

Enhancing Robot Programming with Visual Feedback and Augmented Reality

Stéphane Magnenat
Disney Research Zurich,
Switzerland
stephane@magnenat.net

Morderchai Ben-Ari
Dept. of Science Teaching
Weizmann Inst. Sci., Israel
moti.ben-
ari@weizmann.ac.il

Severin Klingler
Computer Graphics Lab.
ETH Zurich, Switzerland
severin.klingler@inf.ethz.ch

Robert W. Sumner
Disney Research Zurich,
ETH Zurich, Switzerland
sumner@disneyresearch.com

ABSTRACT

In our previous research, we showed that students using the educational robot Thymio and its visual programming environment were able to learn the important computer-science concept of event-handling. This paper extends that work by integrating *augmented reality* (AR) into the activities. Students used a tablet that displays in real time the event executed on the robot. The event is overlaid on the tablet over the image from a camera, which shows the location of the robot when the event was executed. In addition, visual feedback (FB) was implemented in the software. We developed a novel video questionnaire to investigate the performance of the students on robotics tasks. Data were collected comparing four groups: AR+FB, AR+non-FB, non-AR+FB, non-AR+non-FB. The results showed that students receiving feedback made significantly fewer errors on the tasks. Those using AR made fewer errors, but this improvement was not significant, although their performance improved. Technical problems with the AR hardware and software showed where improvements are needed.

Categories and Subject Descriptors

K.3.2 [Computers & Education]: Computer and Information Science Education - *Computer Science Education*; I.2.9 [Robotics]

General Terms

Human Factors

Keywords

robotics in education; Thymio; Aseba; VPL; augmented reality; event-actions pair

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ITiCSE'15, July 04–08, 2014, Vilnius, Lithuania.

Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM 978-1-4503-3440-2/15/07 ...\$15.00.

<http://dx.doi.org/10.1145/2729094.2742585>.

1. INTRODUCTION

Robotics activities are widely used to introduce students to *science, mathematics, engineering, technology* (STEM) in general and to *computer science* in particular [6]. Robotics activities are exciting and fun, but we are also interested in investigating if the activities lead to learning of STEM subjects. In a previous paper [9], we described research conducted during an outreach program using the *Thymio II* education robot and its *Visual Programming Language* (VPL). We showed that students successfully learned the important computer-science concept of event-handling.

However, while students were able to comprehend behaviors consisting of independent events, they had trouble with sequences of events. This paper explores two independent ways of improving their understanding of robotics programming: *visual feedback* (FB) that shows which event handler is currently being executed and *augmented reality* (AR) (as originally suggested by the first author [7]).

The research methodology was improved. In [9], learning was measured by administering a textual questionnaire containing exercises about VPL programs and the behaviors of the robot that could be observed when the programs were run. We observed that some young students found the textual questionnaire difficult to understand. Therefore, we implemented a new type of research instrument—a video questionnaire—where the students were given a multiple-choice among several short video clips.

The performance of the students was measured in a 2×2 experimental setup: treatment groups that used AR compared with control groups that did not, and treatment groups that received FB compared with those that did not.

Section 2 describes the robot and the software environment, while Section 3 discusses previous work on AR in education and the AR system that we developed. The research methodology and the design of the video questionnaire are presented in Section 4. The results of the analysis, the discussion and the limitations of the research appear in Sections 5–7. Section 8 describes our plans for the future.

2. THYMIO II AND ASEBA

The Thymio II robot [11] (Figure 1) and its Aseba software were created at the Swiss Federal Institute of Technology

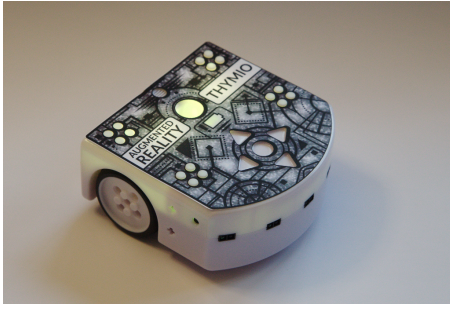


Figure 1: The Thymio II robot with a top image for tracking by the camera of the tablet.

