

Filed on behalf of Oracle America, Inc.
By: Monica Grewal, Reg. No. 40,056 (Lead Counsel)
Donald Steinberg, Reg. No. 37,241 (Back-up Counsel)
Wilmer Cutler Pickering Hale and Dorr LLP
60 State Street
Boston, MA 02109
Phone: (617) 526-6223
Email: Monica.Grewal@wilmerhale.com
Donald.Steinberg@wilmerhale.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ORACLE AMERICA, INC.

Petitioner

v.

REALTIME DATA LLC

Patent Owner of

U.S. Patent No. 7,415,530 to Fallon

IPR Trial No. IPR2016-00376

DECLARATION OF PROFESSOR JAMES A. STORER, Ph.D.

CLAIM 24

U.S. PATENT NO. 7,415,530

UNDER 35 U.S.C. § 312 AND 37 C.F.R. § 42.104

TABLE OF CONTENTS

I.	BACKGROUND AND QUALIFICATIONS	4
II.	MATERIALS REVIEWED	7
III.	THE RELEVANT LEGAL STANDARDS	8
	A. Claim Construction	8
	B. Obviousness.....	9
IV.	STATE OF THE ART AT THE TIME OF THE '530 PATENT	12
V.	OVERVIEW OF THE '530 PATENT	14
	A. Brief Description	14
	B. Summary of the Prosecution History	19
VI.	CLAIM CONSTRUCTION	20
	A. The term “said compression and storage occurs faster than said data stream is able to be stored on said memory device in said received form”.....	20
	B. The term “bandwidth”	21
VII.	PRIOR ART REFERENCES	22
	A. Kawashima	22
	B. Sebastian.....	30
	C. Chu	33
VIII.	THE CHALLENGED CLAIM IS NOT PATENTABLE.....	36
IX.	IDENTIFICATION OF HOW CLAIM 24 IS UNPATENTABLE	36
	A. Claim 24 is Obvious in View of Kawashima, Sebastian, and Chu.....	36
	1. Motivation to Combine Kawashima, Sebastian, and Chu	36
	2. Independent Claim 24 is Obvious in View of Kawashima, Sebastian and Chu.....	39
	i. The preamble of Claim 24 is disclosed in Kawashima. .	40
	ii. Limitation A of Claim 24 is disclosed in Kawashima....	42
	iii. Limitation B of Claim 24 is disclosed in Kawashima....	42
	iv. Limitation C of Claim 24 is disclosed in Kawashima....	44

IPR2016-00376: Storer Declaration
U.S. Patent No. 7,415,530, Claim 24

v.	Limitation D of Claim 24 is obvious in view of Kawashima and Sebastian.	45
vi.	Limitation E of Claim 24 is disclosed in Kawashima.	46
vii.	Limitation F of Claim 24 is obvious in view of Kawashima and Sebastian, as well as obvious in view of Kawashima and Chu.	47
viii.	Limitation G of Claim 24 is obvious in view of Kawashima, Sebastian, and Chu.	52
ix.	Limitation H of Claim 24 is disclosed in Kawashima.	56
x.	Limitation I of Claim 24 is disclosed in Kawashima.	58
xi.	Limitation J of Claim 24 is obvious in view of Kawashima and Sebastian.	60
xii.	Limitation K of Claim 24 is obvious in view of Kawashima and Sebastian.	61

I, Prof. James A. Storer, Ph.D., declare as follows:

I. BACKGROUND AND QUALIFICATIONS

1. My name is James Storer. I am a Professor at Brandeis University in the Computer Science Department. I am an expert in the field of computer algorithms, including data communications and internet related computing, data compression, data and image retrieval, storage and processing of large data sets, and image/video processing. I have studied, researched, and practiced in the field of computer science for more than 35 years, and have taught Computer Science at Brandeis since 1981.

2. I received my Doctor of Philosophy (Ph.D.) degree in the field of Computer Science from Princeton University in 1979. I received my Masters of Arts (M.A.) degree in Computer Science from Princeton University and my Bachelor of Arts (B.A.) degree in Mathematics and Computer Science from Cornell University.

3. After receiving my Ph.D. degree, I worked in industry as a researcher at AT&T Bell Laboratories from 1979 to 1981 before joining the faculty of Brandeis University.

4. I have been involved in computer science research since 1976. My research has been funded by a variety of governmental agencies, including the National Science Foundation (NSF), National Aeronautics and Space

(DARPA). In addition, I have received government Small Business Innovation Research (SBIR) funding, as well as numerous industrial grants.

5. I regularly teach courses in software and hardware technology for data compression and communications (including text, images, video, and audio) at both the undergraduate and graduate level, and in my capacity as co-chair of the Annual Data Compression Conference, I regularly referee academic papers in these areas. In addition, much of my consulting activity has been in the areas of software and hardware for consumer electronic devices, including cell phones/PDAs (including cellular technology), smartphones, digital cameras, digital video and audio recorders, and personal computers (“PCs”), as well as devices for communications over the Internet.

6. I am the author of two books: *An Introduction to Data Structures and Algorithms* and *Data Compression: Methods and Theory* (Ex. 1115). Both books have been used as references for undergraduate level computer science courses in universities. I am the editor or co-editor of four other books, including *Hyperspectral Data Compression* and *Image and Text Compression*.

7. I have three issued U.S. patents that relate to computer software and hardware (two for which I am sole inventor and one for which I am co-inventor).

IPR2016-00376: Storer Declaration

U.S. Patent No. 7,415,530, Claim 24

These are submitted herewith as Exhibits 1106, 1119-1120. I am the author or co-author of well over 100 articles and conference papers.

8. In 1991, I founded the Annual Institute of Electrical and Electronics Engineers (IEEE) Data Compression Conference (DCC), the first major international conference devoted entirely to data compression, and have served as the conference chair since then.

9. I routinely serve as referee for papers submitted to journals such as, for example, JACM, SICOMP, Theoretical CS, Computer Journal, J. Algorithms, Signal Processing, JPDC, Acta Informatica, Algorithmica, IPL, IPM, Theoretical CS, J. Algorithms, Networks, IEEE J. Robotics & Automation, IEEE Trans. Information Theory, IEEE Trans. Computers, IEEE Trans. Image Processing, Proceedings of the IEEE, IBM J. of R&D, and J. Computer and System Sciences.

10. I have served as guest editor for a number of professional journals, including Proceedings of the IEEE, Journal of Visual Communication and Image Representation, and Information Processing and Management. I have served as a program committee member for various conferences, including IEEE Data Compression Conference, IEEE International Symposium on Information Theory, Combinatorial Pattern Matching (CPM), International Conference on String Processing and Information Retrieval (SPIRE), Conference on Information and Knowledge Management (CIKM), Conference on Information Theory and

IPR2016-00376: Storer Declaration

U.S. Patent No. 7,415,530, Claim 24

Statistical Learning (ITSL), Sequences and Combinatorial Algorithms on Words,

Dartmouth Institute for Advanced Graduate Studies Symposium (DAGS),

International Conference on Language and Automata Theory and Applications

(LATA), DIMACS Workshop on Data Compression in Networks and

Applications, Conference on Combinatorial Algorithms on Words.

11. A copy of my latest *curriculum vitae* (C.V.) is attached as Appendix A.

12. My compensation is in no way contingent on the results of these or any other proceedings relating to the above-captioned patent.

II. MATERIALS REVIEWED

13. I have carefully reviewed U.S. Patent No. 7,415,530 patent (“the ’530 patent”).

14. For convenience, a list of the information that I considered in arriving at my opinions is attached as Appendix B. I understand that the prior art relied on in this petition includes:

- U.S. Patent No. 5,805,932 to Kawashima (“Kawashima”, Ex. 1103), which was filed on February 13, 1996, and issued on September 8, 1998. I have been informed that Kawashima qualifies as prior art.
- U.S. Patent No. 6,253,264 to Sebastian (“Sebastian”, Ex. 1104), which was filed on March 6, 1998, claims priority to U.S. Provisional

Application No. 60/036,548 (filed on March 7, 1997), and issued on June 26, 2001. I have been informed that Sebastian qualifies as prior art.

- U.S. Patent No. 5,467,087 to Chu (“Chu”, Ex. 1105), which was filed on December 18, 1992, and issued on November 14, 1995. I have been informed that Chu qualifies as prior art.

15. Based on my review of these materials, I believe that the relevant field for purposes of the ’530 patent is systems and methods of data compression.

16. As described in Section I above, I have extensive experience in computer science and data compression. Based on my experience, I have a good understanding of the relevant field in the relevant timeframe (which is discussed in Section IV below).

III. THE RELEVANT LEGAL STANDARDS

17. I am not an attorney. For the purposes of this declaration, I have been informed about certain aspects of the law that are relevant to my opinions. My understanding of the law is as follows:

A. Claim Construction

18. I have been informed that claim construction is a matter of law and that the final claim construction will ultimately be determined by the Board. For the purposes of my invalidity analysis in this proceeding and with respect to the

prior art, I have applied the broadest reasonable construction of the claim terms as they would be understood by one skilled in the relevant art.

19. I have been informed and understand that a claim in *inter partes* review is given the “broadest reasonable construction in light of the specification.” 37 C.F.R. § 42.100(b). I have also been informed and understand that any claim term that lacks a definition in the specification is therefore also given a broad interpretation.

B. Obviousness

20. It is my opinion that a person of ordinary skill in the art at the time the ’530 patent was effectively filed, is a person who has an undergraduate degree in computer science and two years’ industry experience or a graduate degree in the field of computer science.

21. Based on my experience, I have an understanding of the capabilities of a person of ordinary skill in the relevant field. I have supervised and directed many such persons over the course of my career. Further, I had at least those capabilities myself at the time the patent was filed.

22. The analysis set forth herein evaluates obviousness consistent with the foregoing principles and through the eyes of one of ordinary skill in the art at the time of filing (which is discussed in Section IX below).

23. I have been informed and understand that a patent claim can be considered to have been obvious to a person of ordinary skill in the art at the time the application was filed. This means that, even if all of the requirements of a claim are not found in a single prior art reference, the claim is not patentable if the differences between the subject matter in the prior art and the subject matter in the claim would have been obvious to a person of ordinary skill in the art at the time the application was filed.

24. I have been informed and understand that a determination of whether a claim would have been obvious should be based upon several factors, including, among others:

- the level of ordinary skill in the art at the time the application was filed;
- the scope and content of the prior art;
- what differences, if any, existed between the claimed invention and the prior art.

25. I have been informed and understand that the teachings of two or more references may be combined in the same way as disclosed in the claims, if such a combination would have been obvious to one having ordinary skill in the art. In determining whether a combination based on either a single reference or

multiple references would have been obvious, it is appropriate to consider, among other factors:

- whether the teachings of the prior art references disclose known concepts combined in familiar ways, and when combined, would yield predictable results;
- whether a person of ordinary skill in the art could implement a predictable variation, and would see the benefit of doing so;
- whether the claimed elements represent one of a limited number of known design choices, and would have a reasonable expectation of success by those skilled in the art;
- whether a person of ordinary skill would have recognized a reason to combine known elements in the manner described in the claim;
- whether there is some teaching or suggestion in the prior art to make the modification or combination of elements claimed in the patent; and
- whether the innovation applies a known technique that had been used to improve a similar device or method in a similar way.

26. I understand that one of ordinary skill in the art has ordinary creativity, and is not an automaton.

27. I understand that in considering obviousness, it is important not to determine obviousness using the benefit of hindsight derived from the patent being considered.

28. Given this standard, the Board should conclude, based on the information in this petition, that the challenged claims are merely a predictable combination of old elements that are used according to their established functions.

IV. STATE OF THE ART AT THE TIME OF THE '530 PATENT

29. Using multiple compression techniques to accelerate the transmission/storage of data has been one of the primary applications of compression since compression was first used in the 1950s and 1960s, and continues today, as described in more detail below. *See generally* D. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” Proceedings of the IRE 40, 1098-1101 (1952) (Ex. 1116); U.S. Patent No. 3,394,352 (“Wernikoff”, Ex. 1113); J. Ziv and A. Lempel, “A Universal Algorithm for Sequential Data Compression,” IEEE Transactions on Information Theory 23:3, 337-343 (1977) (Ex. 1117); J. Ziv and A. Lempel, “Compression of Individual Sequences Via Variable-Rate Coding,” IEEE Transactions on Information Theory (1978) (Ex. 1118). In addition, rate control (adjusting system parameters to accommodate bandwidths) was well known in the art, and including in established standards (e.g., MPEG video compression standard).

30. U.S. Patent No. 4,956,808 (“Aakre”) discloses a system for managing efficient compression and storage of data by controlling a continuous flow of compressed data being written to a storage device. Aakre at 1:54-2:4; Ex. 1112. Aakre controls the flow of compressed data “as a function of the data acceptance rate of the [storage] medium” through a buffering mechanism. *See id.* at 1:66-2:4. Aakre’s method has the advantage of operating the storage device “in a continuous or streaming mode thus reducing the time required for the save operations.” *Id.* at 3:13-17.

31. U.S. Patent No. 5,794,229 (“French”) discloses a system for compression and decompression of the data pages. French at Abstract; Ex. 1111. The system uses “flags” to select the optimal type of compression from an array of compression algorithms. *Id.* at 4:18-26. “The pages are further optimized for compression by storing in the page header a status flag indicating whether the data page is a candidate for compression and (optionally) what type of compression is best suited for the data on that page.” *Id.* As “objects are streamed to disk . . . , Manager in turn stores the object on disk using the best compression methodology known to it . . . [such that] data compression is transparent.” *Id.* at 4:27-45, 16:17-24. “Actual compression methodology can be provided using commercially available compression/decompression libraries,” such as “LZ221,” “LZRW1,” “LZW” “PKZIP,” and “PKWare.” *Id.* at 16:25-42.

32. U.S. Patent No. 4,593,324 (“Ohkubo”) describes compressing and transferring image data between a main memory and an auxiliary memory. Ohkubo at 1:60-68; Ex. 1110. As described by Ohkubo, using the data compression/expansion device during transfer between the memories results in increasing the data transfer rate between the main and auxiliary memories, allowing more data to be stored in the auxiliary memory. *Id.* at 4:7-14. Ohkubo manages this by compressing image data at an average compression rate faster than the rate at which data may be stored to the magnetic disk. *Id.* at 2:60-3:16. In some embodiments, “the data compression/expansion device 6 employs a one-dimensional run length coding system; however, this system may be replaced by a two-dimensional coding system.” *Id.* at 3:59-65. Ohkubo describes that “the image data transfer rate between the main memory 5 and the data compression/expansion device 6 is greatly increased, i.e. to 40 M bits/sec at maximum.” *Id.* at 3:25-34.

V. OVERVIEW OF THE ’530 PATENT

A. Brief Description

33. The ’530 patent, titled “Systems and Methods for Accelerated Data Storage and Retrieval,” was filed on October 26, 2006. I understand that the ’530 patent claims priority to several U.S. patent applications, the earliest of which was

34. The '530 patent is directed to systems and methods for providing “accelerated” data storage and retrieval ('530 Patent at Abstract; Ex. 1101) and allegedly teaches systems and methods for improving data storage and retrieval “bandwidth” by using lossless data compression and decompression (*id.* at 4:42-44).

