Introducing Computer Programming to Children through Robotic and Wearable Devices

Alexandros Merkouris
Ionian University
Department of Informatics
Corfu, Greece
c14merk@ionio.gr

Konstantinos Chorianopoulos
Norwegian University of Science and Technology
Trondheim, Norway
choko@acm.org

ABSTRACT
Learning to program in computer code has been considered one of the pillars of contemporary education with benefits that reach well beyond the skills required by the computing industry, into creativity and self-expression. Nevertheless, the execution of computer programs usually takes place on a traditional desktop computer, which has a limited repertoire of input and output interfaces to engage with the user. On the other hand, pedagogy has emphasized that physical representations and tangible interactive objects benefit learning especially for young students. In this work, we explore the benefits of learning to code for ubiquitous computers, such as robots and wearable computers, in comparison to programming for the desktop computer. For this purpose, thirty-six students participated in a within groups study that involved three types of tangibility at the target computer platform: 1) desktop with Scratch, 2) wearable with Arduino LilyPad, and 3) robotic with Lego Mindstorms. Regardless of the target platform, we employed the same desktop visual programming environment (MIT Scratch, Modkit and Enchanting) and we measured emotional engagement and assessed students' programming skills. We found that students expressed more positive emotions while programming with the robotic rather than the desktop computer. Furthermore, tangible computing platforms didn't affect dramatically students' performance in computational thinking.

Categories and Subject Descriptors
Social and professional topics → Computer science education; K-12 education; Student assessment; Computer systems organization → Robotics

General Terms
Experimentation, Human Factors

Keywords
Ubiquitous computing, embodiment, robot, wearable, learning, experiment, children

1. INTRODUCTION
The theory of constructivism supports that children learn by constructing their own understanding of the world when given active learning opportunities (e.g. experiments and real world problem solving). Papert [12] adds to this by introducing the concept of constructionism which suggests that learning occurs when students actively engage in the design and construction of a personally meaningful artifact [15]. Research in educational robotics is based on Papert's hands-on, experiential theory to a great extent. Robotic computing platforms have been proposed as a means to engage students with a particular focus on the Science, Technology, Engineering and Mathematics (STEM) curriculum [1, 11]. Besides the use of educational robotics, another application of constructionism in education, targeting mostly girls, is the use of computational textiles. E-textiles have been effectively used to introduce STEM sciences to students in a more appealing way. [3, 6, 13]. However, limited research has been conducted on effectiveness of target computing platform. Although there is previous research on writing computer code through tangibles and on the effectiveness of using these platforms in order to acquire new skills, there is limited evidence on students' attitudes and their level of engagement in the process of writing code for tangible computing platforms [6, 13].

Previous works in computer programming education for children have taken into consideration many parameters [7], such as visual programming environments (e.g., Scratch [14]), gender issues (e.g., Alice [8]), and pedagogy [5]. Nevertheless, there is limited consideration of the embodied dimension of learning, because most of the approaches are focused only on the cognitive aspect. Indeed, the desktop computer has been employed in most cases of computer education both as a programming tool and as the target for computer program execution. Although computer programming is a highly abstract and as a matter of fact cognitive activity, the learning of computer programming might be benefited if it is channeled through embodied mediums. In this work, we explore the benefits of teaching computer programming through embodied platforms, such as robotics and wearable computers.

Our research questions consider the following issues:

1. Is tangible computing more engaging than desktop computing in learning computer programming?
2. Are there differences between boys and girls with regard to the preference of a tangible platform?
3. Through which target platform, students can develop their programming skills more effectively?

2. RELATED WORK
According to the constructionist learning theory, children are better learners when they construct knowledge voluntarily, for a personally significant purpose, engaged in designing and creating visible objects such as computer programs, animations, robots and e-textiles [12]. For this purpose, they have developed computer programming environments and pedagogic strategies that favor the construction of knowledge through playing with real world metaphors or tangible objects. Nevertheless, there has not been any experimental evaluation of those theories and pedagogies in a real-world classroom.

WiPSCE '15, November 09-11, 2015, London, United Kingdom
© 2015 ACM. ISBN 978-1-4503-3753-3/15/11$15.00
DOI: http://dx.doi.org/10.1145/2818314.2818342

Robotic computing, such as LEGO Mindstorms, has been proposed as an inspiring framework for getting students involved with Science Technology Engineering Math (STEM) disciplines, as well as with computer programming [1, 11]. Benitti's systematic review of the research [1], conducted on the use of educational robotics, indicates that through educational robotics learners developed skills such as: (i) thinking skills, (ii) science process skills/problem-solving approaches, and (iii) social interaction/teamwork skills.