(EPFL and ETHZ) and ECAL (University of Arts and Design). Both the hardware design and the software are open-source.

The robot is small ($11 \times 11 \times 5$ cm), self-contained and robust with two independently-driven wheels for differential drive. It has five proximity sensors on the front and two on the back, and two sensors on the bottom. There are five buttons on the top, a three-axis accelerometer, a microphone, an infrared sensor for receiving signals from a remote control and a thermometer. For output, there are RGB LEDs at the top and bottom of the robot, as well as mono-colored LEDs next to the sensors, and a sound synthesizer. A printed image was attached to the top of the robot so that the camera could recognize the robot when AR was used (Figure 1).

The Aseba programming environment [8] uses the construct **onevent** to create event handlers for the sensors. VPL is a component of Aseba for visual programming.¹ Figure 2 shows a VPL program for following a line of black tape on a white floor. On the left is a column of *event blocks* and on the right is a column of *action blocks*. By dragging and dropping one event block and one or more action blocks to the center pane, an *event-actions pair* is created. Both event and action blocks are parametrized, enabling the user to create many programs from a small number of blocks.

The robotic activities reported here used a development version of VPL; the most important improvement is that several actions can be attached to a single event.²

Visual feedback is implemented by causing an event-actions pair to blink whenever it is executed. This facilitates understanding the temporal relation between the pairs and the spatial relation between the robot and its environment.

3. AUGMENTED REALITY

3.1 Background

Visual programming languages have been used extensively [1] and event-based programming is claimed to be an effective approach for teaching introductory programming [5]. However, we found that the asynchronous nature of visual event-based programming renders the understanding and tracing of their execution difficult [9]. Nevertheless, when a visual language is used, we can perceive a relation between the spatiotemporal location of the robot and the execution of the program. Building on the neurological evidence on

¹A reference manual and a tutorial are available at <http://aseba.wikidot.com/en:thymioprogram>.

²This is the reason we now use the term event-actions pair.

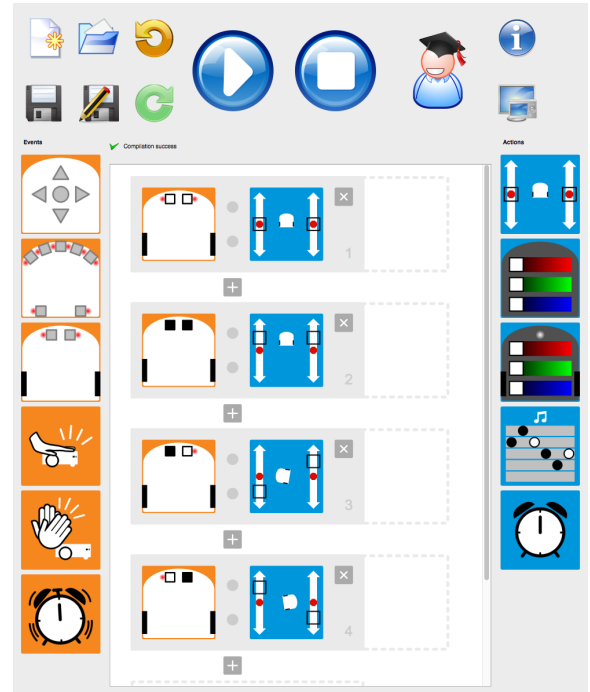


Figure 2: The Aseba/VPL environment

grounded cognition [3], we propose to make this relation explicit and available to the student. Our hypothesis is that this will allow the students to understand better what their program is doing and to lead them to learn faster. We propose to use a tablet to provide a “window” into the “live mind” of the robot, localizing the robot using AR. The resulting live inspection system uses both the spatiality and the temporality of event execution to help students understand what their program is doing.