35. Figure 8 illustrates a detailed block diagram of a system for accelerated data storage according to the '530 patent's preferred embodiment:

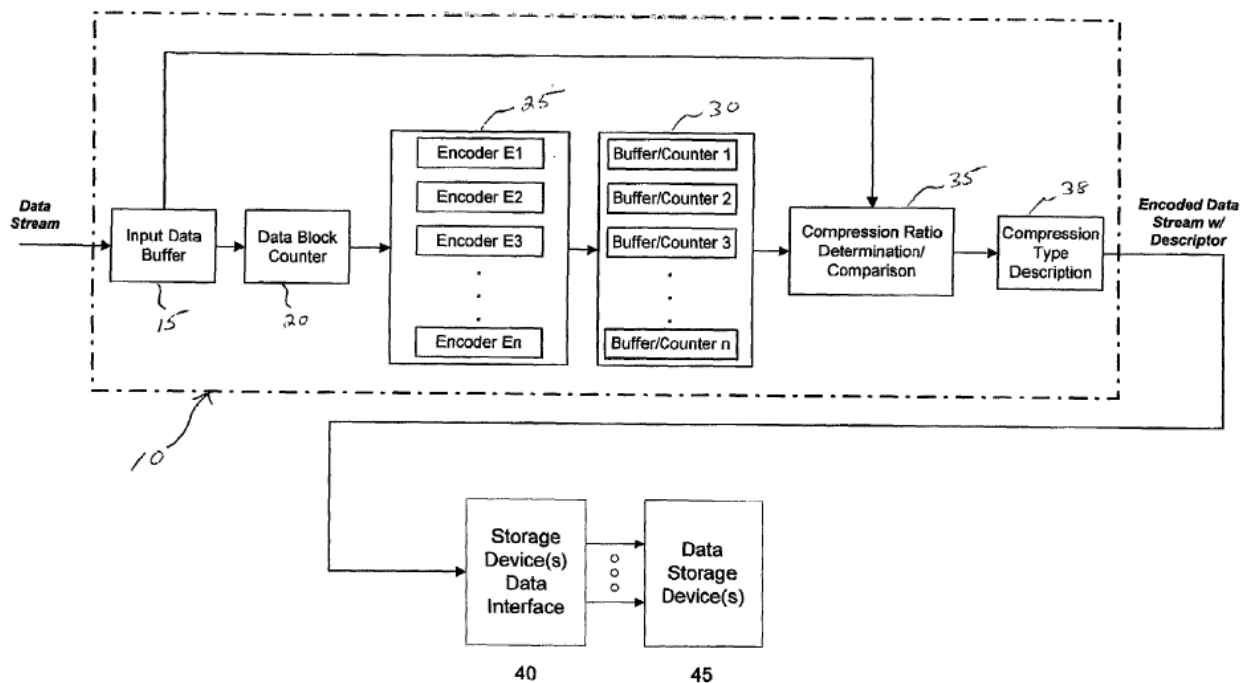


FIGURE 8

36. As shown above, the claimed “data storage accelerator” (10) receives an incoming “data stream” of “data blocks” and optionally stores the blocks in the “input data buffer” (15) and sends the blocks to the “data block counter” (20), where data blocks’ sizes are measured and recorded. *See id.* at 11:23-34. One of ordinary skill in the art would understand that a buffer, such as the “input data buffer” (15), is typically random access memory (RAM). The ’530 patent states that “the input buffer 15 and counter 20 are not required elements of the present invention.” *Id.* at 11:23-51.

37. The ’530 patent explains that the data blocks received and compressed by the “data storage accelerator” may range in size (also referred to in the art as “length”) from “individual bits through complete files or collections of multiple files,” and that they may be either fixed or variable in size. *Id.* at 11:26-30. The “data block counter” “counts” or “otherwise enumerates the size” of the data blocks in “any convenient units including bits, bytes, words, double words.” *Id.* at 11:30-33; Ex. 1101.

38. Data compression is performed by the “encoder module” (25). *Id.* at 11:40. This module may include any number of encoders (i.e., compression engines represented in Figure 8 as “E1,” E2,” E3,” and “En”) that may use any number of the lossless compression techniques “currently well known within the art” such as “run length, Huffman, Lempel-Ziv Dictionary Compression,

arithmetic coding, data compaction, and data null suppression.” *Id.* at 11:40-46; *see also id.* at 12:41-46. The ’530 patent discloses that the compression techniques may be selected based upon their “ability to effectively encode different types of input data” (*id.* at 11:47-48), that more than one encoder may use the same compression technique (*id.* at 12:41-46), and the compression process may be performed in parallel or sequentially (*id.* at 11:62-64). In other words, using the compression process described in the ’530 patent, either the same or different data blocks may be compressed by different encoders simultaneously (in parallel), or the same or different data blocks may be compressed by different encoders sequentially, one block at a time.

39. After a data block is compressed by the “encoder module,” it may be buffered and its newly compressed size may be measured or “counted” by the “buffer/counter module” (30). *Id.* at 12:14-16. Next, the “compression ratio module” (35) determines the “compression ratio” obtained for each of the encoders by calculating the ratio of the size of the uncompressed data block to the size of the compressed block. *Id.* at 12:20-25. If, for example, a single data block is compressed by several different encoders $E_1 \dots E_n$, each using a different compression technique, the “compression ratio module” may also compare each calculated ratio with an “a priori-specified compression ratio threshold limit” to determine if at least one of the compressed blocks were compressed at an equal or

greater ratio. *See id.* at 12:25-30. If at least one of the compressed blocks was compressed at an equal or greater ratio, then the block compressed with the highest ratio is transmitted/stored. *Id.* at 12:46-49. If none of the compressed blocks were compressed at an equal or greater ratio, then the uncompressed block is transmitted/stored. *Id.* at 12:49-53.

40. Before the uncompressed or compressed block is transmitted/stored, the “description module” or “compression type description” module (38) “appends” a descriptor to the block indicating, for a compressed block, the compression technique that was used, or else a “null” descriptor indicating that the block was not compressed. *Id.* at 12:33-59. The block, with its appended descriptor, is then transmitted/stored, and the descriptor is used for “subsequent data processing, storage, or transmittal.” *Id.*

41. The ’530 patent describes that “accelerated data storage and retrieval” is achieved “by utilizing lossless data compression and decompression.” *Id.* at 2:58-60. For example, data storage can be “accelerated” by compressing an input data stream at a compression ratio (e.g., 3:1) that is at least equal to the ratio of the input data transmission rate (e.g., 60 megabytes per second) to the data storage rate (e.g., 20 megabytes per second) “so as to provide continuous storage of the input data stream at the input data transmission rate.” *Id.* at 3:13-18; *see also id.* at 5:29-43. By compressing the data at this compression ratio, 60 megabytes worth of

compressed data can be stored per second, even though the target storage device is capable of storing only 20 megabytes per second, thus “accelerating” the storage of data.

B. Summary of the Prosecution History

42. I understand that the ’530 patent has undergone one reexamination: Reexamination No. 95/001,927 (“the ’927 reexamination”), filed on March 2, 2012. I also understand that independent claims 1 and 24 were found patentable by the examiner. ’927 Reexamination, 5/31/13 Right of Appeal Notice at 6-14; Ex. 1107. I have been informed that the examiner found that the primary references relied upon by the third party requester, U.S. Patent Nos. 4,956,808 (“Aakre”, Ex. 1112), 4,593,324 (“Ohkubo”, Ex. 1110), and 5,150,430 (“Chu ’430”, Ex. 1109)¹, did not disclose the following limitation: “said compression and storage occurs faster than said data stream is able to be stored on said memory device in said received form.” *Id.* I have also been informed that while the examiner found that references before him did disclose “fast” compression and storage systems, he concluded that none specified that those systems compressed and stored faster than storage of the uncompressed stream could otherwise occur. *See id.* at 6 (citing Aakre (Ex. 1112) at Abstract, 1:54-59, 1:62-2:4, 2:10-13, 2:17-21, 2:43-47, 3:30-

¹ I understand that Chu ’430 is a different patent than the Chu patent (U.S. Patent No. 5,467,087) that provides the basis for the ground of this petition.

50), 7 (citing Okhubo (Ex. 1110) at Abstract, 1:54-67, 2:62-63, 2:67-3:8, 3:29-31, 4:10-14, 4:22-29), 8 (citing Chu '430 (Ex. 1109) at 3:65-68, 4:10-12, 4:16-20, 4:24-38, 6:4-28, 17:3-6, and Figure 2).

VI. CLAIM CONSTRUCTION

43. It is my understanding that a claim in *inter partes* review is given the “broadest reasonable construction in light of the specification,” as mentioned in Section III.A above.

A. The term “said compression and storage occurs faster than said data stream is able to be stored on said memory device in said received form”

44. Independent claim 24 of the '530 patent recites the term, “said compression and storage occurs faster than said data stream is able to be stored on said memory device in said received form.” The proposed construction is *wherein the time to compress and store the received data stream is less than the time to store the received data stream without compressing it*.

45. As explained in Section V.A above, the '530 patent is generally directed to “accelerated data storage and retrieval” ('530 Patent at Abstract; Ex. 1101). The '530 patent teaches that storage “acceleration” can be achieved by receiving a data stream at an input data transmission rate (e.g., 80 megabytes per second) that is greater than the data storage rate of a target storage device (e.g., 20 megabytes per second) and compressing the data stream at a compression ratio

(e.g., 4:1) that provides a compression rate (e.g., 40 megabytes per second) that is greater than the data storage rate. *See id.* at 2:63-3:3.

46. Accordingly, a person of ordinary skill in the art would understand that the broadest reasonable construction of the term “said compression and storage occurs faster than said data stream is able to be stored on said memory device in said received form” to mean “wherein the time to compress and store the received data stream is less than the time to store the received data stream without compressing it.”

B. The term “bandwidth”

47. Independent claim 24 of the ’530 patent recites the term “bandwidth.” The proposed construction is *the speed at which data can be transmitted or stored, and is quantified as a certain number of bits per second.*

48. The ’530 specification uses “data [] rate” (speed) synonymously with “bandwidth.” *See, e.g.,* ’530 Patent at 5:25-43, 9:18-10:10 (equating “input bandwidth” with “input data rate,” and “bandwidth of the target storage device” with a “data storage device . . . [that] is capable of storing 20 megabytes per second” (number of bits per second)); Ex. 1101. Also, the ’530 specification notes that “[t]iming and counting [(rate)] enables determination of the bandwidth of the input data stream.” *Id.* at 9:26-27 (emphasis added). This understanding is also consistent with the dictionary definition of bandwidth, which is “[t]he amount of

data that can be transmitted in a fixed amount of time . . . [and] is usually expressed in bits per second (bps) or bytes per second.” Random House Webster’s, Computer & Internet Dictionary at 45 (3d ed. 1999); Ex. 1114.

49. Accordingly, a person of ordinary skill in the art would understand the broadest reasonable construction of the term “bandwidth” to mean “the speed at which data can be transmitted or stored, and is quantified as a certain number of bits per second.”

VII. PRIOR ART REFERENCES

A. Kawashima

50. U.S. Patent No. 5,805,932 to Kawashima (“Kawashima”, Ex. 1103), which was filed on February 13, 1996, and issued on September 8, 1998. I have been informed that Kawashima is prior art under 35 U.S.C. § 102(a) and (e). I understand that Kawashima was among hundreds of references cited during the original prosecution of the ’530 patent, but it was never discussed during original prosecution or reexamination of the ’530 patent.

51. Kawashima discloses a data compression and transmission/storage system that maximizes the effective bandwidths of the system’s transmissions channels and storage devices while maximizing the system’s storage capacity. Figures 1, 3, and 8 show basic components of the Kawashima system, and will be referred to throughout this declaration.

FIG. 1

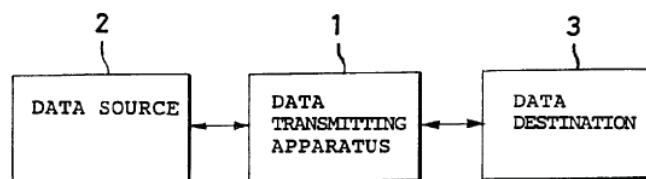


FIG. 3

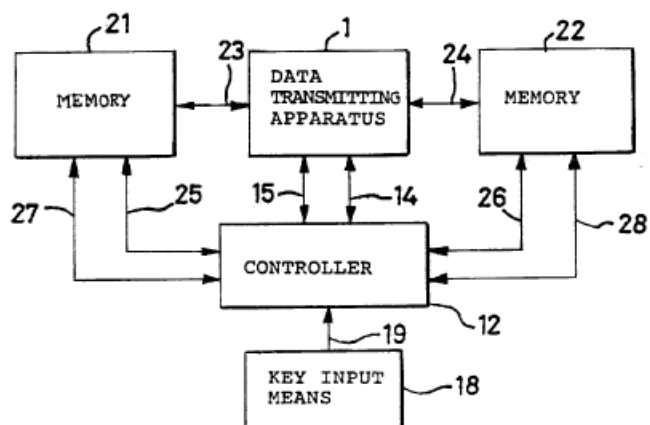
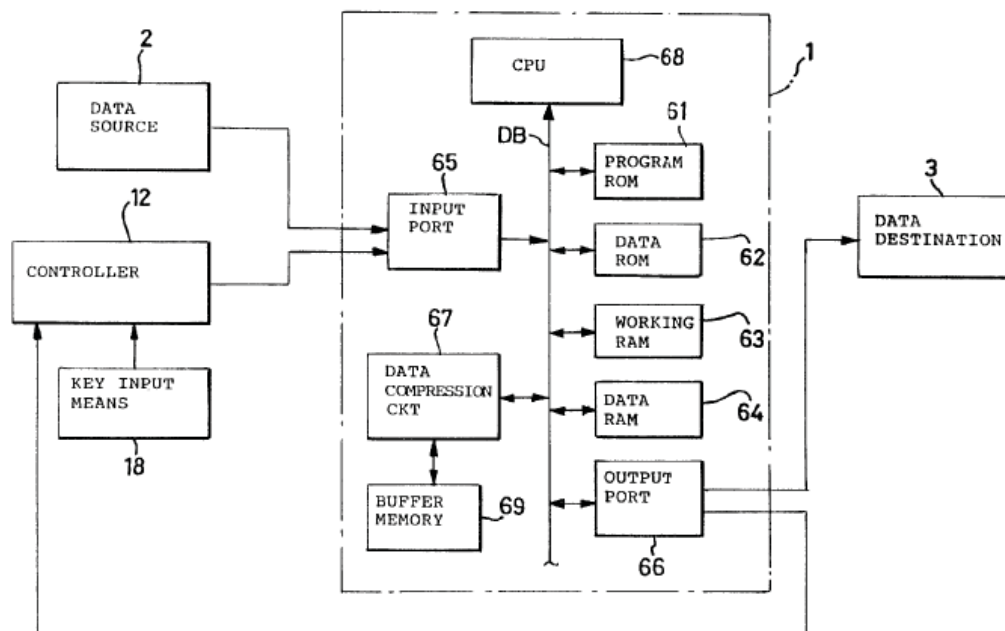


FIG. 8



52. Kawashima discloses a data transmission/storage system that analyzes the compressibility of data before transmission/storage. Kawashima at Abstract; Ex. 1103. Figures 1 and 8 above show a “data transmitting apparatus 1” that is connected to “data source 2” and “data destination 3.” *Id.* at 21:15-18 (Figure 1), 25:8-22 (Figure 8). “Data source 2” and “data destination 3” each may comprise a “communication device” or an “external memory device having an internal storage medium[.]” *Id.* at 21:27-34. Similarly, Figure 3 shows a “data transmitting apparatus 1” that is connected to “memories 21, 22” through “data buses 23, 24.” *Id.* at 22:16-18.

53. Kawashima explains that the “data processing apparatus 1” in Figure 8, which is analogous to the “data transmitting apparatus 1” in Figure 1, includes, for example, an “input port 65 and an output port 66 for inputting data from and outputting data to external circuits,” a “data RAM 64 for storing data from the data source 2,” a “data compression circuit 67 for compressing data stored in the data RAM 64 and storing the compressed data in a given area in the data RAM 64” for further transmission to “data destination 3,” and a CPU (control and logic/arithmetic unit) 68 for controlling the system’s various circuits. *Id.* at 25:8-22.

