 128 and/or the controller 130. The above-disclosed sources of audio content are referred to herein as network-based audio information sources. However, network-based audio information sources are not limited thereto.

[0039] The example home theater zone players 116, 118, 120 are coupled to an audio information source such as a television 132. In some examples, the television 132 is used as a source of audio for the home theater zone players 116, 118, 120, while in other examples audio information from the television 132 can be shared with any of the zone players 102-124 in the audio system 100.

III. Example Playback Device

[0040] Referring now to FIG. 4, there is shown an example functional block diagram of a zone player 400 in accordance with an embodiment. The zone player 400 of FIG. 4 includes a network interface 402, a processor 408, a memory 410, an audio processing component 412, a module 414, an audio amplifier 416, and a speaker unit 418 coupled to the audio amplifier 416. FIG. 2A shows an example illustration of such a zone player. Other types of zone players can not include the speaker unit 418 (e.g., such as shown in FIG. 2B) or the audio amplifier 416 (e.g., such as shown in FIG. 2C). Further, it is contemplated that the zone

player 400 can be integrated into another component. For example, the zone player 400 could be constructed as part of a lamp for indoor or outdoor use.

[0041] Referring back to FIG. 4, the network interface 402 facilitates a data flow between zone players and other devices on a data network (e.g., the data network 128 of FIG. 1) and the zone player 400. In some embodiments, the network interface 402 can manage the assembling of an audio source or file into smaller packets that are to be transmitted over the data network or reassembles received packets into the original source or file. In some embodiments, the network interface 402 can further handle the address part of each packet so that it gets to the right destination or intercepts packets destined for the zone player 400. Accordingly, in certain embodiments, each of the packets includes an Internet Protocol (IP)-based source address as well as an IP-based destination address.

[0042] In some embodiments, the network interface 402 can include one or both of a wireless interface 404 and a wired interface 406. The wireless interface 404, also referred to as an RF interface, provides network interface functions for the zone player 400 to wirelessly communicate with other devices (e.g., other zone player(s), speaker(s), receiver(s), component(s) associated with the data network 128, and so on) in accordance with a communication protocol (e.g., any of the wireless standards IEEE 802.11a, 802.11b, 802.11g, 802.11n, or 802.15). To receive wireless signals and to provide the wireless signals to the wireless interface 404 and to transmit wireless signals, the zone player 400 of FIG. 4 includes one or more antennas 420. The wired interface 406 provides network interface functions for the zone player 400 to communicate over a wire with other devices in accordance with a communication protocol (e.g., IEEE 802.3). In some embodiments, a zone player includes both of the interfaces 404 and 406. In some embodiments, a zone player 400 includes only the wireless interface 404 or the wired interface 406.

[0043] In some embodiments, the processor 408 is a clock-driven electronic device that is configured to process input data according to instructions stored in memory 410. The memory 410 is data storage that can be loaded with one or more software modules 414, which can be executed by the processor 408 to achieve certain tasks. In the illustrated embodiment, the memory 410 is a tangible machine readable medium storing instructions that can be executed by the processor 408. In some embodiments, a task might be for the zone player 400 to retrieve audio data from another zone player or a device on a network. In some embodiments, a task might be for the zone player 400 to send audio data to another zone player or device on a network. In some embodiments, a task might be for the zone player 400 to synchronize playback of audio with one or more additional zone players. In some embodiments, a task might be to pair the zone player 400 with one or more zone players to create a multi-channel audio environment. Additional or alternative tasks can be achieved via the one or more software modules 414 and the processor 408.

[0044] The audio processing component 412 can include one or more digital-to-analog converters (DAC), an audio preprocessing component, an audio enhancement component or a digital signal processor, and so on. In certain embodiments, the audio that is retrieved via the network interface 402 is processed and/or intentionally altered by the audio processing component 412. Further, the audio pro-

cessing component **412** can produce analog audio signals. The processed analog audio signals are then provided to the audio amplifier **416** for play back through speakers **418**. In addition, the audio processing component **412** can include necessary circuitry to process analog or digital signals as inputs to play from zone player **400**, send to another zone player on a network, or both play and send to another zone player on the network. An example input includes a line-in connection (e.g., an auto-detecting 3.5 mm audio line-in connection).

[0045] The audio amplifier **416** is a device that amplifies audio signals to a level for driving one or more speakers **418**. The one or more speakers **418** can include an individual transducer (e.g., a “driver”) or a complete speaker system that includes an enclosure including one or more drivers. A particular driver can be a subwoofer (for low frequencies), a mid-range driver (middle frequencies), and a tweeter (high frequencies), for example. An enclosure can be sealed or ported, for example.