Previous work on AR in education [2, 13] has highlighted that AR systems have a cost in term of weight and bulkiness; furthermore, little quantitative comparison of their effectiveness against non-AR solutions for the same problem has been conducted [13]. Moreover, there has been little use of AR in computer-science education. Some work has explored how to input programs using physical artifacts [4], but none has used AR to provide facilities for tracing and debugging.

3.2 The augmented reality system

The AR system consists of an Android or iOS application, which runs on a tablet and connects to the computer running VPL. The application finds the position of the tablet by detecting a ground image using the tablet’s camera (Figure 3), and the position of the robot by detecting the printed image on its top (Figure 1). The camera’s image is shown on the screen of the tablet and is overlaid with the event-actions pairs at the physical locations they were executed by the robot (Figure 4). At the bottom of the tablet screen, the execution times of these events are shown on a local time line that can be scrolled and zoomed using drag and pinch gestures. Augmented pairs can be selected by touching them on screen, and the corresponding time in the local timeline will be highlighted. A global timeline indicates which part of the total recording the local timeline shows.



Figure 3: The concept of the VPL AR App

The AR system was implemented using the Vuforia library³ for tracking the image of the ground and the top of the robot, and its plugin for the Unity framework.⁴ It communicates with VPL through TCP/IP.

4. RESEARCH METHODOLOGY

4.1 Population

The workshops consisted of 14 sessions of 75 minutes. Two sessions were run in parallel. The workshops took place in Lugano, Switzerland on October 16–17 2014.

There were 10–18 high-school students per session from high schools in the Swiss canton of Ticino. The median age of the students was 16 (low/high quartiles: 16/17). Consent forms were required and participants were allowed to opt out of the study.

There was one robot and tablet per 2 or 3 students. Various models of iOS and Android tablets (7–10 ") were used.

There were four teaching assistants, two per room, who were students at USI (Università della Svizzera italiana). The same two assistants were always paired together. After every two sessions, the assistants exchanged rooms to prevent bias due to a specific pair of assistants.

4.2 Experimental setup

There were two independent variables in our research:

- Augmented reality: AR was used by the students in room 1, while in room 2 VPL was used without AR.
- Visual feedback: The first session in each room used FB from VPL and the AR system, whereas FB was not used during the second session.

The first 15 minutes of each session were devoted to introducing the robot and its built-in behaviors. During the next 15 minutes, the students learned about VPL; this was followed by 30 minutes devoted to solving increasingly challenging tasks.⁵ During the final 15 minutes the students answered the video questionnaire.

³<http://www.qualcomm.com/products/vuforia>

⁴<http://www.unity3d.com>

⁵The tasks were the same as those used in [9]. Some tasks were very simple like changing colors when a button is

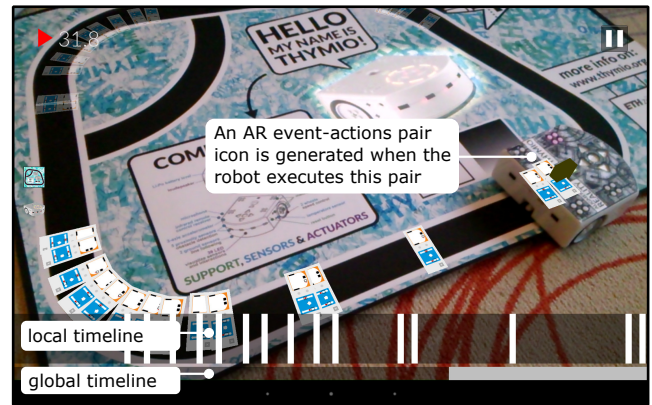


Figure 4: The GUI of the VPL AR App

We collected usage data from the VPL editor: addition and deletion of blocks, change of parameters, and clicks on buttons. For the AR system, we collected usage data from the tablet: its position in 3D, the position of the robot in 2D, and the state of the application.⁶

4.3 The questionnaire

In our previous research [9], the questions consisted of an image of a VPL program, together with multiple-choice responses that were textual descriptions of the behavior of the robot when executing the program.⁷ The students were asked to choose the description that correctly described the behavior. We found that some students had difficulty understanding the textual descriptions.