54. Kawashima addresses the problem of optimally transmitting/storing large amounts of data quickly. *See, e.g., id.* at 1:61-67, 2:58-3:32. Kawashima

observes that the “advent of multimedia information processing”—for example, the transmission of audio and video data in real time—has increased the amount of storage space and memory needed to store and run software programs at a greater rate than the increase in capacity of available storage and memory devices. *Id.* at 1:61-66. This, Kawashima notes, makes “data compression technology”—which, if effective, can reduce the size of data without eliminating any of its content—“more and more important” to data transmission/storage. *Id.* at 1:66-67.

55. Accordingly, Kawashima teaches a compression and transmission/storage system that is designed to compress data and transmit/store the compressed data more quickly than it can transmit/store the data in uncompressed form, as explained in detail further below. Specifically, the Kawashima system achieves fast data compression and transmission/storage by compressing an incoming data stream only if that data stream can be compressed at a relatively high compression ratio that is preset by the system or the system’s user.

Id. at 7:1-15. Figures 43 and 44² below illustrate the effect of compression ratios on the speed of data compression and transmission/storage.³

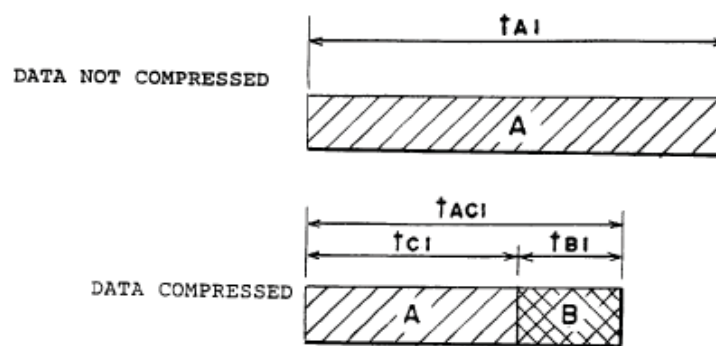
56. As shown in Figure 43 below, if a data stream is compressed at a relatively high compression ratio, then compression will result in a significantly smaller-sized data stream so that compression and transmission/storage will occur significantly more quickly than transmission/storage of the uncompressed data stream. That is, the time it takes to compress at this relatively high ratio will be more than offset by the significant reduction in the amount of time needed to

² Kawashima refers to t_{A1} and t_{C1} in Figure 43 (*id.* at 3:8-15) and t_{A2} and t_{C2} in Figure 44 as the time required for “transmitting” data. But Kawashima treats data “transmission” as covering not only the actual transportation of data, but also the accessing (reading) and storing (writing) of data. *See, e.g., id.* at 22:10-11 (“data is transmitted, i.e., data is read and written”), 21:40-41(same); *see also id.* at 22:55-60, 23:32-37, 24:9-13, 21:51-57.

³ A “compression ratio” is the ratio of the size of data in uncompressed form to the size of that data in compressed form. The size of data is commonly referred to in the art as its “length,” and is quantified by the number of bytes in a given set of data. The more the data is compressed, the smaller number of bytes of the data in compressed form, and the higher the compression ratio.

transmit/store the smaller-sized compressed data stream. Accordingly, the time needed to compress data (which Kawashima calls t_{B1}) and transmit/store the compressed data (which Kawashima calls t_{C1}) (the total time of which Kawashima calls t_{AC1}) will be less than the time needed to transmit/store the same data in uncompressed form (which Kawashima calls t_{A1}). *See id.* at 3:8-23.

FIG. 43

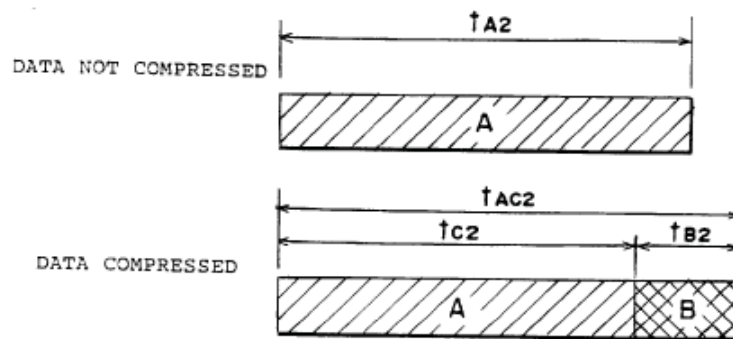


57. That is, referring to the lower portion of Figure 43, Kawashima expressly accounts for the time it takes to compress data (as shown in the double hashed area marked “B” of length t_{B1}) in addition to the time it takes to transmit/store the compressed data (the region labeled “A” of length t_{A1}).

58. Conversely, as shown in Figure 44 below, if a data stream is compressed at a relatively low compression ratio, compression will result in an insignificantly smaller-sized data stream that will be transmitted/stored in roughly the same amount of time as an uncompressed data stream. Therefore, the time it takes to compress at this relatively low ratio will not be offset by the insignificant

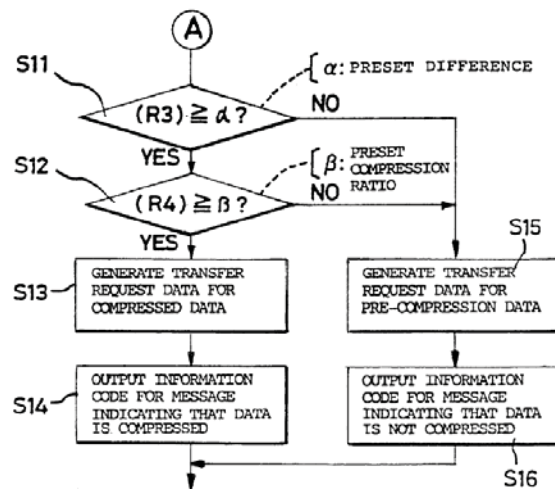
reduction in the amount of time needed to transmit/store the insignificantly smaller-sized compressed data stream. *See id.* at 3:24-32.

FIG. 44



59. The Kawashima system is therefore designed with the objective of compressing and transmitting/storing a compressed data stream only if it determines that the incoming data stream can be compressed at a relatively high ratio, which Kawashima refers to as a “preset ratio,” “preset compression ratio,” or “preset condition.” As shown in Figure 11, the data is compressed and transmitted/stored only if the actual savings from compression (“actual difference”) exceeds the threshold (“preset difference”) and the compression ratio (“actual compression ratio”) exceeds the threshold (“preset compression ratio”). *Id.* at 28:64-29:47.

FIG. 11



60. In particular, Kawashima teaches compressing a portion of a received data stream to calculate a predicted compression ratio of the entire stream, and then compressing the remaining portion of the received data stream and transmitting/storing the stream in compressed form if the predicted compression ratio is equal to or greater than a preset value. *See, e.g., id.* at 36:43-47, 38:50-59.

61. Kawashima recognizes that any lossless technique (e.g., “the known Huffman coding technique or LZW (Lempel-Ziv and Welch) coding technique”) or lossy technique can be used for compression. *See id.* at 3:65-4:6; *see also id.* at 25:23-30 (noting that “[t]he data compression circuit 67 is a circuit for compressing supplied data in a loss-less manner . . . [T]he loss-less data compression technique itself used by the data compression circuit 67 has no direct bearing on the present invention . . . [I]t may be the LZW compressing technique.”). However, Kawashima also recognizes that compressing different

data types with “different attributes[,] including bit map data and an execution file in addition to text data,” can yield different compression ratios. *See id.* at 36:6-20. For example, Kawashima recognizes that the expected compression ratio from the use of a compression technique for text data would yield “only a small error with respect to an actual compression ratio” for a file having text data. *See id.* at 36:6-14. However, the same compression technique would yield “a large error if the same calculations” if carried out for bit map data and execution files. *See id.* at 36:14-20.

B. Sebastian

62. U.S. Patent No. 6,253,264 to Sebastian (“Sebastian”, Ex. 1104), which was filed on March 6, 1998, claims priority to U.S. Provisional Application No. 60/036,548 (filed on March 7, 1997), and issued on June 26, 2001. I have been informed that Sebastian is prior art under 35 U.S.C. § 102(e). I understand that Sebastian was among hundreds of references cited during prosecution of the ’530 patent, but it was never discussed during original prosecution or reexamination of the ’530 patent.

63. Sebastian discloses systems and methods for optimally compressing multiple data blocks using different compression techniques (e.g., Lempel-Ziv, Huffman, and arithmetic coding) on different data blocks based on the format or data type of the data in the data block. *See Sebastian* at Abstract, 19:31-48; Ex.

1104. Sebastian's preferred compression system uses an architecture called a "Base-Filter-Resource" (BFR) system, which integrates the advantages of format or data type-specific compression into a "general-purpose compression tool" designed to serve a wide range of data types. *See id.* at 1:45-50 (referring to data "type" as a data "format").

64. The Sebastian system includes different "filters" (encoders) that each support a specific "data format" (data type), such as for Excel XLS worksheets or Word DOC files. *Id.* at 1:50-51. If an installed filter "matches the format of the data to be encoded, the advantages of format-specific compression can be realized for that data." *Id.* at 1:55-57. If an installed filter does not match up with the format of data to be encoded, a "generic" (default) filter is used which "achieves performance similar to other non-specific data compression systems (such as PKZip, Stacker, etc.)." *Id.* at 1:58-60; *see also id.* at 4:9-23 (other suitable generic filters include those with compression performance similar to generic compression systems that employ Lempel-Ziv (LZ) engines or a SZIP engine). Accordingly, by using data type-specific encoders in combination with default encoders, Sebastian achieves the advantage of being able to handle a "wide range of data formats," including those that may not be recognized by the compression system. *See id.* at 1:45-60.

65. Sebastian is concerned not only with optimizing compression performance and ratios, but also optimizing the broader data processing, compression, and transmission/storage system as a whole. Specifically, Sebastian teaches that selection of the “optimum” compression algorithm should take into account the “desired trade-off” between “compression performance” and the “speed” of the overarching data processing system. *Id.* at 17:49-52. For example, Sebastian explains that “[m]aximum compression can be achieved by encoding [compressing] the full data block using each possible combination of algorithms, but this requires more processing time than most real-world applications can spare.” *Id.* at 21:32-35.

66. But Sebastian also recognizes that better compression performance and greater compression ratios can help optimize overall system performance. For example, Sebastian teaches that data compression can help optimize limited bandwidth transmissions channels—i.e., because more data can be transmitted through channels with limited bandwidth if the data is compressed. *Id.* at 3:31-36 (“The output of the encoder is encoded [compressed] data 4, which can be . . . transmitted to the decoder 5 (in which case the encoding the source data [is] to be transmitted faster when the transmission channel bandwidth is limited).”).

67. Additionally, Sebastian teaches methods for simultaneously optimizing data analysis, data compression, and overall system performance. For

example, Sebastian teaches an optimal pre-compression identification procedure that uses a “search tree method to reduce the time needed to check the list” of available compression techniques, algorithms, or encoders when many such encoders are installed in a data processing and compression system. *Id.* at 5:26-28. Finally, Sebastian teaches that the transmissions/storage systems connected to but not part of the compression system can adjust compression and decompression operations in order to accommodate changes in the overall systems including “available transmission channel bandwidth.” *Id.* at 3:51-55.

C. Chu

68. U.S. Patent No. 5,467,087 to Chu (“Chu”, Ex. 1105), which was filed on December 18, 1992, and issued on November 14, 1995. I have been informed that Chu is prior art under 35 U.S.C. § 102(a), (b), and (e). I understand that Chu was among hundreds of references cited during prosecution of the ’530 patent, but it was never discussed during original prosecution or reexamination.

69. Chu, as its title indicates, teaches “high speed lossless data compression.” Chu at Title; Ex. 1105. Like Sebastian, Chu discloses systems and methods for optimally compressing an input data stream using different compression techniques (e.g., Lempel-Ziv, Huffman, and arithmetic coding) for the data based on the different data types of the data in the data stream. *Id.* at Abstract, Figures 3 and 6, 4:24-65. Chu teaches compressing data in this manner

in order to “maximiz[e] the compression ratio for that input data stream.” *Id.* at Abstract.

70. Chu discloses:

In an alternative embodiment of input data compression process 120 also shown in FIG. 6, a compression rate control signal 103 is provided to data compression process 120. Data compression rate adjustment process 130 adjusts the values of l_{max} , p_{max} , or both l_{max} and p_{max} to increase or decrease the compression process speed in response to data compression rate control signal 103. In an alternative embodiment of data compression rate adjustment process 130, adjustment process 130 indicates to LZ-type compression process 122 whether to use LZ-1 or LZ-2 compression in accordance with data compression rate control signal 103 to adjust the compression time for compressing data. Thus, data compression rate adjustment process 130 provides data compression process 100 with the flexibility to adjust the compression speed during data compression. This flexibility provides compression system 100 means to maximize the CPU's idle time to do data compression and means to increase the data compression speed when the CPU is in preparation to begin another process.

71. That is, Chu teaches that flexible and efficient data compression can be achieved by “alter[ing] the rate of [data] compression.” *Id.* at Abstract. Chu teaches that the speed of compression can be adjusted by applying different compression techniques—for example by “switching” from one Lempel-Ziv method (e.g., LZ1 type method) to a different Lempel-Ziv method (e.g., LZ2 type method), as shown in Figure 6 below.

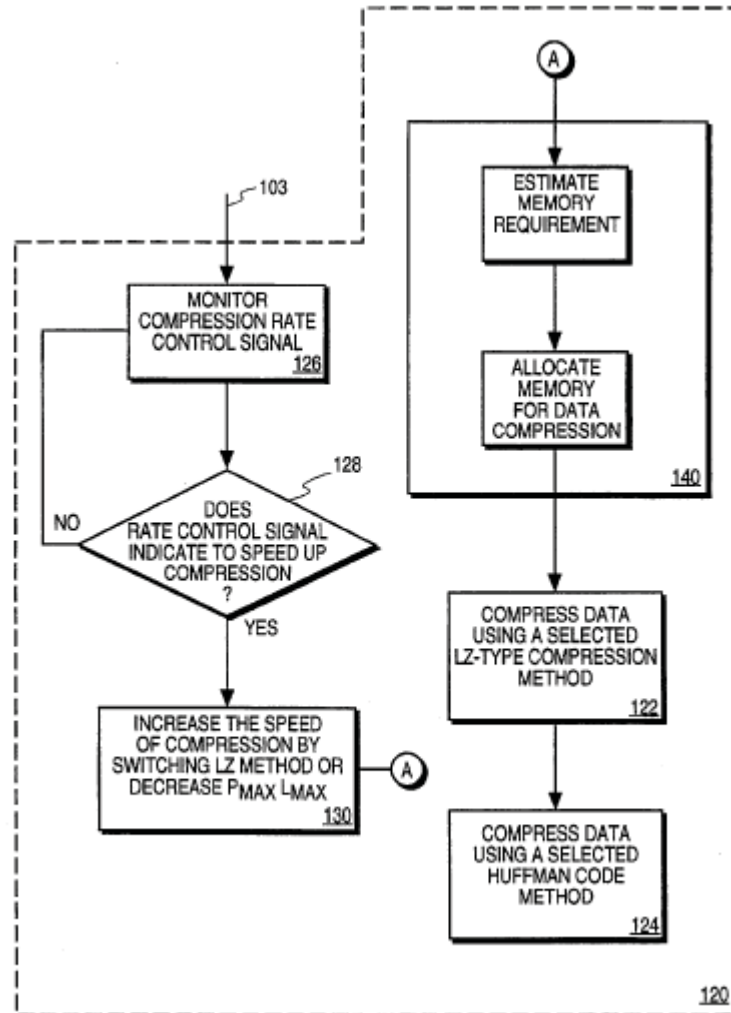


FIG. 6

72. FIG. 6 also discloses adjusting the speed of compression by adjusting pmax and lmax values as disclosed in 130. The pmax and lmax values refer to the size of a history buffer that builds the encoding dictionary database, which corresponds to system parameters for adjusting a history buffer size (e.g., as used in LZ-1 as taught in Chu). *See, e.g., id.* at 2:45-3:22. Lowering the pmax and lmax values creates a smaller history buffer size, and allows for a faster rate of compression (at the trade-off of having a lower compression ratio). *See, e.g., id.* at 2:45-3:22, 5:45-59. Conversely, increasing the pmax and lmax values creates a larger history buffer size, and allows for a higher compression ratio (at the trade-off of a slower rate of compression). *Id.*

VIII. THE CHALLENGED CLAIM IS NOT PATENTABLE

73. The challenged claim recites features well known by persons of ordinary skill in the field of data transmission, compression, and decompression by the time of the alleged invention of the '530 patent. Based on the prior art described below, the claimed system in the challenged claim of the '530 patent is an obvious combination of known functions that perform in expected ways.