[0046] A zone player **400** can also be referred to herein as a playback device. An example playback device includes a Sonos® Play:5, which is manufactured by Sonos, Inc. of Santa Barbara, Calif. The Play:5 is an example zone player with a built-in amplifier and speakers. In particular, the Play:5 is a five-driver speaker system that includes two tweeters, two mid-range drivers, and one subwoofer. When playing audio content via the Play:5, the left audio data of a track is sent out of the left tweeter and left mid-range driver, the right audio data of a track is sent out of the right tweeter and the right mid-range driver, and mono bass is sent out of the subwoofer. Further, both mid-range drivers and both tweeters have the same equalization (or substantially the same equalization). That is, they are both sent the same frequencies, just from different channels of audio. Audio from Internet radio stations, online music and video services, downloaded music, analog audio inputs, television, DVD, and so on, can be played from a Sonos® Play:5. While the Play:5 is an example of a zone player with speakers, it is understood that a zone player with speakers is not limited to one with a certain number of speakers (e.g., five speakers as in the Play:5), but rather can contain one or more speakers. Further, a zone player can be part of another device, which might even serve a purpose different than audio (e.g., a lamp).

IV. Example Controller

[0047] Referring now to FIG. 5, there is shown an example controller **500**, which can correspond to the controlling device **130** in FIG. 1. The controller **500** can be used to facilitate the control of multi-media applications, automation and others in a system. In particular, the controller **500** is configured to facilitate a selection of a plurality of audio sources available on the network and enable control of one or more zone players (e.g., the zone players **102-124** in FIG. 1) through a wireless network interface **508**. According to one embodiment, the wireless communications is based on an industry standard (e.g., infrared, radio, wireless standards IEEE 802.11a, 802.11b 802.11g, 802.11n, or 802.15). Further, when a particular audio is being accessed via the controller **500** or being played via a zone player, a picture (e.g., album art) or any other data, associated with the audio source can be transmitted from a zone player or other electronic device to the controller **500** for display.

[0048] The controller **500** is provided with a screen **502** and an input interface **514** that allows a user to interact with the controller **500**, for example, to navigate a playlist of many multimedia items and to control operations of one or more zone players. The screen **502** on the controller **500** can be an LCD screen, for example. The screen **500** communicates with and is commanded by a screen driver **504** that is controlled by a microcontroller (e.g., a processor) **506**. The memory **510** can be loaded with one or more application modules **512** that can be executed by the microcontroller **506** with or without a user input via the user interface **514** to achieve certain tasks. In some embodiments, an application module **512** is configured to facilitate grouping a number of selected zone players into a zone group and synchronizing the zone players for audio play back. In some embodiments, an application module **512** is configured to control the audio sounds (e.g., volume) of the zone players in a zone group. In operation, when the microcontroller **506** executes one or more of the application modules **512**, the screen driver **504** generates control signals to drive the screen **502** to display an application specific user interface accordingly.

[0049] The controller **500** includes a network interface **508** that facilitates wireless communication with a zone player. In some embodiments, the commands such as volume control and audio playback synchronization are sent via the network interface **508**. In some embodiments, a saved zone group configuration is transmitted between a zone player and a controller via the network interface **508**. The controller **500** can control one or more zone players, such as **102-124** of FIG. 1. There can be more than one controller for a particular system. Further, a controller can be integrated into a zone player.

[0050] It should be noted that other network-enabled devices such as an iPhone®, iPad® or any other smart phone or network-enabled device (e.g., a networked computer such as a PC or Mac®) can also be used as a controller to interact or control zone players in a particular environment. In some embodiments, a software application or upgrade can be downloaded onto a network enabled device to perform the functions described herein.

[0051] In certain embodiments, a user can create a zone group including at least two zone players from the controller **500**. The zone players in the zone group can play audio in a synchronized fashion, such that all of the zone players in the zone group play back an identical audio source or a list of identical audio sources in a synchronized manner such that no (or substantially no) audible delays or hiccups could be heard. Similarly, in some embodiments, when a user increases the audio volume of the group from the controller **500**, the signals or data of increasing the audio volume for the group are sent to one of the zone players and causes other zone players in the group to be increased together in volume.

[0052] A user via the controller **500** can group zone players into a zone group by activating a “Link Zones” or “Add Zone” soft button, or de-grouping a zone group by activating an “Unlink Zones” or “Drop Zone” button. For example, one mechanism for ‘joining’ zone players together for audio play back is to link a number of zone players together to form a group. To link a number of zone players together, a user can manually link each zone player or room one after the other. For example, assume that there is a

multi-zone system that includes the following zones: Bathroom, Bedroom, Den, Dining Room, Family Room, and Foyer.

[0053] In certain embodiments, a user can link any number of the six zone players, for example, by starting with a single zone and then manually linking each zone to that zone.

[0054] In certain embodiments, a set of zones can be dynamically linked together using a command to create a zone scene or theme (subsequent to first creating the zone scene). For instance, a “Morning” zone scene command can link the Bedroom, Office, and Kitchen zones together in one action. Without this single command, the user would need to manually and individually link each zone. The single command might include a mouse click, a double mouse click, a button press, a gesture, or some other programmed action. Other kinds of zone scenes can be programmed.