For the current research we used a novel type of questionnaire based upon video clips.⁸ There were eight questions, four each of the following types:

- The student is shown a video of the behavior of a robot and then asked to select a program (one out of four) that causes the behavior.
- The student is shown a program and four videos and then be asked to select the video that demonstrates the behavior of the robot running the program.

The questionnaire was constructed using the Forms facility of Google Drive. In Forms, you can include videos by uploading the videos to YouTube and providing the URL.

The questionnaire in [9] was constructed using the taxonomy in [10] that combines the Bloom and SOLO taxonomies. For this research, we limited the questions to those at the *applying* level of the taxonomy, because a student who is able to track the execution of a program can be assumed to understand what the individual instructions do and how they work together, whereas at the *understanding* level, rote learning might be sufficient to select a correct answer. Evaluating higher cognitive levels such as *creating* will be done in later phases of the research. In VPL, every program has both pressed; others were more difficult like following a track or navigating a labyrinth.

⁶The raw data is available at <https://aseba.wikidot.com/en:thymiopaper-vpl-iticse2015>.

⁷<http://thymio.org/en:thymiopaper-vpl-iticse2014>

⁸<http://thymio.org/en:thymiopaper-vpl-iticse2015>

	treatment	control	p-value
Feedback	0.81 (n = 47)	1.74 (n = 34)	0.003
Augm. Reality	1.00 (n = 41)	1.40 (n = 40)	0.10

Table 1: The mean mistake count and the p-value of Pearson’s chi-square test for different conditions.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
AF	0.0	0.0	4.0	32.0	4.0	24.0	0.0	4.0
AN	18.8	0.0	0.0	37.5	18.8	25.0	18.8	31.2
NF	0.0	0.0	0.0	22.7	18.2	18.2	18.2	18.2
NN	16.7	5.6	5.6	38.9	16.7	44.4	33.3	33.3

Table 2: The error rate (%) of the questionnaire answers. n = AF:25, AN:16, NF:22, NN:18

an event and an action, so we did not make the unistructural / multistructural distinction from the SOLO taxonomy.

5. RESULTS

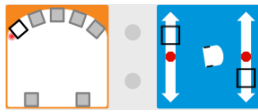
5.1 The questionnaire

To compare the treatment vs. the control groups, we counted the number of mistakes for every participant. Table 1 shows the mean mistake count and the p-value of Pearson’s chi-square test of its histograms, for the null hypothesis of no effect. We used Laplace smoothing (adding 1 to each bin) to apply the chi-square test even when the control group has 0 entries for a given mistake count. We see that using FB decreases the mistake count significantly, while using AR is not significant.

Table 2 shows the error rate of the answers for the four setups in the experiment: AF = AR and FB, AN = AR with no FB, NF = no AR but with FB, NN = neither AR nor FB. We see that some questions were answered correctly by almost all students, while other questions were more difficult and more than 30 % of the students gave the wrong answers. We also see that the error rate depends on the setup.

We see from Table 3 that the error rate is always lower with FB than without, and that this difference is significant for Q1 and Q8, and borderline significant for Q7.