IX. IDENTIFICATION OF HOW CLAIM 24 IS UNPATENTABLE

A. Claim 24 is Obvious in View of Kawashima, Sebastian, and Chu

1. Motivation to Combine Kawashima, Sebastian, and Chu

74. It would have been obvious to a person of ordinary skill in the art to combine the teachings of Kawashima, Sebastian and Chu. All three references are in the same technical field of data compression. *See, e.g.*, Kawashima at 1:9-25; Ex. 1103; Sebastian at Abstract, 1:30-60; Ex. 1104; Chu at Abstract; Ex. 1105. All three references teach systems and methods of optimizing not only data compression, but also overall system performance, including maximizing the bandwidths of transmission channels and storage capacity. *See, e.g.*, Kawashima at 2:58-3:32 (teaching optimal interconnected compression and transmission/storage systems); Ex. 1103; Sebastian at 3:31-36, 3:51-55, 5:26-29 (teaching optimal interconnected compression, transmission/storage, and computer processing systems); Ex. 1104; Chu at Abstract, 8:6-11 (teaching “data compression for memory allocation efficiency”); Ex. 1105. Regarding data compression in particular, all three references recognize that certain compression methods compress certain types of data well but not all types of data, and all three references teach systems and methods for compressing a data stream containing different types of data. *See, e.g.*, Kawashima at 36:15-20; Ex. 1103; Sebastian at 1:13-33; Ex. 1104; Chu at Abstract, Figures 3 and 6, 4:24-65; Ex. 1105. All three references also teach the general use of the same types of compression techniques (e.g., Huffman and Lempel-Ziv type methods). *See, e.g.*, Kawashima at 4:1-3; Ex. 1103; Sebastian at 17:15-40; Ex. 1104; Chu at 2:13-44, 4:38-65; Ex. 1105.

Therefore, all three references recognize the significance of analyzing incoming data to determine data type and teach systems and methods of efficiently determining the data type of incoming data (e.g., by reviewing header information associated with the data) in order to best compress such data. *See, e.g.*, Kawashima at 36:6-32; Ex. 1103; Sebastian at Abstract, 4:46-54; Ex. 1104; Chu at Abstract, 2:63-3:8, 4:38-65; Ex. 1105. It would therefore have been obvious to combine the teachings of Kawashima, Sebastian and Chu to solve the common problem of optimally compressing a collection of data blocks containing different types of data, such as, as expressly identified by Kawashima, in an incoming data stream containing “bit map” data and “execution” files. *See* Kawashima at 36:15-20; Ex. 1103. Moreover, Kawashima recognizes that compressing different data types with “different attributes[,] including bit map data and an execution file in addition to text data,” can yield different compression ratios. *See id.* at 36:6-20. Therefore, Kawashima recognizes the efficacy of using different compression techniques for different data types.

75. The Kawashima system is focused on the overall efficiency of the transmission/storage aspects of compressing an incoming data stream whereas Sebastian and Chu are focused on the details of the compression process. *See, e.g.*, Sections VII.A, VII.B, and VII.C above. Kawashima accounts for an incoming data stream with multiple data types during, for example, the process of calculating

predicted compression ratios of an incoming data stream by weighting the average compression ratio of sample portions of that stream with the number of data types detected, to account for the fact that not all of this data will be compressed as effectively with the single compression technique applied by the Kawashima system as described in Kawashima. *See* Kawashima at 36:6-32; Ex. 1103. It would have been obvious for a person of ordinary skill in the art concerned with the common and obvious problem of optimally compressing a data stream containing different types of data to modify the Kawashima system to incorporate Sebastian's and Chu's teachings of using different compression techniques to compress the different data types detected in the data stream, including by using the same types of techniques taught by both references (e.g., Huffman and Lempel-Ziv type methods). Doing so would amount to nothing more than the combination of familiar elements according to known methods to yield predictable results.

2. Independent Claim 24 is Obvious in View of Kawashima, Sebastian and Chu.

76. All of the limitations recited in claim 24 are obvious over Kawashima in view of Sebastian and Chu. I understand that the limitations of claim 24 have been broken down by Petitioner as follows:

Claim 24
24. A system comprising: [Preamble]

Claim 24	
A	a memory device; and
B	a data accelerator, wherein said data accelerator is coupled to said memory device,
C	a data stream is received by said data accelerator in received form,
D	wherein a bandwidth of the received data stream is determined,
E	said data stream includes a first data block and a second data block,
F	said data stream is compressed by said data accelerator to provide a compressed data stream by compressing said first data block with a first compression technique and said second data block with a second compression technique, said first and second compression techniques are different,
G	wherein a data rate of the compressed data stream is adjusted, by modifying a system parameter, to make a bandwidth of the compressed data stream compatible with a bandwidth of the memory device,
H	said compressed data stream is stored on said memory device,
I	said compression and storage occurs faster than said data stream is able to be stored on said memory device in said received form,
J	a first data descriptor is stored on said memory device indicative of said first compression technique, and
K	said first descriptor is utilized to decompress the portion of said compressed data stream associated with said first data block.

i. The preamble of Claim 24 is disclosed in Kawashima.

77. The preamble of claim 24 of the '530 patent recites: “[a] system.”

78. Kawashima discloses a data compression and transmission/storage system that, as explained in detail further below, maximizes the effective bandwidths of the system's transmissions channels and storage devices while maximizing the system's storage capacity. Figures 1, 3, and 8, as illustrated in Section VII.A above, show basic components of the Kawashima system.

79. Figures 1 and 8 show a "data transmitting apparatus 1" that is connected to "data source 2" and "data destination 3." Kawashima at 21:15-18 (Figure 1), 25:8-22 (Figure 8); Ex. 1103. "Data source 2" and "data destination 3" each may comprise a "communication device" or an "external memory device having an internal storage medium[.]" *Id.* at 21:27-34. Similarly, Figure 3 shows a "data transmitting apparatus 1" that is connected to "memories 21, 22" through "data buses 23, 24." *Id.* at 22:16-18.

80. Kawashima explains that the "data **processing** apparatus 1" in Figure 8, which is analogous to the "data **transmitting** apparatus 1" in Figure 1, includes, for example, an "input port 65 and an output port 66 for inputting data from and outputting data to external circuits," a "data RAM 64 for storing data from the data source 2," a "data compression circuit 67 for compressing data stored in the data RAM 64 and storing the compressed data in a given area in the data RAM 64" for further transmission to "data destination 3," and a CPU (control and

logic/arithmetic unit) 68 for controlling the system's various circuits. *Id.* at 25:8-22 (emphasis added).

ii. Limitation A of Claim 24 is disclosed in Kawashima.

81. Limitation A of claim 24 of the '530 patent recites: "a memory device." This limitation is found in Kawashima.

82. Kawashima discloses a data compression and transmission/storage system that includes a memory device that stores data compressed by the system, as specifically shown and described in Kawashima with reference to Figures 3 through 8. In Figure 8, the "data compression circuit 67" compresses input data and stores it "in a given area in the data RAM 64" (memory) (*id.* at 25:18-21) for further transmission to an "external memory device" located at "data destination 3" (*id.* at 25:55-57, *see also id.* at 21:27-34 ("data destination 3 may alternatively comprise . . . an external memory device")). In Figure 3, for example, the "data transmitting apparatus 1 either compresses [input] data or does not compress the [input] data, and transmits the [compressed or uncompressed] data to the memory 22 which is the data destination." *Id.* at 22:38-41.

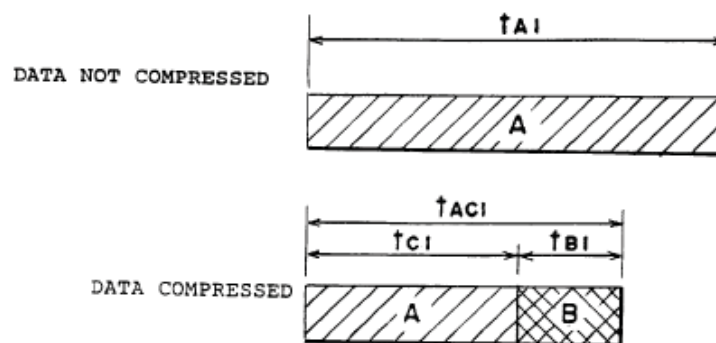
iii. Limitation B of Claim 24 is disclosed in Kawashima.

83. Limitation B of claim 24 of the '530 patent recites: "a data accelerator, wherein said data accelerator is coupled to said memory device." This limitation is found in Kawashima.

84. Kawashima discloses a data compression and transmission/storage system which includes a “data accelerator” that is “coupled” to the memory device described in Section IX.A.2.ii above.

85. In the Kawashima system, data is transmitted/stored faster—i.e., “accelerated”—by compressing it before storing it, as shown in Figure 43 below and explained in more detail in Section IX.A.2.x further below.

FIG. 43



86. Figure 43 illustrates that if data is compressed at a high enough compression ratio, it can be both compressed, transmitted/stored more quickly (t_{AC1}) than it can be transmitted/stored in uncompressed form (t_{A1}). *See also id.* at 3:8-23.

87. Accordingly, the component in the Kawashima system that performs the data compression (or the component wherein the data compression is performed) is the “data accelerator”—i.e., the “data transmitting apparatus 1” in Figures 3 and 8 and, more specifically, the “data compression circuit 67” in data

transmitting apparatus 1 as shown in Figure 8. *Id.* at 22:38-41 (Figure 3), 25:8-22 (Figure 8). These components are “coupled” to the memory device described in Section IX.A.2.ii above. For example, in Figure 8, the “data transmitting apparatus 1” (data accelerator) is “connected” (coupled) to “data destination 3.” *Id.* at 25:8-22 (Figure 8); *see also id.* at 21:15-18 (Figure 1). Kawashima explains that “data destination 3” may comprise an “external memory device having an internal storage medium” (memory). *Id.* at 21:27-34. Similarly, Figure 3 shows a “data transmitting apparatus 1” (data accelerator) that is “connected” (coupled) to “memories 21, 22” (memory). *Id.* at 22:16-18. Figure 8 also shows “data compression circuit 67” connected to “data RAM 64.” *Id.* at 25:8-22.

iv. Limitation C of Claim 24 is disclosed in Kawashima.

88. Limitation C of claim 24 of the ’530 patent recites: “a data stream is received by said data accelerator in received form.” This limitation is found in Kawashima.

89. Kawashima discloses a “data accelerator”—i.e., the “data transmitting apparatus 1” in Figures 3 through 8 and, more specifically, the “data compression circuit 67” in Figure 8. *Id.* at 22:38-41 (Figure 3), 25:8-22 (Figure 8). These components receive a stream of data in received form and transmit/store this data either in received form or in compressed form. *See, e.g., id.* at 7:15-32 (describing means for storing a portion of an incoming stream of data while the system

compresses another “portion” of the stream). For example, the “data transmitting apparatus 1” in Figure 3 receives a data stream in received form from the “memory 21” before transmitting the data (either in received or compressed form) to “memory 22.” *Id.* at 22:36-41. Likewise, the “data processing apparatus 1” in Figure 8 receives a stream of data in received form from the “data source 2” before transmitting the data (either in received or compressed form) to “data destination 3.” *Id.* at 25:8-22, 55-60. Data source 2 includes a “data processing system for processing an inputted analog signal or digital data [(data stream)].” *Id.* at 21:18-26. In Figure 8, the “data compression circuit 67” receives data in received form from “data RAM 64” (which previously received that data in received form from “data source 2”). *Id.* at 25:8-22. The “data compression circuit 67” then stores that data (either in received or compressed form) “in a given area in the data RAM 64.” *Id.*

v. Limitation D of Claim 24 is obvious in view of Kawashima and Sebastian.

90. Limitation D of claim 24 of the ’530 patent recites: “wherein a bandwidth of the received data stream is determined.” This limitation is obvious in view of Kawashima and Sebastian.

91. Sebastian teaches that after the Sebastian system compresses data, that data may be stored in memory or transmitted elsewhere, in which case it would be “transmitted faster when the transmission channel bandwidth is limited.”

Sebastian at 3:31-36; Ex. 1104. Sebastian further teaches that while data storage, retrieval, and transmissions applications may not be part of a compression system, they may adjust certain compression and decompression operations according to “systems conditions such as available transmission channel bandwidth.” *Id.* at 3:51-55.

92. As discussed above (*see. e.g.*, Section IX.A.1), it would have been obvious to combine the teachings of Kawashima and Sebastian. All are in the field of data compression, all are concerned with optimizing not only the compression process but also overall system processes, all recognize the importance of efficiently analyzing data for different data types in order to optimally compress the data, and all are directed towards achieving optimal compression and high compression ratios.

vi. Limitation E of Claim 24 is disclosed in Kawashima.

93. Limitation E of claim 24 of the '530 patent recites: “said data stream includes a first data block and a second data block.” This limitation is found in Kawashima.

94. Kawashima discloses an input data stream, as explained in Section IX.A.2.iv above. Kawashima explains that the data streams the Kawashima system can receive and compress include multiple “files” or multiple “**blocks** of fixed length divided from the files.” Kawashima at 40:32-35 (emphasis added) (“Some

systems compress data by files or blocks of fixed length divided from files. The data transmitting apparatus 1 according to the first and second embodiments are also applicable to these systems.”); Ex. 1103.

vii. Limitation F of Claim 24 is obvious in view of Kawashima and Sebastian, as well as obvious in view of Kawashima and Chu.

95. Limitation F of claim 24 of the ’530 patent recites: “said data stream is compressed by said data accelerator to provide a compressed data stream by compressing said first data block with a first compression technique and said second data block with a second compression technique, said first and second compression techniques are different.” This limitation is obvious in view of Kawashima and Sebastian, as well as obvious in view of Kawashima and Chu.

96. Kawashima explains that the Kawashima system’s data accelerator (for example, the “data transmitting apparatus 1” in Figures 3 through 8 as well as the “data compression circuit 67” in Figure 8) can compress a data stream consisting of multiple “files” or multiple “blocks of fixed length divided from the files.” *See id.* at 40:32-35; *see also id.* at 7:15-32 (data stream). The Kawashima system also uses a “loss-less data compression **process**” (*see, e.g., id.* at 1:9-17, 3:8-15 (emphasis added)) and Kawashima teaches that different lossless compression **techniques** include “the known Huffmann [*sic*] coding **technique**” as well as the “LZW (Lempel-Ziv and Welch) coding **technique**” (*id.* at 4:1-3

(emphasis added)). Kawashima also discloses compressing with a lossless data

compression technique that has already been compressed with some other

unknown, undisclosed, or unidentified compression technique. *Id.* at 2:20-28.

Kawashima, however, intentionally does not specify a preferred number of or

particular selection of different lossless compression techniques to be used by the

Kawashima system. This is because, as Kawashima explains, the “loss-less data

compression technique itself used by the data compression circuit 67 has no direct

bearing on the present invention.” *Id.* at 25:26-30. In other words, the

compression process in the Kawashima system is designed to be used in

conjunction with any number of known lossless compression techniques.