[0055] In certain embodiments, a zone scene can be triggered based on time (e.g., an alarm clock function). For instance, a zone scene can be set to apply at 8:00 am. The system can link appropriate zones automatically, set specific music to play, and then stop the music after a defined duration. Although any particular zone can be triggered to an “On” or “Off” state based on time, for example, a zone scene enables any zone(s) linked to the scene to play a predefined audio (e.g., a favorable song, a predefined playlist) at a specific time and/or for a specific duration. If, for any reason, the scheduled music failed to be played (e.g., an empty playlist, no connection to a share, failed Universal Plug and Play (UPnP), no Internet connection for an Internet Radio station, and so on), a backup buzzer can be programmed to sound. The buzzer can include a sound file that is stored in a zone player, for example.

V. Example Ad-Hoc Network

[0056] Certain particular examples will now be provided in connection with FIGS. 6-8B to describe, for purposes of illustration only, certain base systems and methods to provide and facilitate connection to a playback network. FIG. 6 shows that there are three zone players **602**, **604** and **606** and a controller **608** that form a network branch that is also referred to as an Ad-Hoc network **610**. The network **610** may be wireless, wired, or a combination of wired and wireless. In general, an Ad-Hoc (or “spontaneous”) network is a local area network or other small network in which there is no one access point for all traffic. With an established Ad-Hoc network **610**, the devices **602**, **604**, **606** and **608** can all communicate with each other in a “peer-to-peer” style of communication, for example. Furthermore, devices may come/and go from the network **610**, and the network **610** will automatically reconfigure itself without needing the user to reconfigure the network **610**.

[0057] Using the Ad-Hoc network **610**, the devices **602**, **604**, **606**, and **608** can share or exchange one or more audio sources and be grouped to play the same or different audio sources. For example, the devices **602** and **604** are grouped to playback one piece of music, and at the same time, the device **606** plays back another piece of music. In other words, the devices **602**, **604**, **606** and **608**, as shown in FIG. 6, form a HOUSEHOLD that distributes audio and/or reproduces sound. As used herein, the term HOUSEHOLD (provided in uppercase letters to disambiguate from the user’s domicile) is used to represent a collection of networked devices that are cooperating to provide an application or

service. An instance of a HOUSEHOLD is identified with a household **10** (or household identifier).

[0058] In certain embodiments, a household identifier (HHID) is a short string or an identifier that is computer-generated to help ensure that it is unique. Accordingly, the network **610** can be characterized by a unique HHID and a unique set of configuration variables or parameters, such as channels (e.g., respective frequency bands), SSID (a sequence of alphanumeric characters as a name of a wireless network), and WEP keys (wired equivalent privacy or other security keys). In certain embodiments, SSID is set to be the same as HHID.

[0059] In certain embodiments, each HOUSEHOLD includes two types of network nodes: a control point (CP) and a zone player (ZP). The control point controls an overall network setup process and sequencing, including an automatic generation of required network parameters (e.g., WEP keys). In an embodiment, the CP also provides the user with a HOUSEHOLD configuration user interface. The CP function can be provided by a computer running a CP application module, or by a handheld controller (e.g., the controller **308**) also running a CP application module, for example. The zone player is any other device on the network that is placed to participate in the automatic configuration process. The ZP, as a notation used herein, includes the controller **308** or a computing device, for example.

[0060] In certain embodiments, configuration of a HOUSEHOLD involves multiple CPs and ZPs that rendezvous and establish a known configuration such that they can use a standard networking protocol (e.g., IP over Wired or Wireless Ethernet) for communication. In an embodiment, two types of networks/protocols are employed: Ethernet 802.3 and Wireless 802.11g. Interconnections between a CP and a ZP can use either of the networks/protocols. A device in the system as a member of a HOUSEHOLD can connect to both networks simultaneously. In an environment that has both networks in use, it is assumed that at least one device in a system is connected to both as a bridging device, thus providing bridging services between wired/wireless networks for others. The zone player **606** in FIG. 6 is shown to be connected to both networks, for example. The connectivity to the network **612** is based on Ethernet while the connectivity to other devices **602**, **604** and **608** is based on Wireless. It is understood, however, that in some embodiments each zone player **606**, **604**, **602** may access the Internet when retrieving media from the cloud (e.g., Internet) via the bridging device. For example, zone player **602** may contain a uniform resource locator (URL) that specifies an address to a particular audio track in the cloud. Using the URL, the zone player **602** may retrieve the audio track from the cloud, and ultimately play the audio out of one or more zone players.

VI. Example Music Sharing and Playback Configuration

[0061] Certain embodiments enable a user to stream music from a music-playing application (e.g., browser-based application, native music player, other multimedia application, and so on) to a local multimedia content playback (e.g., Sonos™) system. Certain embodiments provide secure systems and methods for multimedia content playback across a plurality of systems and locations. Certain embodiments

facilitate integration between content partners and a playback system as well as supporting maintenance of such content and system.

[0062] FIG. 7 shows a system including a plurality of networks including a cloud-based network and at least one local playback network. The network includes a plurality of playback devices or players, though it is understood that the network may contain only one playback device. In certain embodiments, each player has an ability to retrieve its content for playback. Control and content retrieval can be distributed or centralized, for example. Input can include streaming content provider input, third party application input, mobile device input, user input, and/or other playback network input into the cloud for local distribution and playback.