Q1 showed a video of the robot turning right when presented with an object in front of its left sensor. The students had to select one of four programs, which differed in inverting the left/right sensors and the direction of movement. The correct program was:



	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
F	0.0	0.0	2.1	27.7	10.6	21.3	8.5	10.6
N	17.6	2.9	2.9	38.2	17.6	35.3	26.5	32.4
p	0.01	0.87	0.62	0.44	0.56	0.25	0.06	0.03

Table 3: The error rate (%) for fb/non-fb; p-values of Pearson’s chi-square test. n = F:47, N:34

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
A	7.3	0.0	2.4	34.1	9.8	24.4	7.3	14.6
N	7.5	2.5	2.5	30.0	17.5	30.0	25.0	25.0
p	0.69	0.99	0.48	0.87	0.49	0.75	0.06	0.37

Table 4: The error rate (%) for AR/non-AR; p-values of Pearson’s chi-square test. n = A:41, N:40

Q8 showed the following program:



The students had to select one of four video clips that varied in the condition that could cause the robot’s LEDs to become red. The correct video showed red when *either* the front button was pressed *or* an obstacle was placed in front of the robot. Although these questions are relatively simple, the students must reason on the spatial and logical relations between sensing and acting. The significant improvement when using FB probably indicates that the FB caused the students to become more aware of these relations while experimenting with the robot.

Table 4 shows that the error rate is generally lower with AR than without, but the difference is not significant except for Q7 whose significance is borderline. In Q7, the students had to select one of four video clips that varied in the behavior of the robot when an object was placed in front of its sensors. The program caused the robot to turn right or left when an object was detected in front of the left or right sensors, respectively; when the object was detected by the center sensor, the robot moved forward.⁹ This question required understanding the relation between two event-actions pairs in sequence and the specific sensor events. We believe that seeing the execution of the event-actions pairs in context improved the understanding of these relations.

5.2 The usage data

To better understand the differences between treatment and control groups for the different conditions we investigated the usage data that we collected during the study. Figure 5 compares the median time between consecutive clicks on the run button in the VPL environment with the median number of actions between two consecutive runs, when using AR or not and when using FB or not. For AR, there is a significant difference between the treatment group and the control group. With AR, there were significantly fewer actions between the runs and significantly less time between clicks (Mann-Whitney U test, $p < 0.001$). When FB is given, there is no significant difference in the usage data of the treatment and control groups.

⁹The program and videos can be examined at <http://thymio.org/en:thymiopaper-vpl-iticse2015>.

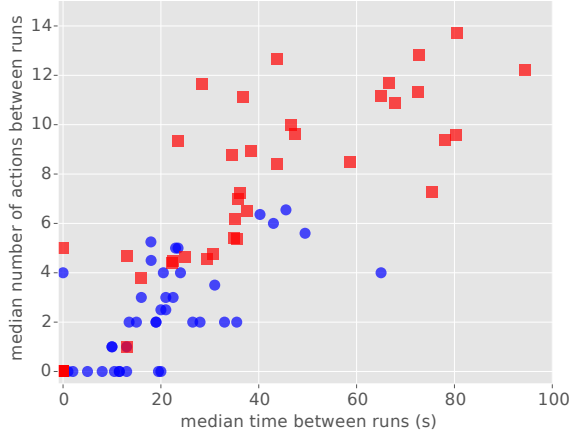


Figure 5: Comparison of usage behavior of students using AR (circles) or not (squares).

	treatment	control	p-value
Time between runs (s)			
Feedback	23.5 (n = 47)	24.8 (n = 34)	0.39
Augm. Reality	19.0 (n = 41)	36.0 (n = 40)	< 0.001
Action count between runs			
Feedback	4.0 (n = 47)	4.8 (n = 34)	0.45
Augm. Reality	2.0 (n = 41)	7.9 (n = 40)	< 0.001

Table 5: Median statistics of usage behavior and the p-value of Mann-Whitney U test.

A possible explanation for this difference is that using AR enabled students to identify possible errors in their programs quicker and more precisely, so they had a better understanding of the necessary steps and were addressing smaller problems at a time. Therefore, the more advanced learning environment led to the reduced reaction times and fewer actions between runs. Conversely, this difference could be interpreted as follows: the additional complexity introduced by the AR system caused stress for the students and prevented them from focusing on the programming task. In turn, this could have led to a trial-and-error behavior where the students tested different programs at random without understanding the underlying concepts. Further research is necessary to show which of these two hypotheses is correct, or find another reason.