Nevertheless, Kawashima notes that the “LZW compressing technique,” which it

describes as a dictionary-based technique, is among the lossless compression

techniques that can be used. *Id.* at 25:26-46.

97. Sebastian teaches the use of the same types of lossless compression techniques taught by Kawashima—e.g., Huffman and Lempel-Ziv type methods—as well as other lossless techniques such as arithmetic coding. Sebastian at 19:31-48; Ex. 1104. Sebastian further teaches a compression process that uses a plurality of these techniques on an incoming data stream depending on the various types of data in the stream. *See id.* at Abstract, 19:31-48.

98. The Sebastian patent's preferred system includes different compression algorithms/techniques (which Sebastian refers to as "filters") that each support a specific data type or format, such as for Excel XLS worksheets or Word DOC files. *Id.* at 1:50-52. Sebastian discloses that these techniques can be applied on a block-by-block basis. *Id.* at Abstract, 2:33-42, 4:9-15. If the system has an installed technique that supports the data type of the data to be compressed, the system will use that technique to compress the data. *See id.* at 1:55-57. If the system does not have an installed technique that specifically supports the data type of the data to be encoded, a "generic" technique is used, which "achieves performance similar to other non-specific data compression systems (such as PKZip, Stacker, etc.)." *Id.* at 1:58-60; *see also id.* at 4:9-23 (other suitable generic techniques include those with compression performance similar to generic compression systems that employ Lempel-Ziv (LZ) engines or a SZIP engine). Accordingly, by using data type-specific techniques in combination with one or more generic techniques, the Sebastian system achieves the advantage of being able to optimally compress a "wide range" of data types or format. *See id.* at 1:45-60.

99. As discussed above (*see. e.g.*, Section IX.A.1), it would have been obvious to a person of ordinary skill in the art to combine the teachings of Kawashima and Sebastian to solve the common and obvious problem of optimally

compressing a data stream containing different types of data, each type of which can be optimally compressed using different compression techniques. Both therefore recognize the significance of analyzing incoming data to determine data type and teach systems and methods of efficiently determining the data type of incoming data (by reviewing header information associated with the data) in order to optimally compress such data. *See, e.g.,* Kawashima at 36:6-32; Ex. 1103; Sebastian at Abstract, 2:33-42, 4:46-54; Ex. 1104. Kawashima is concerned with maximizing the bandwidths of data transmissions/storage systems by compressing data at a high ratio. *See* Kawashima at 3:8-32; Ex. 1103. Kawashima suggests that any lossless encoding technique can be used to achieve sufficiently high compression ratios and even teaches the general use of the same types of compression techniques taught by Sebastian (e.g., Huffman and Lempel-Ziv type methods). *See* Kawashima at 4:1-3, 25:26-30; Ex. 1103; Sebastian at 17:15-40; Ex. 1104. It would therefore have been obvious for a person of ordinary skill in the art to modify the system taught in Kawashima to incorporate Sebastian's teachings of using a combination of different compression techniques (e, g., Huffman and Lempel-Ziv type methods) to optimally compress different types of data in an incoming data stream. Doing so would accomplish Kawashima's fundamental objective of compressing data at a high ratio in order to compress data and transmit/store the compressed data more quickly.

100. Chu describes a “compression rate adjustment process 130” indicating to an “LZ-type compression process 122” whether “to use LZ-1 or LZ-2 compression in accordance with data compression rate control signal 103 to adjust the compression time for compressing data.” *Id.* at 6:54-59. That is, Chu describes switching between LZ-1 and LZ-2 compression in response to a rate control signal within the same data stream. Accordingly, Chu teaches that different compression techniques may be applied to different blocks of data.

101. One of ordinary skill in the art would have been motivated to combine the teachings of Kawashima and Chu to solve the common and obvious problem of optimally compressing a data stream containing different types of data, each type of which can be optimally compressed using different compression techniques. Both references teach systems and methods of optimizing not only data compression, but also overall system performance, including maximizing the bandwidths of transmission channels and storage capacity. *See, e.g.*, Kawashima at 2:58-3:32 (teaching optimal interconnected compression and transmission/storage systems); Ex. 1103; Chu at Abstract, 8:6-11 (teaching “data compression for memory allocation efficiency”); Ex. 1105. Regarding data compression in particular, both references recognize that certain compression methods compress certain types of data well, and both references teach systems and methods for compressing a data stream containing different types of data. *See,*

e.g., Kawashima at 36:15-20; Ex. 1103; Chu at Abstract, Figures 3 and 6, 4:24-65;

Ex. 1105. Both references also teach the general use of the same types of compression techniques (*e.g.*, Huffman and Lempel-Ziv type methods). *See, e.g.*, Kawashima at 4:1-3; Ex. 1103; Chu at 2:13-44, 4:38-65; Ex. 1105. Therefore, both references recognize the significance of analyzing incoming data to determine data type and teach systems and methods of efficiently determining the data type of incoming data in order to best compress such data. *See, e.g.*, Kawashima at 36:6-32; Ex. 1103; Chu at Abstract, 2:57-3:8, 4:38-65; Ex. 1105. It would therefore have been obvious to combine the teachings of Kawashima and Chu to solve the common problem of optimally compressing a collection of data blocks containing different types of data, such as, as expressly identified by Kawashima, in an incoming data stream containing “bit map” data and “execution” files. *See* Kawashima at 36:15-20; Ex. 1103. Moreover, Kawashima recognizes that compressing different data types with “different attributes[,] including bit map data and an execution file in addition to text data,” can yield different compression ratios. *See id.* at 36:6-20. Therefore, Kawashima recognizes the efficacy of using different compression techniques for different data types.

viii. Limitation G of Claim 24 is obvious in view of Kawashima, Sebastian, and Chu.

102. Limitation G of claim 24 of the '530 patent recites: “wherein a data rate of the compressed data stream is adjusted, by modifying a system parameter,

to make a bandwidth of the compressed data stream compatible with a bandwidth of the memory device.” This limitation is obvious in view of Kawashima, Sebastian, and Chu.

103. Sebastian teaches that compressed data can be “transmitted faster when the transmission channel bandwidth is limited” than uncompressed data. Sebastian at 3:31-36; Ex. 1104. Sebastian teaches that while data storage, retrieval, and transmissions applications may not be part of a compression system, they may adjust certain compression and decompression operations (system parameters) according to “systems conditions such as available transmission channel bandwidth.” *Id.* at 3:51-55. A person of ordinary skill in the art would understand that data storage, retrieval, and transmissions applications may adjust compression and decompression system parameters according to other “systems conditions” and “bandwidths” such as the bandwidth of the memory device.

104. Chu teaches that flexible and efficient data compression can be achieved by “alter[ing] the rate of compression.” Chu at Abstract; Ex. 1105. Chu teaches that the speed of compression can be adjusted by applying different compression techniques—for example by “switching” to a different Lempel-Ziv technique, as shown in Figure 6 below.

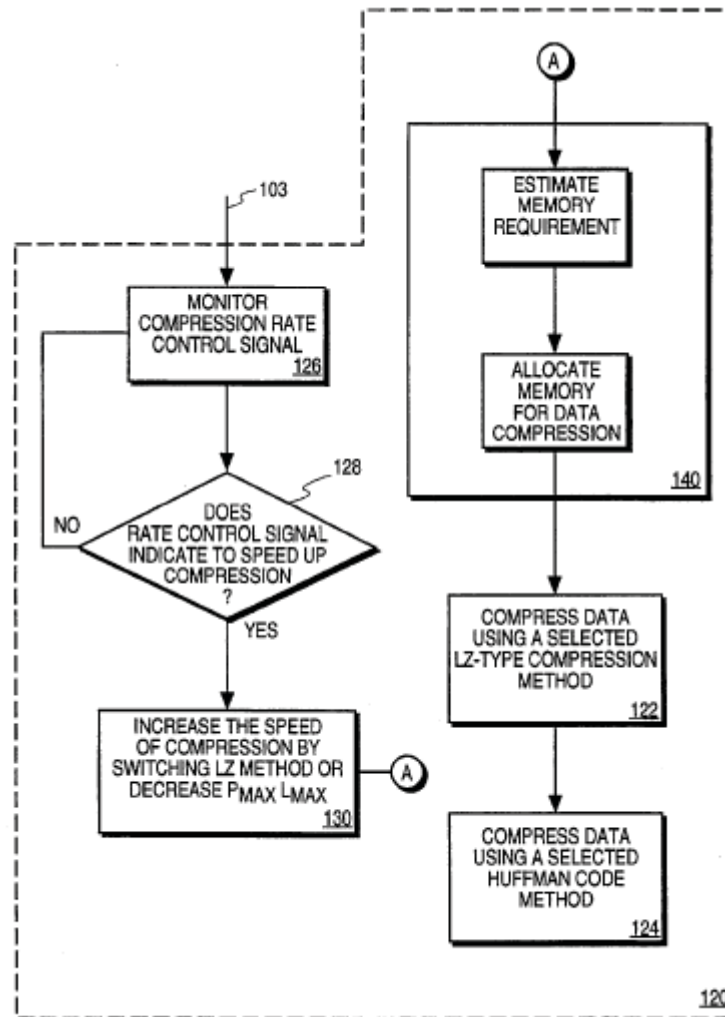


FIG. 6

105. As illustrated in Figure 6 above, Chu teaches that a “system memory allocation process 140 is provided to allow the system or the user control of the amount of system memory to be allocated for data compression.” *Id.* at 6:15-19.

106. In one embodiment, the “compression rate adjustment process 130 . . . indicates to LZ-type compression process 122 whether to use LZ-1 or LZ-2 compression in accordance with data compression rate control signal 103 to adjust

the compression time for compressing data.” *Id.* at 6:54-59. In addition to being taught in Chu, one of ordinary skill in the art would understand that different methods of the Lempel-Ziv family, such as a parameter indicating whether LZ-1 or LZ-2 compression methods are employed, allow for different trade-offs between the rate of compression and the compression ratio. “Rate control” is a standard term of the art, and is understood by one of skill in the art to refer to the issue of making two bandwidths compatible.

107. In another embodiment, Chu teaches that the memory allocation process “estimates the memory requirement in accordance to the identified input data type, a selected compression ratio, a selected speed of data compression, a selected pmax and lmax value, or a selected combination of these features.” *Id.* at 6:23-28. The pmax and lmax values refer to the size of a history buffer that builds the encoding dictionary database, which corresponds to system parameters for adjusting a history buffer size (e.g., as used in LZ-1 as taught in Chu). *See, e.g., id.* at 2:45-3:22. Lowering the pmax and lmax values creates a smaller history buffer size, and allows for a faster rate of compression (at the trade-off of having a lower compression ratio). Conversely, increasing the pmax and lmax values creates a larger history buffer size, and allows for a higher compression ratio (at the trade-off of a slower rate of compression). *Id.; see also id.* at 5:45-59.

108. Accordingly, Chu teaches modifying a system parameter (e.g., by either switching between LZ-1 and LZ-2 or adjusting the pmax and lmax values) to adjust the data rate of the compressed stream. Chu teaches that this modification can be made to adjust compression speed and performance to account for rate control considerations and to account for changes in the availability of system memory to process this compression. It would have been obvious to a person of ordinary skill in the art to practice this teaching in Chu to adjust compression speed and performance to account for restrictions or limitations in other types of memory—for example, to account for the differences in the bandwidth of the compressed data stream and the bandwidth of the memory device to which the compressed data is to be stored. Chu therefore discloses this limitation.

109. As discussed above (*see. e.g.*, Section IX.A.1), it would have been obvious to combine the teachings of Kawashima, Sebastian, and Chu. All are in the field of data compression, all are concerned with optimizing not only the compression process but also overall system processes, all recognize the importance of efficiently analyzing data for different data types in order to optimally compress the data, and all are directed towards achieving optimal compression and high compression ratios. *See, e.g.*, Sections VII.A, VII.B, VII.C above.

ix. Limitation H of Claim 24 is disclosed in Kawashima.

110. Limitation H of claim 24 of the '530 patent recites: “said compressed data stream is stored on said memory device.” This limitation is found in Kawashima.

111. Kawashima discloses that after the Kawashima system's data accelerator compresses data, that compressed data is stored on a memory device. As explained in Section IX.A.2.iii above, the Kawashima system's “data accelerator” (i.e., the “data transmitting apparatus 1” (Figure 3), “data processing apparatus 1” (Figure 8), and “data compression circuit 67” (Figure 8), Kawashima at 22:38-41 (Figure 3), 25:8-22 (Figure 8); Ex. 1103) is coupled to the system's memory device (i.e., the “memory . . . 22” (Figure 3), “data destination 3” (Figure 8), and “data RAM 64” (Figure), *id.* at 22:16-24, 25:8-22). Kawashima discloses that the memory device stores data compressed by the data accelerator. *See, e.g., id.* at 22:38-41 (“The data transmitting apparatus 1 either compresses the transferred data or does not compress the transferred data, and transmits the data to the memory 22 which is the data destination.” (Figure 3)), 25:18-22 (“a “data compression circuit 67 for compressing data stored in the data RAM 64 and storing the compressed data in a given area in the data RAM 64” (Figure 8)), 22:55-57 (“Data (compressed data or uncompressed data) stored in the data RAM 64 and destined for the data destination 3 is supplied through the output port 66 to the data destination 3 (Figure 8)).

x. Limitation I of Claim 24 is disclosed in Kawashima.

112. Limitation I of claim 24 of the '530 patent recites: "said compression and storage occurs faster than said data stream is able to be stored on said memory device in said received form." This limitation is found in Kawashima.

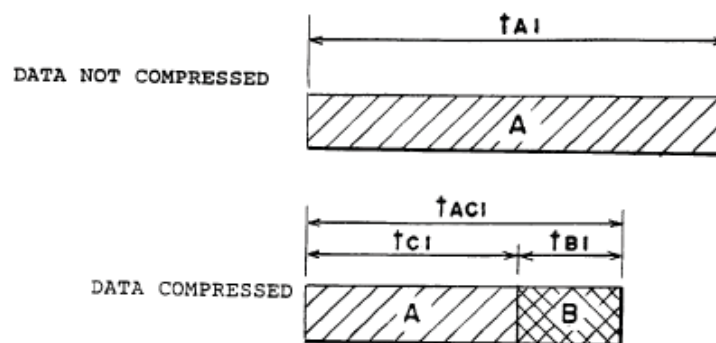
113. Kawashima discloses a compression and transmission/storage system that is designed to compress data and transmit/store the compressed data more quickly than it can transmit/store the data in uncompressed form, as explained in detail below.

114. The Kawashima system achieves fast data compression and transmission/storage by transmitting/storing the compressed version of an incoming data stream when that data stream can be compressed at a relatively high compression ratio that is preset by the system or the system's user. *Id.* at 7:1-15. Figures 43 and 44 below illustrate the effect of compression ratios on the speed of data compression and transmission/storage.

115. As shown in Figure 43 below, and also explained in Section VII.A above, if a data stream is compressed at a relatively high compression ratio, then compression will result in a significantly smaller-sized data stream so that compression and transmission/storage will occur significantly more quickly than transmission/storage of the uncompressed data stream. That is, the time it takes to compress at this relatively high ratio will be more than offset by the significant

reduction in the amount of time needed to transmit/store the smaller-sized compressed data stream. Accordingly, the time needed to compress data (which Kawashima calls t_{B1}) and transmit/store the compressed data (which Kawashima calls t_{C1}) (the total time of which Kawashima calls t_{AC1}) will be less than the time needed to transmit/store the same data in uncompressed form (which Kawashima calls t_{A1}). *See id.* at 3:8-23.

FIG. 43



116. That is, referring to the lower portion of Figure 43, Kawashima expressly accounts for the time it takes to compress data (as shown in the double-hashed area marked “B” of length t_{B1}) in addition to the time it takes to transmit/store the compressed data (the region labeled “A” of length t_{A1}).