[0063] As illustrated by the example system 700 of FIG. 7, a plurality of content providers 720-750 can be connected to one or more local playback networks 760-770 via a cloud and/or other network 710. Using the cloud 710, a multimedia playback system 720 (e.g., Sonos™) a mobile device 730, a third party application 740, a retail location 750, and so on can provide multimedia content (requested or otherwise) to local playback networks 760, 770. Within each local network 760, 770, a controller 762, 772 and/or playback device 764, 774 can provide a song identifier, song name, playlist identifier, playlist name, genre, preference, and so on, and/or simply receive content from a connected system via the cloud.

[0064] For example, a user listens to a third party music application (e.g., Pandora™ Rhapsody™, Spotify™, and so on) on her smart phone while commuting. She's enjoying the current channel and, as she walks in the door to her home, selects an option to continue playing that channel on her household music playback system (e.g., Sonos™). The playback system picks up from the same spot on the selected channel that was on her phone and outputs that content (e.g., that song) on speakers and/or other playback devices connected to the household playback system. A uniform resource indicator (URI) (e.g., a uniform resource locator (URL)) can be passed to a playback device to fetch content from a cloud and/or other networked source, for example. A playback device, such as a zone player, can fetch content on its own without use of a controller, for example. Once the zone player has a URL (or some other identification or address) for a song and/or playlist, the zone player can run on its own to fetch the content. Songs and/or other multimedia content can be retrieved from the Internet rather than a local device (e.g., a compact disc (CD)), for example. A third party application can open or utilize an application programming interface (API) to pass music to the household playback system without tight coupling to that household playback system.

[0065] In another example of an application determining a playlist and/or other content for playback, a user enjoys listening to music on an online music service (e.g., turntable.fm or other virtual room that a user can enter to choose from a plurality of online disc jockeys (DJs) deciding what to play next) using his Mac Book Pro™ at home. He likes the unique user experience the service offers, and he frequently hops from room to room discovering new music. To maximize sound quality, he plays the music on his household playback system (e.g., Sonos™). A button or other indicator can be added to the turntable.fm Web application to switch the content being played to the playback system for output

(e.g., to the Sonos™ system rather than or in addition to the Mac Book™). While Web-based applications typically do not have access to items on a local network, certain embodiments enable a third-party Web-based application (e.g., Turntable.fm) to talk to a playback system (e.g., Sonos™) in a certain way (e.g., may have to log in with a username and password), and the identified user has the website send audio or audio and video down to a playback device (e.g., a zone player) on the playback system local network to play music there (or some other media).

[0066] In another example, a first user creates a playlist (e.g., a Spotify™ playlist). The first user visits a second user's house, pulls out her smart phone and shares her playlist by playing it on the second user's household playback (e.g., Sonos™) system using her third party (e.g., Spotify™) application. The first user may also go to the third party content provider's (e.g., Spotify's™) website and share her playlist on the second user's playback system.

[0067] Thus, certain embodiments provide cross-service linking such that a song identifier can be passed from one user and/or service to another to be fetched and played. A user having a playlist on his or her phone can visit a friend and, using her account on her friend's system, play a song to which she has an access right. A retrieved song can be streamed locally to a user's phone, or an application can pass a song identifier to a local playback system which looks up the song identifier and finds an available audio stream to which the user has a right to play and then plays that song.

[0068] In another example, a user is staying in a hotel room or other facility including a local playback network. For example, a speaker and/or other playback device (e.g., a Sonos™ Play:3, Play: 5 and so on) in a hotel room can be utilized to play multimedia content to which the user has access from his or her playback network account, streaming audio source, third party application, and so on. Content can be output to one or more devices based on availability, access, configuration, priority, preference, and so on. In certain embodiments, a playback network includes a plurality of nodes, and each node has a capability to play sound in response to an input. Requested output is provided to a most logical connection, for example.

[0069] In certain embodiments, a phone device, a television device, and so on can be used to play music, audio, video and/or other multimedia content. In an example, a push button on a microphone or household intercom system to tell the kids dinner is ready is provided over the local playback network.

[0070] FIG. 8 shows a flow diagram for a method 800 to provide audio content to a local playback system. In the example method 800 of FIG. 8, a third party application acts as a "virtual line-in" to the local playback system. At block 810, streaming of music or other content from a third party application to a local content playback system is triggered. For example, a "Play to Sonos" button is pressed on a Rhapsody™ application. At block 820, content is streamed to one or more components in a household playback network. The music may be streamed to predetermined zones or players in a household, for example. The music may be further directed to be played in different zones or players throughout the household. Playback on the local network can be facilitated to one or more zones/players based on a configuration (e.g., a zone scene, theme, and so on). Thus, certain embodiments allow a large degree of flexibility in where the music is actually played. For example, the music

can be played in the kitchen, the family room, the patio, and so on. Further, the music may be redirected to different zones.