5.3 The observations

In this section we present some issues that we observed during the sessions.

5.3.1 Design and implementation of the AR

- The groups with AR required intensive support because AR significantly increased the complexity of the setup. By intensive support we mean that students had more than two questions per hour in average and answering these questions required more than one minute.
- Some students found using the tablet as a debugger to be unintuitive.

- Several students tended to keep the tablet too close to the robot and therefore the tablet did not see the ground image.
- The use of the tablet was not uniform: some students did not use the tablet at all, while others seemed lost in contemplating reality through the tablet.
- The students did not always realize whether the tablet was tracking the ground or not.
- The software did not work uniformly well on different devices, especially those with different screen sizes.
- Energy consumption was a problem.
- The current setup of a computer running VPL while a separate tablet runs the AR is cumbersome.
- Some tablets have poor focusing abilities and sometimes stayed out of focus for several minutes.
- The AR system sometimes lost track of the robot.

These problems point out the technical and pedagogical difficulties of deploying AR in an educational context. Several technological difficulties can be solved by investing more effort in development. For example, VPL should run on the tablet so a computer is not needed, energy-saving algorithms should be implemented, as should algorithms for robust localization [12, Chapter 5]. The pedagogical difficulties point to the need for careful instruction on how to use AR and how to debug programs.

5.3.2 Implementation of the questionnaire

Feedback from colleagues and observations from a pilot use of the questionnaire led us to re-do the video clips. The original clips were taken with the robot *facing* the user and the camera. This makes it easy to see the horizontal proximity sensors, which were widely used in the questionnaire’s programs, but it required mental effort to interpret the directions “right” and “left” when they referred to the body of the robot. The video clips were photographed again, this time from the back of the robot. The advantage is that there is no need to mentally translate the directions; the disadvantage is that the sensors cannot be seen.

We found Google Forms easy to use, but there were two disadvantages: there is little flexibility in the format of the questions, and YouTube shows unrelated videos after each clip, which proved to be distracting to some students.

6. DISCUSSION

For all the questions in the questionnaire, students who received FB achieved lower error rates than those who did not receive FB. Similarly, students who used AR achieved lower error rates in seven out of the eight questions. However, the improved performance was only significant in a few cases. The results are therefore more encouraging than conclusive. They do seem to indicate that visualization such as that provided both by FB and AR can improve students’ spatial and temporal understanding of programs in the context of robotics.

The observations in Section 5.3 show that implementing AR is a difficult technical challenge. One has to take into account physical aspects such as the weight and position of the

tablet, as well as algorithmic aspects such as the localization, and interaction design aspects such as the user interface. It is also not surprising that intensive support and explicit instruction is needed if students are to obtain the maximum benefit from a sophisticated technology like AR.

We found the video-based questionnaire to be very successful; it allowed the students to answer the questionnaire in less time than with the textual questionnaire of [9]. A video questionnaire is more appropriate when studying young children as it does not require a high level of linguistic capabilities. Even when language is not a problem, we believe that video questionnaires should be used when asking about the physical behavior of robots. Free and easily available tools—Google Forms and YouTube—enabled us to quickly construct an adequate questionnaire, although more flexible software support is needed for an optimal experience.

7. LIMITATIONS OF THE RESEARCH

The large population ensures that the results are reliable, but the experiment was carried out in one location, at one time and using a specific robot and AR system, so it may not be generalizable.

While the technical difficulties described in Section 5.3 were not surprising in a first attempt to use AR in this context, they did cause the activities to be sub-optimal and possibly prevented AR from realizing its full potential.

The limitation of the questions to those at the analyzing level of the Bloom taxonomy means that the research focused on only one form of learning.

8. CONCLUSIONS

We carried out a quantitative study of the effect of visual FB and AR on the learning of a CS concept using a mobile robot. Visual FB had a significant positive effect on some questions, while AR had a positive effect but the improvement was not significant, although it did improve the students' performance. Together with our previous study [9], this research supports the claim that robotics activities are effective for teaching introductory CS at the K-12 level.