117. The Kawashima system is therefore designed with the objective of compressing and transmitting/storing a compressed data stream when it determines that the incoming data stream can be compressed at a relatively high ratio.

Accordingly, the Kawashima system compresses and stores a compressed data

stream faster than said data stream is able to be stored on said memory device in said received form.

xi. Limitation J of Claim 24 is obvious in view of Kawashima and Sebastian.

118. Limitation J of claim 24 of the '530 patent recites: “a first data descriptor is stored on said memory device indicative of said first compression technique.” This limitation is obvious in view of Kawashima and Sebastian.

119. Sebastian discloses that after the appropriate compression technique is applied to compress a data block, the “encoded” (compressed) “data stream” includes an “identification (ID) code” (descriptor) indicating which “filter” (compression technique) was used to “encode” (compress) the data. *See* Sebastian at 5:15-18; Ex. 1104. Sebastian also discloses that “the output of the encoder” (the encoded data and the identification (ID) code) “can be stored in memory.” *Id.* at 3:31-32.

120. As discussed above (*see, e.g.*, Section IX.A.1), it would have been obvious to a person of ordinary skill in the art to combine the teachings of Kawashima and Sebastian. Both recognize the significance of analyzing incoming data to determine data type and teach systems and methods of efficiently determining the data type of incoming data (by reviewing header information associated with the data) in order to optimally compress such data. *See, e.g.*, Kawashima at 36:6-32; Ex. 1103; Sebastian at Abstract, 4:46-54; Ex. 1104. Both

teach the general use of the same types of compression techniques (e.g., Huffman and Lempel-Ziv type methods). *See, e.g.*, Kawashima at 4:1-3; Ex. 1103; Sebastian at 17:15-40; Ex. 1104. It would therefore have been obvious for a person of ordinary skill in the art to modify the system taught in Kawashima to incorporate Sebastian's teachings of using a plurality of compression techniques to optimally compress different types of data in an incoming data stream and subsequently generate and store descriptors indicating which techniques were used. The descriptors are then used during decompression.

121. Moreover, the '530 patent itself recognizes that one of ordinary skill in the art would know how to extract from a compressed data block a descriptor indicative of the compression algorithm applied. *See* '530 Patent at 14:5-10 (describing a "descriptor extraction module 60" that receives an "input data block and then parses, lexically, syntactically, or otherwise **analyzes the input data block using methods known by those skilled in the art to extract the data compression type descriptor associated with the data block.**" (emphasis added)). Therefore, it would not only have been obvious, but completely expected for a compression system to generate, store, and even append to a compressed data block a descriptor indicative of the compression algorithm applied, for use during decompression.

xii. Limitation K of Claim 24 is obvious in view of Kawashima and Sebastian.

122. Limitation K of claim 24 of the '530 patent recites: “said first descriptor is utilized to decompress the portion of said compressed data stream associated with said first data block.” This limitation is obvious in view of Kawashima and Sebastian.

123. Sebastian discloses that after the appropriate compression technique is applied to compress a data block, the “encoded” (compressed) “data stream” includes an “identification (ID) code” (descriptor) indicating which “filter” (compression technique) was used to “encode” (compress) the data, and the “decoder has to have a corresponding decoding filter.” *See id.* at 5:15-18.

Sebastian further discloses that during decompression, the “decoder” retrieves the encoded data from memory and reverses the compression techniques used to encode the data. *Id.* at 3:37-39; *see also id.* at 3:31-32 (encoded data stored in memory). A person of ordinary skill in the art would have understood that in order to efficiently reverse the compression techniques during decompression, the decoder would refer to the identification (ID) code indicating which techniques were originally applied during compression.

124. As discussed above (*see. e.g.*, Sections IX.A.1 and IX.A.2.xi), it would have been obvious to a person of ordinary skill in the art to combine the teachings of Kawashima and Sebastian. Both recognize the significance of analyzing incoming data to determine data type and teach systems and methods of

efficiently determining the data type of incoming data (by reviewing header information associated with the data) in order to optimally compress such data. *See, e.g.*, Kawashima at 36:6-32; Ex. 1103; Sebastian at Abstract, 4:46-54; Ex. 1104. Both teach the general use of the same types of compression techniques (e.g., Huffman and Lempel-Ziv type methods). *See, e.g.*, Kawashima at 4:1-3; Ex. 1103; Sebastian at 17:15-40; Ex. 1104. It would therefore have been obvious for a person of ordinary skill in the art to modify the system taught in Kawashima to incorporate Sebastian's teachings of using a plurality of compression techniques to optimally compress different types of data in an incoming data stream and subsequently generate, store, and use during decompression descriptors indicating which compression techniques were used.

125. Moreover, the '530 patent itself recognizes that one of ordinary skill in the art would know how to extract from a compressed data block a descriptor indicative of the compression algorithm applied. *See* '530 Patent at 14:5-10 (describing a "descriptor extraction module 60" that receives an "input data block and then parses, lexically, syntactically, or otherwise **analyzes the input data block using methods known by those skilled in the art to extract the data compression type descriptor associated with the data block.**" (emphasis added)). Therefore, it would not only have been obvious, but completely expected for a compression system to generate and store a descriptor indicative of the

IPR2016-00376: Storer Declaration

U.S. Patent No. 7,415,530, Claim 24

compression algorithm applied, for use during decompression, and to use that descriptor during decompression.

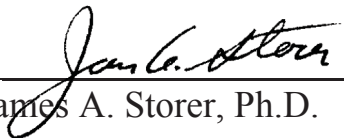
X. Conclusion

126. Based on the foregoing, claim 24 of the '530 patent recite subject matter that is obvious.

Jurat

127. I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1101 of Title 18 of the United States Code.

Date: December 23, 2015



James A. Storer, Ph.D.

APPENDIX A

James A. Storer

Professor of Computer Science

Computer Science Department
Brandeis University
Waltham, MA 02454
phone: 781-736-2714
email: storer@cs.brandeis.edu

Education:

Ph.D. Princeton University, computer science (1979)
M.A. Princeton University, computer science (1977)
B.A. Cornell University, mathematics and computer science (1975)

Employment History:

Brandeis University, Professor of Computer Science (1993 - present)
Brandeis University, Associate Professor of Computer Science (1986 - 1992)
Harvard University, Visiting Professor of Computer Science (1987 - 1988)
Brandeis University, Assistant Professor of Computer Science (1981 - 1986)
Bell Laboratories at Murray Hill, MTS (1979 - 1981)

Research Interests:

Computer algorithms, image retrieval, data compression and archiving (including text, images, video, and multi-media), data similarity measures, pattern matching, data communications, processing of large data sets (including hyperspectral imagery), text, image, and video processing, parallel computing, machine learning.

Professional Activities:

- In 1991 I founded the Annual IEEE Data Compression Conference (DCC), the first major international conference devoted entirely to data compression, and have served as the conference chair since then.
- I am a member of the ACM and IEEE Computer Society. I routinely serve as referee for papers submitted to journals (JACM, SICOMP, Theoretical CS, J. Algorithms, Algorithms, *Signal Processing*, JPDC, Acta Inf., Algorithmica, IPL, IPM, Theoretical CS, J. Alg., Networks, IEEE J. Rob. & Aut., IEEE Trans. Inf. Theory, IEEE Trans. Comp., IEEE Trans. Image Proc., Proceedings of the IEEE, IBM J. of R&D, JCSS, etc.). I have served as an editor for *Information Processing and Management*, *Journal of Visual Communication and Image Representation*, and the *Proceedings of the IEEE*. I have served as a program committee member for various conferences, including *IEEE Data Compression Conference*, *Combinatorial Pattern Matching (CPM)*, *International Conference on String Processing and Information Retrieval (SPIRE)*, *Conference on Information and Knowledge Management (CIKM)*, *Sequences and Combinatorial Algorithms on Words, DAGS*.
- I consult in the areas of computer algorithms, data compression and communications (including text, image, and video), data storage and backup, cell phone, digital camera, and DVR technology, image and video processing, information and image retrieval, including providing expert services for computing technology related litigation.
- I have obtained patents and SBIR funding to engage in research projects (such as high speed data compression hardware) that were not possible within the university environment, but which complemented my academic research and provided practical experience on which to base research directions.

Books

Hyperspectral Data Compression

G. Motta, F. Rizzo, and J. A. Storer, Editors

Springer-Verlag, www.springer.com, November 2006

(425 pages, 6" x 9", hard-bound)

ISBN: 0-387-28579-2

This book provides a survey of recent results in the field of compression of remote sensed 3D data, with a particular interest in hyperspectral imagery. This material is intended to be of interest to researchers in a variety of areas, including multi dimensional data compression, remote sensing, military and aerospace image processing, homeland security, archival of large volumes of scientific and medical data, target detection, and image classification.

The interest in remote sensing applications and platforms (including airborne and spaceborne) has grown dramatically in recent years. Remote sensing technology has shifted from panchromatic data (a wide range of wavelengths merged into a single response), through multispectral (a few possibly overlapping bands in the visible and infrared range with spectral width of 100-200nm each), to hyperspectral imagers and ultraspectral sounders, with hundreds or thousands of bands. In addition, the availability of airborne and spaceborne sensors has increased considerably, followed by the widespread availability of remote sensed data in different research environments, including defense, academic, and commercial.

Remote sensed data present special challenges in the acquisition, transmission, analysis, and storage process. Perhaps most significant is the information extraction process. In most cases accurate analysis depends on high quality data, which comes with a price tag: increased data volume. For example, the NASA JPL's *Airborne Visible/Infrared Imaging Spectrometer* (AVIRIS, <http://aviris.jpl.nasa.gov>) records the visible and the near-infrared spectrum of the reflected light of an area 2 to 12 kilometers wide and several kilometers long (depending on the duration of the flight) into hundreds of non overlapping bands. The resulting data volume typically exceeds 500 Megabytes per flight and it is mainly used for geological mapping, target recognition, and anomaly detection. On the other hand, ultraspectral sounders such as the NASA JPL's *Atmospheric Infrared Sounder* (AIRS, <http://www-airs.jpl.nasa.gov>), which has recently become a reference in compression studies on this class of data, records thousands of bands covering the infrared spectrum and generates more than 12 Gigabytes of data daily. The major application of this sensor is the acquisition of atmospheric parameters such as temperature, moisture, clouds, gasses, dust concentrations, and other quantities to perform weather and climate forecast.

Chapter 1 addresses compression architecture and reviews and compares compression methods. Chapter 2 through 4 focus on lossless compression (where the decompressed image must be bit for bit identical to the original). Chapter 5 (contributed by the editors) describes a lossless algorithm based on vector quantization with extensions to near lossless and possibly lossy compression for efficient browsing and pure pixel classification. Chapters 6 deals with near lossless compression while Chapter 7 considers lossy techniques constrained by almost perfect classification. Chapters 8 through 12 address lossy compression of hyperspectral imagery, where there is a tradeoff between compression achieved and the quality of the decompressed image. Chapter 13 examines artifacts that can arise from lossy compression.

An Introduction to Data Structures and Algorithms

James A. Storer

Birkhäuser / Springer, www.springer.com, February 2002

(600 pages, 7" x 10", hard-bound)

ISBN 0-8176-4253-6, ISBN 3-7643-4253-6

A highly accessible format presents algorithms with one page displays that will appeal to both students and teachers of computer science. The thirteen chapters include: Models of Computation (including Big O notation), Lists (including stacks, queues, and linked lists), Induction and Recursion, Trees (including self-adjusting binary search trees), Algorithms Design Techniques, Hashing, Heaps (including heapsort and lower bounds on sorting by comparisons), Balanced Trees (including 2-3 trees, red-black trees, and AVL trees), Sets Over a Small Universe (including on-the-fly array initialization, in-place permutation, bucket sorting, bit-vectors, and the union-find data structure), Discrete Fourier Transform (including an introduction to complex numbers, development of the FFT algorithm, convolutions, the DFT on an array of reals, the discrete cosine transform, computing the DCT with a DFT of $n/2$ points, 2D DFT and DCT, and an overview of JPEG image compression), Strings (including lexicographic sorting, KMP / BM / Rabin-Karp / Shift-And string matching, regular expression pattern matching, tries, suffix tries, edit distance, Burrows-Wheeler transform, text compression examples), Graphs (including DFS / BFS, biconnected and strongly connected components, spanning trees, topological sort, Euler paths, shortest paths, transitive closure, path finding, flow, matching, stable marriage, NP-complete graph problems), Parallel Models of Computation (including the PRAM, generic PRAM simulation, the hypercube/CCC/butterfly, the mesh, and hardware area-time tradeoffs).

- Concepts are expressed clearly, in a single page, with the least amount of notation, and without the "clutter" of the syntax of a particular programming language; algorithms are presented with self-explanatory "pseudo-code".
- Each chapter starts with an introduction and ends with chapter notes and exercises that promote further learning.
- Sorting, often perceived as rather technical, is not treated as a separate chapter, but is used in many examples (including bubble sort, merge sort, tree sort, heap sort, quick sort, and several parallel algorithms). Lower bounds on sorting by comparisons are included with the presentation of heaps in the context of lower bounds for comparison based structures.
- Chapters 1-4 focus on elementary concepts, the exposition unfolding at a slower pace. Sample exercises with solutions are also provided. These chapters assume a reader with only some basic mathematics and a little computer programming experience. An introductory college-level course on data structures may be based on Chapters 1 -4 and the first half of Chapters 5 (algorithms design), 6 (hashing), and 12 (graphs).
- Chapters 5-13 progress at a faster pace. The material is suitable for undergraduates or graduates who need only review Chapters 1-4. A more advanced course on the design and analysis of algorithms may be based on these chapters.
- Chapter 13 on parallel models of computation is something of mini-book itself. The idea is to further teach fundamental concepts in the design of algorithms by exploring exciting models of computation, including the PRAM, generic PRAM simulation, HC/CCC/Butterfly, the mesh, and parallel hardware area-time tradeoffs (with many examples). A sampling of this chapter can be a fun way to end a course based on earlier portions of the book. In addition, a seminar style course can spend its first half covering this chapter in detail and then study papers from the literature.
- Apart from classroom use, this book serves as an excellent reference text on the subject of data structures, algorithms, and parallel algorithms. Its page-at-a-time format makes it easy to review material that the reader has studied in the past.

Data Compression: Methods and Theory

James A. Storer

Computer Science Press (a subsidiary of W. H. Freeman Press), 1988

(419 pages, 6" x 9", hard-bound)

ISBN 0-88175-161-8

The first two chapters contain introductory material on information and coding theory. The remaining four chapters cover some of my data compression research performed in the period 1977 - 1987 (including substantial material that has not been reported elsewhere). Chapter 3 considers on-line textual substitution methods that employ "learning" heuristics to adapt to changing data characteristics. Chapter 4 considers massively parallel algorithms for on-line methods and their VLSI implementations. Chapter 5 considers off-line methods (including the NP-completeness of certain methods). Chapter 6 addresses program size (Kolmogorov) complexity. The appendices present source code and empirical results.

Image and Text Compression

James A. Storer, Editor

Kluwer Academic Press (part of Springer), 1992

(354 pages, 6" x 9", hard-bound)

ISBN 0-7923-9243-4

This is an edited volume of papers by leading researchers in the field; topics include: vector quantization, fractals, optical algorithms, arithmetic coding, context modeling, LZ methods, massively parallel hardware (the chapter I contributed), bounds on Huffman codes, coding delay, and 2D lossless compression. Also included is a 75 page bibliography of data compression research that I compiled specifically for this book.

Proceedings Compression and Complexity

B. Carpentieri, A. De Santis, U. Vaccaro, and J. A. Storer, Editors

IEEE Computer Society Press, 1998

(400 pages, 6" x 9", hard-bound)

ISBN 0-8186-8132-2

This is an edited volume of the papers presented at the *International Conference on Compression and Complexity of Sequences*, held in Positano, Italy in 1997.