[0071] At block **830**, the incoming content (e.g., audio) stream is provided directly from a third party application or other external source to the local playback network for playback. For example, rather than passing track identifiers, an audio stream is provided to a Sonos household system for playback to one or more configured zones. At block **840**, the local playback system consumes the stream and plays it as it would other content on the local playback (e.g., Sonos™) network (e.g., via zones and so on). At block **850**, a playback device (e.g., a zone player, Play:3™, Play:5™, and so on) adds timing information to the streaming content signal (e.g., the device takes the streaming audio signal and repackages it for local synchronized playback). In some embodiments, timing information is not added to the signal unless two or more playback devices are configured to play the audio in synchrony.

[0072] FIG. 9 shows a flow diagram for a method **900** to provide audio content to a local playback system. In the example method **900** of FIG. 9, a uniform resource indicator (URI) handler approach is provided for content output. At block **910**, a link or other reference is embedded in a third party application (e.g., Facebook™ or Twitter). At block **920**, when the link is selected (e.g., clicked), a local playback (e.g., Sonos™) controller, if available, is launched. At block **930**, the application (e.g., accessed on a phone, tablet, computer, and so on) passes a URI for associated content (e.g., an audio track and so on) to a local playback system (e.g., Sonos™) controller. At block **940**, the local controller outputs the associated content (e.g., plays the music) via the URI. For example, music is streamed from the cloud to one or more playback devices on the local playback network.

[0073] In certain embodiments, an application associated with the operating system can register to handle all URIs (URLs) that start with a certain prefix and can define how data is encoded into those URLs so a local playback system application can generate a link (e.g., “sonos:”) and put that link into a message (e.g., email, text message, instant message (IM), etc.). The local playback application registered to handle such URLs can parse the URLs to determine what song, playlist, streaming radio station, etc., to play. This launches the controller application. For example, if a first listener likes a song and tweets that song, Twitter™ can include a clickable link which launches a playback application and starts the music playing on a local playback system if the local system can find the song (e.g., if have the application, if have rights/access to the song, etc.). In certain embodiments, the system knows to trigger the receiving user’s system rather than the sending user’s system to play associated content based on the transmitted link/identifier.

[0074] For example, an application can register with the system to handle all URLs that start with a custom prefix (e.g., an HTTP “scheme”). For instance, Sonos controller apps can register to handle any URL that begins with “sonos:” or “x-sonos:”. In certain embodiments, a playback system provider can define and publish the format of its URLs so that any third party application can create a link or reference to content. A large amount of data can be encoded into a URL using query parameters, for example.

[0075] In an example, when an application tries to “open” or “browse” to a URL, the system checks to see if the scheme of the URL matches the “sonos:” scheme that has

been registered with the application. If a URL handler application is found, the system launches that application (e.g., the application can but does not need to be running in the background) and passes the URL to the application. The application then parses the URL and executes functionality based on the data in the URL. For example, the URL can contain the name of a music service and a playlist identifier from that service, plus the name of a Sonos™ Zone Player, causing the Sonos controller to start that playlist playing on that zone.

[0076] FIG. 10 shows a flow diagram for a method **1000** to provide audio content to a local playback system. In the example method **1000** of FIG. 10, at block **1010**, a link or other reference is embedded in a third party application (e.g., Facebook™). At block **1020**, when the link is selected, a playback system (e.g., Sonos™) server is contacted and provided with information regarding selected content for playback. For example, rather than launching a local controller application, a server is contacted regarding music for playback on a local network. At block **1030**, using the provided information, the server identifies and provides the content locally on a user’s local playback system. For example, the server can then start playing the music directly on the user’s Sonos™ system (e.g., without going through a Sonos™ controller application).

[0077] In certain embodiments, a “single sign-on” technology is provided so that the user does not need to re-enter a username and password in order to authenticate to the playback server. Example single sign-on technologies include Facebook Connect™, Windows Live ID™, etc.

[0078] In certain embodiments, instead of using a specialized link, such as a “sonos:” link, a normal URL can be used to point to a playback system (e.g., Sonos™) webserver, which generates links with special data embedded in the link. A playback system is identified, and content identified by the URL can be playing at via the local playback network (e.g., mesh network configured for home, hotel room, etc.). Parameters such as authentication, security, location, and so on can be configured for local playback of remote content.

[0079] FIG. 11 shows a flow diagram for a method **1100** to provide audio content to a local playback system. The example method **1100** of FIG. 11 provides a “throw it over the wall” approach to content delivery to a local playback system. At block **1110**, a third party application provides a multimedia playback device (e.g., a Sonos™ zone player (ZP)) with enough information about content (e.g., an audio track) so that, at block **1120**, the local playback system (e.g., SonosNet™) can directly access a source of the content and, at block **1130**, play the content directly off the network (e.g., the Internet) or cloud.

[0080] In certain embodiments, a local playback controller application is not involved. Information passed over to the local playback device may include an identifier for a single track, a playlist, a streaming radio station, a programmed radio station, and so on. This information can also include a current play position within a list to enable near-seamless “handoff” of music from a portable device to a local playback system. Once the music information is handed from the third-party application to the local playback system, there is no further synchronization between the two systems.