We described a new research tool—the video-based questionnaire—that is appropriate for investigating learning in young students.

We believe that this paper is the first report of a study of using AR to improve learning in CS education, and one of the small number of quantitative study on the use of AR in education in general [13]. We described some of the difficulties of using AR in the context of educational robotics activities. As next step, the AR hardware and software need to be made more robust and easy to use, and learning materials designed for AR must be developed. Then further research is needed to accurately characterize when FB and AR improve learning.

9. ACKNOWLEDGEMENTS

We thank Alessia Marra, Maurizio Nitti, Maria Beltran and Manon Briod for their artistic contributions to VPL and its augmented reality version. We also thank Prof. Matthias Hauswirth for allowing us to run the USI workshop, and Elisa Larghi for her help in organizing it. We thank the assistants of the USI workshop: Christian Vuerich, Matteo Morisoli, Gabriele Cerfoglio, Filippo Ferrario for

their dedication running the workshops. We thank the students attending the workshops for their cooperation. Finally, we thank the anonymous reviewers who provided useful feedback that improved the article. The research leading to these results has received funding from the European Union's Seventh Framework Programme under grant agreement n° 603662.

10. REFERENCES

- [1] A. L. Ambler, T. Green, T. D. Kumura, A. Repenning, and T. Smedley. 1997 visual programming challenge summary. In *Proceedings IEEE Symposium on Visual Languages*, pages 11–18, 1997.
- [2] T. N. Arvanitis, A. Petrou, J. F. Knight, S. Savas, S. Sotiriou, M. Gargalakos, and E. Gialouri. Human factors and qualitative pedagogical evaluation of a mobile augmented reality system for science education used by learners with physical disabilities. *Personal and Ubiquitous Computing*, 13(3):243–250, 2009.
- [3] L. W. Barsalou. Grounded cognition. *Annual Review of Psychology*, 59(1):617–645, 2008.
- [4] M. U. Bers, L. Flannery, E. R. Kazakoff, and A. Sullivan. Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72:145–157, 2014.
- [5] K. Bruce, A. Danyluk, and M. Thomas. *Java: An Eventful Approach*. Prentice Hall, 2006.
- [6] K. P. King and M. Gura, editors. *Classroom Robotics: Case Stories of 21st Century Instruction for Millennial Students*. Information Age Publishing, Charlotte, NC, 2007.
- [7] S. Magnenat and F. Mondada. Improving the Thymio Visual Programming Language experience through augmented reality. Technical Report EPFL-200462, ETH Zürich and EPFL, March 2014. <http://infoscience.epfl.ch/record/200462> (last accessed 16 November 2014).
- [8] S. Magnenat, P. Rétornaz, M. Bonani, V. Longchamp, and F. Mondada. ASEBA: A Modular Architecture for Event-Based Control of Complex Robots. *IEEE/ASME Transactions on Mechatronics*, PP(99):1–9, 2010.
- [9] S. Magnenat, J. Shin, F. Riedo, R. Siegwart, and M. Ben-Ari. Teaching a core CS concept through robotics. In *Proceedings of the Nineteenth Annual Conference on Innovation & Technology in Computer Science Education*, pages 315–320, Uppsala, Sweden, 2014.
- [10] O. Meerbaum-Salant, M. Armoni, and M. Ben-Ari. Learning computer science concepts with Scratch. *Computer Science Education*, 23(3):239–264, 2013.
- [11] F. Riedo, M. Chevalier, S. Magnenat, and F. Mondada. Thymio II, a robot that grows wiser with children. In *IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, 2013.
- [12] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza. *Introduction to Autonomous Mobile Robots (Second Edition)*. MIT Press, Cambridge, MA, 2011.
- [13] P. Sommerauer and O. Müller. Augmented reality in informal learning environments: A field experiment in a mathematics exhibition. *Computers & Education*, 79:59–68, 2014.