Papers address the theoretical aspects of data compression and its relationship to problems on sequences, and include contributions from the editors.

Proceedings of the Data Compression Conference

James A. Storer, Co-Chair

IEEE Computer Society Press

1991 - present (approximately 500 pages hard-bound)

I have chaired DCC since it was founded in 1991; starting in 2013 the conference leadership has been expanded; I am currently co-chair (with M. Marcellin, formally committee chair).

The DCC proceedings are co-edited with the DCC program committee chair(s), which over the years has been J. Reif (1991), M. Cohn (1992-2006), M. Marcellin (2007-2012), Ali Bilgin & Joan Serra-Sagrista (2013-present).

Each volume has ten page extended abstracts of the presentations at technical sessions and one page abstracts of presentations at the posters session. The call for papers states that topics of interest include but are not limited to:

An international forum for current work on data compression for text, images, video, audio, and related areas. Topics of interest include but are not limited to: Lossless and lossy compression algorithms for specific types of data (text, images, multi-spectral and hyper-spectral images, palette images, video, speech, music, maps, instrument and sensor data, space data, earth observation data, graphics, 3D representations, animation, bit-maps, etc.), source coding, text compression, joint source-channel coding, multiple description coding, quantization theory, vector quantization (VQ), multiple description VQ, compression algorithms that employ transforms (including DCT and wavelet transforms), bi-level image compression, gray scale and color image compression, video compression, movie compression, geometry compression, speech and audio compression, compression of multi-spectral and hyper-spectral data, compression of science, weather, and space data, source coding in multiple access networks, parallel compression algorithms and hardware, fractal based compression methods, error resilient compression, adaptive compression algorithms, string searching and manipulation used in compression applications, closest-match retrieval in compression applications, browsing and searching compressed data, content based retrieval employing compression methods, minimal length encoding and applications to learning, system issues relating to data compression (including error control, data security, indexing, and browsing), medical imagery storage and transmission, compression of web graphs and related data structures, compression applications and issues for computational biology, compression applications and issues for the internet, compression applications and issues for mobile computing, applications of compression to file distribution and software updates, applications of compression to files storage and backup systems, applications of compression to data mining, applications of compression to information retrieval, applications of compression to image retrieval, applications of compression and information theory to human-computer interaction (HCI), data compression standards including the JPEG, JPEG2000, MPEG (MPEG1, MPEG2, MPEG4, MPEG7, etc.), H.xxx, and G.xxx families.

Patents

In-Place Differential File Compression

United States patent number: 7,079,051

INVENTORS: James A. Storer and Dana Shapira

FILED: March 18, 2004

GRANTED: July 18, 2006

(23 claims, 6 of them independent)

Addresses in-place differential file compression methods that can be used in software update and backup systems.

Method and Apparatus for Data Compression

United States patent number: 5,379,036

INVENTOR: James A. Storer

FILED: April 1, 1992

GRANTED: January 1, 1995

(23 claims, 3 of them independent)

Addresses high speed parallel algorithms and hardware for data compression.

System for Dynamically Compressing and Decompressing Electronic Data

United States patent number: 4,876,541

INVENTOR: James A. Storer

FILED: October 15, 1987

GRANTED: October 24, 1989

(55 claims, 5 of them independent)

Addresses dictionary based adaptive data compression.

Computational Modeling of Rotation and Translation Capable Human Visual Pattern Recognition

U.S. Provisional Patent Application No. 60/712,596

INVENTORS: John Lisman, James A. Storer, Martin Cohn, Antonella DiLillo

FILED: August 29, 2005

(assigned to Brandeis University)

Addresses rotation and translation invariant recognition.

Papers

- "Compression-Based Tools for Navigation with an Image Database", *Algorithms* 5, 2012, 1-17 (coauthored with A. DiLillo, A. Daptardar, K. Thomas, G. Motta)
- "Edit Distance With Block Deletions", *Algorithms* 4, 2011, 40-60 (coauthored with D. Shapira).
- "Applications of Compression to Content Based Image Retrieval and Object Recognition", *Proceedings International Conference On Data Compression, Communication, and Processing* (CPP 2011), Palinuro, Italy, 179 - 189 (coauthored with Antonella Di Lillo, Ajay Daptardar, Giovanni Motta, and Kevin Thomas)
- "A Rotation And Scale Invariant Descriptor For Shape Recognition", *Proceedings International Conference On Image Processing* (ICIP), Hong Kong, 2010, MA-L9:1, 257-260 (coauthored with A. DiLillo and G. Motta).
- "Shape Recognition, With Applications To A Passive Assistant", *Proceedings PETRA - Pervasive Technologies Related to Assistive Environments*, Samos, Greece, June 23-25; to appear *ACM International Conference Proceedings Series* (coauthored with A. DiLillo, G. Motta, and K. Thomas), 2010.
- "Shape Recognition Using Vector Quantization", *Proceedings Data Compression Conference*, IEEE Computer Society Press, March 2010, 484-493 (coauthored with A. Di Lillo and G. Motta).
- "Network Aware Compression Based Rate Control for Printing Systems", *Proceedings 10th International Symposium on Pervasive Systems (ISPAN 2009)*, Kaohsiung, Taiwan, December 2009, 123-128 (coauthored with Chih-Yu Tang).
- "VQ Based Image Retrieval Using Color and Position Features", *Proceedings Data Compression Conference*, IEEE Computer Society Press, March 2008, 432-441 (coauthored with A. Daptardar).
- "Multiresolution Rotation-Invariant Texture Classification Using Feature Extraction in the Frequency Domain and Vector Quantization", *Proceedings Data Compression Conference*, IEEE Computer Society Press, March 2008, 452-461 (coauthored with A. DiLillo and G. Motta).
- "Edit Distance with Move Operations", *Journal of Discrete Algorithms* 5:2, June 2007, 380-392 (coauthored with D. Shapira).
- "Texture Classification Based on Discriminative Feature Extracted in the Frequency Domain", *Proceedings IEEE International Conference on Image Processing* (ICIP), September 2007, II.53-II.56 (coauthored with A. DiLillo and G. Motta).
- "Supervised Segmentation Based on Texture Signature Extracted in the Frequency Domain", *Third Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA)*, June 2007 (coauthored with A. DiLillo and G. Motta).

- "Texture Classification Using VQ with Feature Extraction Based on Transforms Motivated by the Human Visual System", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 392, 2007 (coauthored with A. DiLillo and G. Motta).
- "Reduced Complexity Content-Based Image Retrieval using Vector Quantization", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 342-351, 2006 (coauthored with A. H. Daptardar).
- "In-Place Differential File Compression", *Computer Journal* 48:6, 677-691, November 2005 (coauthored with D. Shapira).
- "Compression of Hyper/Ultra-Spectral Data", *Proceedings of SPIE, Optics and Photonics, Satellite Data Compression, Communication and Archiving*, Jul. 2005, Vol. 5889, pp. 588908-1--588908-10 (coauthored with G. Motta and F. Rizzo).
- "Content-Based Image Retrieval Via Vector Quantization", In *Advances in Visual Computing - Springer Lecture Notes in Computer Science* 3804/2005 (ISBN 3-540-30750-8), 502-509; also appeared in *International Symposium on Visual Computing*, December 5-7, 2005 (coauthored with A. Daptardar).
- "Low Complexity Lossless Compression of Hyperspectral Imagery via Linear Prediction", *IEEE Signal Processing Letters* 12:2, 138-141, 2005 (coauthored with F. Rizzo, B. Carpentieri, and G. Motta).
- "Overlap and Channel Errors in Adaptive Vector Quantization for Image Coding", *Information Sciences* 171:1-3, 125-143, 2005 (coauthored with B. Carpentieri and F. Rizzo).
- "Compression of Hyperspectral Imagery", *Proceedings International Conference on E-Business and Telecommunications (ICETE)*, Sebutal, Portugal, 2004; an extended version of this paper is a chapter in the book *E-Business and Telecommunication Networks*, Edited by J. Ascenso, L. Vasiu, C. Belo, and M. Saramago, Kluwer / Springer, 317-324 (coauthored with B. Carpentieri, G. Motta, and F. Rizzo).
- "Real-Time Software Compression and Classification of Hyperspectral Images", *Proceedings 11th SPIE International Symposium on Remote Sensing Europe*, X. L. Bruzzone, Ed., Maspalomas, Gran Canaria, Spain, Sept. 13-17, 2004, Vol. 5573, 182-192, (coauthored with B. Carpentieri, G. Motta, and F. Rizzo).
- "Lossless Compression of Hyperspectral Imagery: A Real Time Approach", *Proceedings 11th SPIE International Symposium on Remote Sensing Europe*, X. L. Bruzzone, Ed., Maspalomas, Gran Canaria, Spain, Sept. 13-17, 2004, Vol. 5573, 262-272, (coauthored with B. Carpentieri, G. Motta, and F. Rizzo).
- "High Performance Compression of Hyperspectral Imagery with Reduced Search Complexity in the Compressed Domain", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 2004, 479-488 (coauthored with B. Carpentieri, G. Motta and F. Rizzo).

"In-Place Differential File Compression of Non-Aligned Files With Applications to File Distribution and Backups", *Proceedings Data Compression Conference (DCC)*, IEEE Computer Society Press, 2004, 82-91 (coauthored with D. Shapira).

"Report of the National Science Foundation Workshop on Information Theory and Computer Science Interface", produced by workshop of 20 invited participants (Chicago, IL 2003), report completed and submitted to the NSF in November 2004.

"Finding Non-Overlapping Common Substrings in Linear Time", *Proceedings Symposium on String Processing and Information Retrieval (SPIRE)*, 2003, *Lecture Notes on Computer Science*, Volume 2857/2003 ISBN: 3-540-20177-7, Springer-Verlag, 369-377 (coauthored with M. Meyerovich and D. Shapira).

"In-Place Differential File Compression", *Proceedings Data Compression Conference (DCC)*, IEEE Computer Society Press, 263-272, 2003 (coauthored with D. Shapira).

"Compression of Hyperspectral Imagery", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 333-342, 2003 (coauthored with G. Motta and F. Rizzo).

"Partitioned Vector Quantization: Application to Lossless Compression of Hyperspectral Images", *Proceedings International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, IMSP-P2.2-III, 241-244, 2003; also presented at *IEEE International Conference on Multimedia and Expo (ICME2003)*, 2003, 553-556 (coauthored with G. Motta and F. Rizzo).

"Generalized Edit Distance with Move Operations", *Thirteenth Annual Symposium on Combinatorial Pattern Matching (CPM)*, 85-98, 2002 (coauthored with D. Shapira).

"A Lossless Data Compression Algorithm for Electro-Cardiograms", Technical Report, Computer Science Department, Brandeis University (coauthored with G. Motta and M. Hosang); see also senior honors thesis of M. Hosang.

"LZ-Based Image Compression", *Information Sciences* 135, 107-122, 2001 (coauthored with F. Rizzo and B. Carpentieri).

"Optimal Encoding of Non-Stationary Sources", *Information Sciences*, 87-105, 2001 (coauthored with J. Reif).

"Overlap in Adaptive Vector Quantization", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 401-410, 2001 (coauthored with F. Rizzo).

"Lossless Image Coding via Adaptive Linear Prediction and Classification", *Proceedings of the IEEE* 88:11, 1790-1796 (coauthored with G. Motta and B. Carpentieri), November 2000.

"Optimal Frame Skipping for H263+ Video Encoding", *Proceedings 11th Annual International Conference on Signal Processing Applications and Technology*, Dallas, TX October 2000, 373-377 (coauthored with G. Motta).

- "Digital Image Compression", *Encyclopedia of Computer Science*, 4th Edition, John Wiley and Sons, 836-840, 2003 (coauthored with G. Motta and F. Rizzo). Also in *Concise Encyclopedia of Computer Science*, John Wiley & Sons, 5th edition, 2004, 379-382.
- "Error Resilient Dictionary Based Compression", *Proceedings of the IEEE* 88:11, 1713-1721, 2000.
- "Improving Scene Cut Quality for Real-Time Video Decoding", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 470-479, 2000 (coauthored with G. Motta and B. Carpentieri).
- "On Sequence Assembly", Technical Report, Computer Science Department, Brandeis University (coauthored with F. Mignosi, A. Rivesto, and M. Sciortino), 2000.
- "Adaptive Linear Prediction Lossless Image Coding", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 491-500, 1999 (coauthored with G. Motta and B. Carpentieri).
- "Experiments with Single-Pass Adaptive Vector Quantization", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 546, 1999 (coauthored with F. Rizzo and B. Carpentieri).
- "Improving Single-Pass Adaptive VQ", *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, IMDSP2.10, Phoenix, Arizona, 1999 (coauthored with F. Rizzo and B. Carpentieri).
- "The Prevention of Error Propagation in Dictionary Compression with Update and Deletion", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 1998, 199-208.
- "Optimal Lossless Compression of a Class of Dynamic Sources", *Proceedings IEEE Data Compression Conference*, 1998, 501-510 (coauthored with J. Reif).
- "Error Resilient Data Compression with Adaptive Deletion", in *Compression and Complexity of Sequences*, IEEE Press, 285-294, 1998.
- "Lossless Image Compression by Block Matching", *The Computer Journal* 40:2/3, 137-145, 1997 (coauthored with H. Helfgott).
- "Error Resilient Optimal Data Compression", *SIAM Journal of Computing* 26:4, 934-939, 1997 (coauthored with J. Reif).
- "Low-Cost Prevention of Error-Propagation for Data Compression with Dynamic Dictionaries", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 171-180, 1997 (co-authored with J. Reif).
- "Generalized Node Splitting and Bi-Level Image Compression", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 443, 1997 (co-authored with H. Helfgott).

- “Selective Resolution for Surveillance Video Compression”, *Proceedings Data Compression Conference*, IEEE Computer Society Press, 468, 1997 (co-authored with I. Schiller, C. Chuang, and S. King).
- "A Video Coder Based on Split-Merge Displacement Estimation", *Journal of Visual Communication and Visual Representation* 7:2, 137-143, 1996; an abstract of this paper appeared in *Proceedings DCC 1993*, 492 (co-authored with B. Carpentieri).
- "On-Line versus Off-Line Computation in Dynamic Text Compression", *Information Processing Letters* 59, 169-174, 1996 (co-authored with S. DeAgostino).
- “Lossless Image Compression using Generalized LZ1-Type Methods”, *Proceedings Data Compression Conference*, IEEE Computer Society Press, 290-299, 1996.
- “High Performance Adaptive Data Compression”, *Proceedings DAGS: Electronic Publishing and the Information Superhighway*, Boston, MA, 29-38, 1995 (co-authored with C. Constantinescu and B. Carpentieri).
- “Single-Pass Adaptive Lossy Compression with Pattern Matching”, *Twenty-Fourth Annual IEEE Communication Theory Workshop*, Santa Cruz, CA, April 23-26, 1995.
- “Near Optimal Compression with Respect to a Static Dictionary on a Practical Massively Parallel Architecture”, *Proceedings Data Compression Conference*, IEEE Computer Society Press, 172-181, 1995 (co-authored with D. Belinskaya and S. DeAgostino).
- “Application of Single-Pass Adaptive VQ to BiLevel Images”, *Proceedings Data Compression Conference*, IEEE Computer Society Press, 423, 1995 (co-authored with C. Constantinescu).
- “Classification of Objects in a Video Sequence”, *Proceedings SPIE Symposium on Electronic Imaging*, San Jose, CA, 1995 (co-authored with B. Carpentieri).
- "Optimal Inter-Frame Alignment for Video Compression", *International Journal of Foundations of Computer Science* 5:2, 65-177, 1994 (co-authored with B. Carpentieri).
- "Improved Techniques for Single-Pass Vector Quantization", *Proceedings of the IEEE* 82:6, 933-939, 1994; an extended abstract of this paper appeared in the *Proceedings DCC 1994*, 410-419 (co-authored with C. Constantinescu).
- "Split-Merge Video Displacement Estimation", *Proceedings of the IEEE* 82:6, 940-947, 1994; an extended abstract of this paper appeared as "A Split and Merge Parallel Block-Matching Algorithm for Displacement Estimation", *Proceedings DCC 1992*, 239-248 (co-authored with B. Carpentieri).