[0081] A connection between the third-party application and the local playback device (e.g., Sonos ZonePlayer™) can be direct over a local area network (LAN), remote through a proxy server in the cloud, and so on. A LAN

delivery approach may be easier to integrate into “native” applications (e.g., applications written for iOS or Android), and a proxy server approach may be easier for third party applications that are browser-based, for example.

[0082] In certain embodiments, information is provided from a third party application to a local playback system without being routed through or by a controller application. Here, the third party application is communicating with the multimedia playback device (e.g., a Sonos ZonePlayer™). Information can be passed locally, rather than through the Internet, for example. The local playback device accesses the Internet to find content to stream, and the third party application takes the place of the controller application (e.g., throw it over the wall—the application passes information and the local playback system runs it).

[0083] Certain embodiments provide an approach similar to the “throw it over the wall” or one way communication approach of FIG. 11 except that the third party application not only tells the local playback system what to play, but also maintains two-way communication with the local playback (e.g., Sonos™) system. Two-way communication helps enable features such as keeping a local playback queue synchronized with a queue that the user is editing/managing in the third party application; allow the third party application to know what is currently playing on the local playback system; allow integrated transport control between the third party application and the local playback system; and so on.

[0084] In certain embodiments, a local playback system can pass information back to a third party application to indicate a current point of playback (e.g., now playing a third song in a playlist, fourth song in the playlist, and so on). The local playback system can pass parameter information, such as a change in volume, from a local multimedia playback device to the third party application so the application can reflect the change in volume to the user via its graphical user interface. The third party application can instruct the local playback system to skip a song, go to a certain location, and so on.

[0085] Certain embodiments provide a third party mode that allows users to select from any local playback network (e.g., Sonos™) controller to listen to audio from one or more third party applications on their smartphones or tablets (e.g., Android™ devices). For example, a user may be using a local playback network controller application and now wants a third party application to appear as an audio source within the controller application. The user can then select the controller application that he or she wishes to play audio from the third party application, for example.

[0086] Certain embodiments provide queue management to allow a third party application to control a local playback queue. That is, the local playback system has a queue, but the third party application allows users to add, delete and so on from the queue, for example. Rather than switch from content that the user is currently playing, the local playback system allows a user to create a playlist on the fly. For example, if last.fm users vote that they do not like a song and it should be skipped, then the local playback system will skip it.

[0087] Certain embodiments allow a third party application to override a local playback queue with its own application-specific queue. The local playback system periodically fetches a short list of tracks to play next. The list of tracks to play is determined by the third-party application, for example. In certain embodiments, a shared queue is

provided between the local playback system and the third party application to keep the local system and application synchronized.

[0088] Certain embodiments allow control of playback system functions and/or settings via an external (e.g., third party) application. For example, a local playback system can allow volume control, play/pause, and so on and can interact with an application running on a given platform/operating system (OS). Certain embodiments provide a Web API that can be used to access functionality.

[0089] Certain embodiments facilitate control of a local playback system from outside a household or other location at which the local playback network is configured. For example, a user can queue up music while away from his or her house. The application can facilitate setup and/or configuration. For example, a third party application may ask the user to enter a Sonos customer email address and password. The application can then make a request to a Sonos server in the cloud to determine the zone groups on which music can be played.

[0090] Various inventions have been described in sufficient detail with a certain degree of particularity. It is understood to those skilled in the art that the present disclosure of embodiments has been made by way of examples only and that numerous changes in the arrangement and combination of parts can be resorted to without departing from the spirit and scope of the present disclosure as claimed. While the embodiments discussed herein can appear to include some limitations as to the presentation of the information units, in terms of the format and arrangement, the embodiments have applicability well beyond such embodiment, which can be appreciated by those skilled in the art. Accordingly, the scope of the present disclosure is defined by the appended claims rather than the forgoing description of embodiments.

We claim:

1. A network device comprising:

at least one processor;

a non-transitory computer-readable medium; and

program instructions stored on the non-transitory computer-readable medium that, when executed by the at least one processor, cause the network device to perform functions comprising:

operating in a first playback-control mode comprising:

retrieving, via a first network interface, first media identified in a playback queue;

streaming, via a second network interface to a headphone, the retrieved first media; and

causing the headphone to play back the streamed first media;

while streaming the retrieved first media to the headphone, displaying a representation of one or more playback devices in a media playback system;

while displaying the representation of the one or more playback devices, receiving user input indicating a selection of at least one given playback device from the one or more playback devices;

based on receiving the user input, causing the at least one given playback device to begin playback of second media identified in the playback queue, wherein causing the at least one given playback device to begin playback of the second media identified in the playback queue comprises configuring the at least one given playback device to:

- obtain data indicating at least the second media identified in the playback queue;
 use the obtained data to retrieve the second media identified in the playback queue; and
 play back the retrieved second media; and
 after receiving the user input:
 stopping streaming the retrieved first media; and
 transitioning from i) operating in the first playback-control mode to ii) operating in a second playback-control mode in which the network device is configured to control the at least one given playback device's playback of the second media identified in the playback queue.
2. The network device of claim 1, wherein the first media and the second media each comprise a different respective portion of a same media content.
3. The network device of claim 1, further comprising program instructions stored on the non-transitory computer-readable medium that, when executed by the at least one processor, cause the network device to perform functions comprising:
 obtaining data indicating at least the first media identified in the playback queue, wherein (i) the data indicating at least the first media identified in the playback queue and (ii) the data indicating at least the second media identified in the playback queue each comprise a same media identifier.
4. The network device of claim 1, wherein the data indicating at least the second media identified in the playback queue comprises a uniform resource indicator (URI).
5. The network device of claim 1, wherein the data indicating at least the second media identified in the playback queue comprises a track identifier.
6. The network device of claim 1, wherein the data indicating at least the second media identified in the playback queue comprises a playlist identifier.
7. The network device of claim 1, wherein the data indicating at least the second media identified in the playback queue comprises a list of track identifiers.
8. The network device of claim 1, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming, via the second network interface to the headphone, the retrieved first media comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming the retrieved first media in accordance with an IEEE 802.15 wireless standard.
9. The network device of claim 1, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming, via the second network interface to the headphone, the retrieved first media comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming the retrieved first media in accordance with a short-range wireless standard.
10. The network device of claim 1, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming, via the second network interface to the headphone, the retrieved first media comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming the retrieved first media in accordance with a wireless personal area network standard.
11. The network device of claim 1, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming, via the second network interface to the headphone, the retrieved first media comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising streaming the retrieved first media in accordance with a Bluetooth standard.
12. The network device of claim 1, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising retrieving, via the first network interface, the first media identified in the playback queue comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising receiving the first media identified in the playback queue in accordance with an IEEE 802.11 wireless standard.
13. The network device of claim 1, wherein, for at least a portion of a time period during which the network device is operating in the second playback-control mode, at least a portion of the playback queue is maintained as a remote playback queue provided by a cloud-based computing system.
14. The network device of claim 1, wherein, for at least a portion of a time period during which the network device is operating in the first playback-control mode, at least a portion of the playback queue is maintained as a local playback queue stored by the network device.
15. The network device of claim 1, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising configuring the least one playback device to obtain the data indicating at least the second media identified in the playback queue comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising configuring the at least one playback device to obtain, from a cloud-based computing system, the data indicating at least the second media.
16. The network device of claim 15, wherein the data indicating at least the second media comprises multiple media identifiers.
17. The network device of claim 1, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising configuring the at least one playback device to obtain the data indicating at least the second media identified in the playback queue comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising configuring the at least one playback device to obtain, from the network device, the data indicating at least the second media, and wherein the network device further comprises program instructions stored on the non-transitory computer-readable medium that, when executed by the at least one processor, cause the network device to perform functions comprising:
 sending, via the first network interface to the at least one playback device, the data indicating at least the second media.
18. The network device of claim 1, wherein each of the one or more playback devices in the media playback system

is communicatively coupled, over a digital data network via the first network interface, to the network device.

19. The network device of claim **1**, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising displaying the representation of the one or more playback devices in the media playback system comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising displaying an indication that the one or more playback devices are available to accept playback responsibility for the playback queue.

20. The network device of claim **1**, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising causing the at least one given playback device to begin playback of second media identified in the playback queue comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising causing the at least one given playback device to begin playback of second media identified in the playback queue in synchrony with a group of at least one other playback device of the one or more playback devices in the media playback system, and wherein the network device further comprises program instructions stored on the non-transitory computer-readable medium that, when executed by the at least one processor, cause the network device to perform functions comprising:

causing the at least one given playback device and the at least one other playback device to form a synchrony group.

21. The network device of claim **1**, further comprising program instructions stored on the non-transitory computer-readable medium that, when executed by the at least one processor, cause the network device to perform functions comprising:

transmitting, to a cloud-based computing system, an indication of a current point of playback of the playback queue by the headphone.

22. The network device of claim **21**, wherein the program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising transmitting the indication of the current point of playback of the playback queue by the headphone comprise program instructions that, when executed by the at least one processor, cause the network device to perform functions comprising transmitting an indication of a current point of playback of the first media by the headphone.

23. The network device of claim **1**, further comprising: at least one transducer; and

program instructions stored on the non-transitory computer-readable medium that, when executed by the at least one processor, cause the network device to perform functions comprising:

operating in a third playback-control mode comprising: receiving, via the first network interface, third media identified in the playback queue; and playing back, via the at least one transducer, the received third media.

24. The network device of claim **23**, wherein the first media, the second media, and the third media each comprise a different respective portion of a same media content.

25. The network device of claim **1**, wherein the headphone comprises a first headphone, and wherein the network

device further comprises program instructions stored on the non-transitory computer-readable medium that, when executed by the at least one processor, cause the network device to perform functions comprising:

while operating in the first playback-control mode:

streaming, via the second network interface to a second headphone, the retrieved first media; and causing the second headphone to play back the streamed first media.

26. A system comprising:

a network device configured to communicate over at least one data network; and

a given playback device configured to communicate over at least one data network;

the network device comprising:

at least one first processor;

a first non-transitory computer-readable medium; and program instructions stored on the first non-transitory computer-readable medium that, when executed by the at least one first processor, cause the network device to perform functions comprising:

operating in a first playback-control mode comprising:

retrieving, via a first network interface, first media identified in a playback queue;

streaming, via a second network interface to a headphone, the retrieved first media; and

causing the headphone to play back the streamed first media;

while streaming the retrieved first media to the headphone, displaying a representation of one or more playback devices in a media playback system, the one or more playback devices in the media playback system comprising the given playback device;

while displaying the representation of the one or more playback devices, receiving user input indicating a selection of at least the given playback device from the one or more playback devices;

based on receiving the user input, causing at least the given playback device to begin playback of second media identified in the playback queue, wherein causing at least the given playback device to begin playback of the second media identified in the playback queue comprises configuring at least the given playback device to:

obtain data indicating at least the second media identified in the playback queue;

use the obtained data to retrieve the second media identified in the playback queue; and

play back the retrieved second media; and

after receiving the user input:

stopping streaming the retrieved first media; and transitioning from i) operating in the first playback-control mode to

ii) operating in a second playback-control mode in which the network device is configured to control at least the given playback device's playback of the second media identified in the playback queue;

the given playback device comprising:

at least one second processor;

a second non-transitory computer-readable medium; and program instructions stored on the second non-transitory computer-readable medium that are executable by the at least one second processor such that the given playback device is configured to:

begin playback of the second media identified in the playback queue.

27. The system of claim 26, further comprising program instructions stored on the second non-transitory computer-readable medium that, when executed by the at least one second processor, cause the given playback device to perform functions comprising:

obtaining, from a cloud-based computing system, the data indicating at least the second media.

28. A system comprising:

a network device configured to communicate over at least one data network; and

a cloud-based computing device configured to communicate over at least one data network;

the network device comprising:

at least one first processor;

a first non-transitory computer-readable medium; and program instructions stored on the first non-transitory computer-readable medium that, when executed by the at least one first processor, cause the network device to perform functions comprising:

operating in a first playback-control mode comprising:

retrieving, via a first network interface, first media identified in a playback queue;

streaming, via a second network interface to a headphone, the retrieved first media; and

causing the headphone to play back the streamed first media;

while streaming the retrieved first media to the headphone, displaying a representation of one or more playback devices in a media playback system;

while displaying the representation of the one or more playback devices, receiving user input indicating a selection of at least one given playback device from the one or more playback devices;

based on receiving the user input, causing the at least one given playback device to begin playback of second media identified in the playback queue, wherein causing the at least one given playback device to begin playback of the second media identified in the playback queue comprises configuring the at least one given playback device to:

obtain data indicating at least the second media identified in the playback queue;

use the obtained data to retrieve the second media identified in the playback queue; and

play back the retrieved second media; and

after receiving the user input:

stopping streaming the retrieved first media; and

transitioning from i) operating in the first playback-control mode to ii) operating in a second playback-control mode in which the network device is configured to control the at least one given playback device's playback of the second media identified in the playback queue;

the cloud-based computing device comprising:

at least one second processor;

at least one second non-transitory computer-readable medium; and

program instructions collectively stored on the at least one second non-transitory computer-readable medium that are executable by the at least one second processor such that the cloud-based computing device is configured to: transmit, to the given playback device, the second media identified in the playback queue.

29. The system of claim 28, further comprising program instructions stored on the at least one second non-transitory computer-readable medium that, when executed by the at least one second processor, cause the cloud-based computing device to perform functions comprising:

transmit, to the given playback device, the data indicating at least the second media identified in the playback queue.

30. A non-transitory computer-readable medium, wherein the non-transitory computer-readable medium is provisioned with program instructions that are executable to cause a network device to perform functions comprising:

operating in a first playback-control mode comprising:

retrieving, via a first network interface, first media identified in a playback queue;

streaming, via a second network interface to a headphone, the retrieved first media; and

causing the headphone to play back the streamed first media;

while streaming the retrieved first media to the headphone, displaying a representation of one or more playback devices in a media playback system;

while displaying the representation of the one or more playback devices, receiving user input indicating a selection of at least one given playback device from the one or more playback devices;

based on receiving the user input, causing the at least one given playback device to begin playback of second media identified in the playback queue, wherein causing the at least one given playback device to begin playback of the second media identified in the playback queue comprises configuring the at least one given playback device to:

obtain data indicating at least the second media identified in the playback queue;

use the obtained data to retrieve the second media identified in the playback queue; and

play back the retrieved second media; and

after receiving the user input:

stopping streaming the retrieved first media; and

transitioning from i) operating in the first playback-control mode to ii) operating in a second playback-control mode in which the network device is configured to control the at least one given playback device's playback of the second media identified in the playback queue.

* * * * *