- "On-Line Adaptive Vector Quantization with Variable Size Codebook Entries", *Information Processing and Management* 30:6, 745-758, 1994; a preliminary version of this paper appeared as "On-Line Adaptive Vector Quantization with Variable Size Vectors", *Proceedings DCC 1993*, 32-41 (co-authored with C. Constantinescu).
- "Design and Performance of Tree-Structured Vector Quantizers", *Information Processing and Management* 30:6, 1994, 851-862; a preliminary version of this paper appeared in the *Proceedings DCC 1993*, 292-301 (co-authored with J. Lin).
- "Shortest Paths in the Plane with Polygonal Obstacles", *Journal of the ACM* 41:5, 982-1012, 1994 (co-authored with J. Reif).
- "A Single-Exponential Upper Bound for Finding Shortest Paths in Three Dimensions" *Journal of the ACM* 41:5, 1013-1019, 1994 (co-authored with J. Reif).
- "Image Compression", *Encyclopedia of Computer Science and Technology*, Volume 29, 231-256, 1994 (co-authored with J. Lin).
- "On the Design and Implementation of a Lossless Data Compression and Decompression Chip", *IEEE Journal of Solid-State Circuits*, 948-953, 1993 (co-authored with D. M. Royals, T. Markas, N. Kanopoulos, and J. H. Reif).
- "On Parallel Implementations and Experimentations of Lossless Data Compression Algorithms", *Proceedings Picture Coding Symposium*, Lausanne, Switzerland, Section 5.3, March 1993; a preliminary version of this paper appears in *Proceedings DCC 1992*, (co-authored with J. Reif and T. Markas).
- "High Performance Compression of Science Data", *NASA Science Information Systems* 29, September 1993, 1-6.
- "Split Merge Displacement Estimated Video Compression", *Proceedings 7th International Conference on Image Analysis and Processing*, Bari, Italy, September, 1993 (co-authored with B. Carpentieri).
- "A Video Coder Based on Split-Merge Displacement Estimation", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 492, 1993 (co-authored with B. Carpentieri).
- "Design and Performance of Tree-Structured Vector Quantizers", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 292-301, 1993 (co-authored with J. Lin).
- "On-Line Adaptive Vector Quantization with Variable Size Codebook Entries", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 32-41, 1993 (co-authored with C. Constantinescu).

“Adaptive Lossless Data Compression over a Noisy Channel”, *Methods in Communication, Security, and Computer Science*, Springer-Verlag, 104-117, 1993; a preliminary version of this paper appeared in *Proceedings Sequences 91*, Positano, Italy 1991 (co-authored with J. Reif).

"Improving Search for Tree-Structured Vector Quantization", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 339-348, 1992 (co-authored with J. Lin).

“Parallel Algorithms for Optimal Compression Using Dictionaries with the Prefix Property”, *Proceedings DCC '92*, 152-61 (co-authored with S. DeAgostino).

"Parallel Algorithms for Data Compression", invited presentation, 1992 ALGOFEST symposium, Bristol, RI.

“Summary of Data Compression Splinter Group”, *Proceedings AISRP Workshop*, Boulder Colorado, August, 1992.

“Massively Parallel Systolic Algorithms for Real-Time Dictionary-Based Text Compression”, *Image and Text Compression*, Kluwer Academic Press, 159-178, 1992.

“Data Compression Bibliography”, *Image and Text Compression*, Kluwer Academic Press, 277-349, 1992.

“Optimal Pruning for Tree-Structured Vector Quantization”, *Information Processing and Management* 28:6, 723-733, 1992; a preliminary version of this paper appeared as "On the Complexity of Optimal Tree Pruning for Source Coding", *Proceedings DCC 1991*, 63-72, 1991 (co-authored with E. Lin and M. Cohn).

“Data Compression”, *Information Processing and Management* 28:6, 1992 (co-authored with A. Bookstein).

"A Split and Merge Parallel Block-Matching Algorithm for Displacement Estimation", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 239-248, 1992 (co-authored with B. Carpentieri).

"On Parallel Implementations and Experimentations of Lossless Data Compression Algorithms", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 425, 1992 (co-authored with J. Reif and T. Markas).

“A Report on Sequences91”, SIGACT News, 1991.

“A Parallel Architecture for High Speed Data Compression”, *Journal of Parallel and Distributed Computing* 13, 222-227, 1991; a preliminary version of this paper appeared as "A Parallel Architecture for High Speed Data Compression", *Third Symposium on the Frontiers of Massively Parallel Computation*, College Park, Maryland, IEEE Press, 1990, 238-243 (co-authored with J. Reif).

- “Processor-Efficient Algorithms for the Knapsack Problem”, *Journal of Parallel and Distributed Computing* 11, 332-337, 1991; a preliminary version of this paper appeared as "A New Parallel Algorithm for the Knapsack Problem and its Implementation on the Hypercube", *Third Symposium on the Frontiers of Massively Parallel Computation*, College Park, Maryland, 1990, 2-7 (co-authored with J. Lin).
- “Resolution Constrained Tree Structured Vector Quantization for Image Compression”, *Proceedings IEEE Symposium on Information Theory*, Budapest, Hungary, 1991 (co-authored with J. Lin).
- “The Worst-Case Performance of Adaptive Huffman Codes”, *Proceedings IEEE Symposium on Information Theory*, Budapest, Hungary, 1991.
- "On the Complexity of Optimal Tree Pruning for Source Coding", *Proceedings Data Compression Conference*, IEEE Computer Society Press, 63-72, 1991 (co-authored with E. Lin and M. Cohn).
- “Adaptive Lossless Data Compression over a Noisy Channel”, *Proceedings Sequences 91*, Positano, Italy, 1991 (co-authored with J. Reif).
- “A Massively Parallel VLSI Design for Data Compression using a Compact Dynamic Dictionary”, *Proceedings IEEE VLSI Signal Processing Conference*, San Diego, CA, 329-338, 1990 (co-authored with J. H. Reif and T. Markas).
- “Quantization Techniques for Image Compression”, *Microcomputers* 14, 139-164 (co-authored with J. Lin).
- “Reliable Dynamic Compression for Bandwidth Reduction of Noisy Data Sent from Space”, *Proceedings CESDIS Conference*, Greenbelt, MD, 1990.
- “Lossy On-Line Dynamic Data Compression”, *Combinatorics, Compression, Security and Transmission*, Springer-Verlag, 348-357, 1990; a preliminary version of this paper was presented as "Lossy Data Compression", *Proceedings Sequences 88*, Positano, Italy, 1988.
- "A New Parallel Algorithm for the Knapsack Problem and its Implementation on the Hypercube", *Third Symposium on the Frontiers of Massively Parallel Computation*, College Park, Maryland, 2-7, 1990 (co-authored with J. Lin).
- "A Parallel Architecture for High Speed Data Compression", *Third Symposium on the Frontiers of Massively Parallel Computation*, College Park, Maryland, IEEE Press, 238-243, 1990 (co-authored with J. Reif).
- “The Amalfi and Maratea Conferences”, SIGACT News, 1988.
- “Parallel Algorithms for On-Line Dynamic Data Compression”, *Proceedings IEEE International Conference on Communications Philadelphia, PA*, 385-389, 1988.

- “Real-Time Dynamic Compression of Video on a Grid-Connected Parallel Computer”, *Proceedings Third International Conference on Super-Computing*, Boston, MA, 453-462, 1988.
- “3-Dimensional Shortest Paths in the Presence of Polyhedral Obstacles”, *Mathematical Foundations of Computer Science 3:24*, 85-92, 1988 (co-authored with J. Reif).
- “Science Data Management”, *Proceedings NASA Conference on Scientific Data Compression*, Snowbird, Utah, 1988 (co-authored with R. B. Miller).
- "Lossy Data Compression", *Proceedings Sequences 88*, Positano, Italy, 1988.
- “Minimizing Turns for Discrete Movement in the Interior of a Polygon”, *IEEE Journal of Robotics and Automation 3:3*, 182-193, 1987 (co-authored with J. Reif).
- “Parallel Algorithms for Data Compression”, *Journal of the ACM 32:2*, 344-373, 1985 (co-authored with M. Gonzalez).
- “Textual Substitution Techniques for Data Compression”, *Combinatorial Algorithms on Words*, Springer-Verlag, 111-129, 1985.
- “Uniform Circuit Placement”, *VLSI Algorithms and Architectures 1*, 255-273, 1985 (co-authored with A. J. Nicas and J. Becker).
- “On Minimal Node Cost Planar Embeddings”, *Networks 14:2*, 181-212, 1984; a preliminary version of this paper appeared as "The Node Cost Measure for Embedding Graphs on the Planar Grid", *Proceedings Twelfth Annual ACM Symposium on the Theory of Computing*, Los Angeles, CA, 201-210, 1980.
- “On the Complexity of Chess”, *Journal of Computer and System Sciences 27:1*, 77-100, 1983; a preliminary version of this paper appeared as "A Note on the Complexity of Chess", *Proceedings Thirteenth Annual CISS*, 160-166, 1979.
- “An Abstract Theory of Data Compression”, *Theoretical Computer Science 24*, 221-237, 1983; a preliminary version of this paper appeared as "Toward an Abstract Theory of Data Compression", *Proceedings Twelfth Annual CISS*, 391-399, 1978.
- “Data Compression Via Textual Substitution”, *Journal of the ACM 29:4*, 928-951, 1982; a preliminary version of this paper appeared as "The Macro Model for Data Compression", *Proceedings Tenth Annual ACM Symposium on the Theory of Computing*, San Diego, CA, 30-39, 1978 (co-authored with T. G. Szymanski).
- “Constructing Full Spanning Trees for Cubic Graphs”, *Information Processing Letters 13:1*, 8-11, October, 1981.
- “On Finding Minimal Length Superstrings”, *Journal of Computer and System Sciences 20*, 50-58, 1980; a preliminary version of this paper appeared as "A Note on the Complexity of

- the Superstring Problem", *Proceedings Twelfth Annual CISS*, 52-56, 1978 (co-authored with D. Maier).
- "The Node Cost Measure for Embedding Graphs on the Planar Grid", *Proceedings Twelfth Annual ACM Symposium on the Theory of Computing*, Los Angeles, CA, 201-210, 1980.
- "A Note on the Complexity of Chess", *Proceedings Thirteenth Annual CISS*, 160-166, 1979.
- "The Macro Model for Data Compression", *Proceedings Tenth Annual ACM Symposium on the Theory of Computing*, San Diego, CA, 30-39, 1978 (co-authored with T. G. Szymanski).
- "A Note on the Complexity of the Superstring Problem", *Proceedings Twelfth Annual CISS*, 52-56, 1978 (co-authored with D. Maier).
- "Toward an Abstract Theory of Data Compression", *Proceedings Twelfth Annual CISS*, 391-399, 1978.
- "Data Compression: Methods and Complexity Issues", Ph.D. Thesis, Dept. of Electrical Engineering and Computer Science, Princeton University Princeton, NJ, 1979.

Technical Reports (not published elsewhere)

- "Edit Distance with Move Operations", Technical Report (coauthored with D. Shapira).
- "Longest Common Substrings and Edit Distance", Technical Report, Computer Science, Brandeis University, Waltham, MA (co-authored with M. Meyerovich and D. Shapira); see also Senior Honors thesis of M. Meyerovich.
- "Parallel Implementation of Single-Pass Adaptive VQ", Technical Report, Department of Computer Science, Brandeis University, Waltham, MA (co-authored with C. Constantinescu).
- "Application of Single-Pass VQ to Mixed Images", Technical Report, Department of Computer Science, Brandeis University, Waltham, MA (co-authored with C. Constantinescu).
- "Geometric Clustering to Minimize the Mean-Square Error" Technical Report, Department of Computer Science, Brandeis University, Waltham, MA (co-authored with J. Lin).
- "Optimal Resolution-Constrained Tree-Structured Vector Quantization" Technical Report, Dept. of Computer Science, Brandeis University, Waltham, MA (co-authored with J. Lin).
- "Computing Range Sets, with applications to Automatic Worst-Case Asymptotic Analysis of Programs", Technical Report CS-85-123, Department of Computer Science, Brandeis University, Waltham, MA, August, 1985.
- "A Note on the 2-Dimensional Channel Assignment Problem", manuscript, 1984.
- "Combining Trees and Pipes in VLSI", Technical Report CS-82-107, Dept. of Computer Science, Brandeis University, Waltham, MA, June, 1982.
- "The Page-Control Macros for UNIX troff Typesetting", Technical Report CS-82-106, Dept. of Computer Science, Brandeis University, Waltham, MA, September, 1982.
- "Bridge: A Unix Experiment for Home Computer Services", Technical Report 80-1352-13, Bell Laboratories Murray Hill, NJ, April, 1980.
- "Improved Lower Bound for On-Line Two Dimensional Packing with Decreasing Width", Technical Report 79-1352-8, Bell Laboratories Murray Hill, NJ, July, 1979; also: "A Further Note Concerning On-Line Two Dimensional Bin Packing With Decreasing Width", Technical Report 80-1352-18, Bell Laboratories, Murray Hill, NJ, August, 1980.
- "PLCC: A Compiler-Compiler for PL/1 and PL/C Users", Technical Report 236, Dept. of Electrical Engineering and Computer Science, Princeton University, Princeton, NJ, December, 1977.
- "NP-Completeness Results Concerning Data Compression", Technical Report 234, Dept. of Electrical Engineering and Computer Science, Princeton University, Princeton, NJ, November, 1977.
- "Non-Deterministic Switching Circuits", manuscript, 1975.

APPENDIX B

APPENDIX B: MATERIALS CONSIDERED

Exhibit	Description
1101	U.S. Patent No. 7,415,530 (“’530 Patent”)
1103	U.S. Patent No. 5,805,932 (“Kawashima”)
1104	U.S. Patent No. 6,253,264 (“Sebastian”)
1105	U.S. Patent No. 5,467,087 (“Chu”)
1106	U.S. Patent No. 7,079,051 (“Storer ’051”)
1107	’927 Reexamination File History, 5/31/13 Right of Appeal Notice
1108	U.S. Patent No. 6,601,104 (“’104 Patent”)
1109	U.S. Patent No. 5,150,430 (“Chu ’430”)
1110	U.S. Patent No. 4,593,324 (“Ohkubo”)
1111	U.S. Patent No. 5,794,229 (“French”)
1112	U.S. Patent No. 4,956,808 (“Aakre”)
1113	U.S. Patent No. 3,394,352 (“Wernikoff”)
1114	Random House Webster’s, Computer & Internet Dictionary at 45 (3d ed. 1999)
1115	Storer, <i>Data Compression: Methods and Theory</i> , Computer Science Press (1988) (“Storer 1988”)
1116	D. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” Proceedings of the IRE 40, 1098-1101 (1952)
1117	J. Ziv and A. Lempel, “A Universal Algorithm for Sequential Data Compression,” IEEE Transactions on Information Theory 23:3, 337-343 (1977)
1118	J. Ziv and A. Lempel, “Compression of Individual Sequences Via Variable-Rate Coding,” IEEE Transactions on Information Theory

Exhibit	Description
	(1978)
1119	U.S. Patent No. 5,379,036 (“Storer ’036”)
1120	U.S. Patent No. 4,876,541 (“Storer ’541”)