Computational textiles (e-textiles) toolkits, such as Buechley's LilyPad Arduino [3], although similar in many functional aspects to robotics construction kits, make use of soft materials instead of motors and gears, and incorporate crafting techniques such as sewing. E-textiles educational activities introduce other forms of expression, which historically have a more feminine orientation, therefore attracting a different population of students in engineering, programming and computer science [3, 6]. Qiu confirmed that using LilyPad to construct e-textiles can both draw attention to a diverse population and increase students' comfort, enjoyment and interest in working with electronics and programming [13].

Another important research issue concerns the attitudes of the students. Attitudes and perceptions of expected behavior, determine how a person is likely to act in different situations such as learning computer programming. Therefore, positive attitudes toward computers can increase computer use and understanding of emergent skills in young and older users [4]. According to Beisser [2] prior technological experiences affect attitudes towards computing. Furthermore, it [2] was found that the technological confidence of girls has benefited by visual programming environments. With respect to confidence, multiple studies have also found girls' comfort level increases with experience [16]. Therefore, it is important to evaluate computer programming systems in terms of student attitudes and intention to learn programming in the future.

3. METHODOLOGY

The goal of our research was to experimentally evaluate the comparative benefits of wearable and robotic computing as target platforms for learning to program. In addition we employed the desktop computing target platform as a point of reference. In each case, the visual programming environment was based on Scratch[1]. The attitudes and emotions of the students was measured with questionnaires before and after the use of the computing platforms. Furthermore computational thinking tests were applied for the assessment of students' programming skills [9].

3.1 Materials

The Desktop computer was employed in all cases as the development platform, but the program execution was performed on a different target platform, in order to reveal benefits that can be attributed to the type of tangibility (disembodied, textile, and robotic respectively). Firstly, as a point of reference, we used the desktop computer as a target platform and students developed with Scratch. For the wearable computing target platform, we employed the Arduino LilyPad system, which was programmed with the Modkit [10] visual programming environment. Finally, in the case of the robotic treatment the Lego Mindstorms platform was selected and the students developed their programs with the Enchanting visual programming environment. Both the Modkit[2] and the Enchanting[3] are similar to the Scratch one, so we can safely assume that the differences between the different target platforms are due to the types of tangibility (disembodied, textile, and robot respectively) and not due to differences in the developer workstation. In both cases of the tangible computing platforms, the main motivation for selecting the hardware was the availability of a Scratch-like visual programming environment on the desktop computer (Enchanting and Modkit respectively).

Table 1. We compare the benefits to learning computer programming on Different types of tangibility of the target platform.

Tangibility	Target platform	Development software
Disembodied	Desktop computer	Scratch 2.0
Robotic	Lego Mindstorms NXT	Enchanting
Wearable	Arduino LilyPad	Modkit

The creation of the instructional material was guided by the need to represent the same computational concepts (e.g., for loop) and a time constraint of forty-five minutes for each target platform. The instructional material consisted of two parts. In the first part, the students were asked to put together an object on the respective target platform (desktop, robotic, wearable), which was a virtual Christmas tree, a moving robot, and a messenger bag with leds. In each case, the students were provided with the basic elements of the object and instructions for construction. In the second part of the instructional materials, the students were asked to write a computer program for the object they put together in the first part. We focused on three basic computational: 1) sequence, 2) repeat, and 3) if-then-else, concepts [9] and we asked the students to use the above programing notions in order to bring more life into their creations from the first part. Moreover, the instructional material included code examples that demonstrated the use of the computational concepts. Regardless of the type of tangibility the students were asked to create very similar computer programs, at least in terms of code. A two-month pilot study refined the activities in order to make them as similar as possible in terms of visual programming despite the fact that the three target platforms have significant differences. Both the preparation of the instrumental material and the tutoring of the courses were conducted by the some researcher. In summary, as long as the only difference in the instructional materials is the target platform, we can expect that any difference in attitude or emotion should be attributed to the treatment.

3.2 Subjects

The participants of the study were thirty-six students (eighteen girls and eighteen boys), which were randomly selected from the first grade class (between twelve and to thirteen years old) of a secondary education school. Three subgroups, with 6 boys and 6 girls each, were created: the *Disembodied – First*, the *Wearable – First* and the *Robotic – First*. The order that each subgroup dealt iteratively with each target platform was randomized in order to minimize the learning effects of the within groups design. No student had received teaching in computer programming as part of previous formal education, but we also employed a demographic questionnaire in order to record previous computing experiences.

[1] Scratch: http://scratch.mit.edu/

[2] Modkit: http://www.modkit.com/

[3] Enchanting: http://enchanting.robotclub.ab.ca/

3.3 Measuring Instruments and Data Analysis

The pre-tests before the computer programming activities consisted of a four-level Likert questionnaire that recorded their previous experience and attitude towards computers, coding, robotics, and electronics.

The post-tests after the computer programming activities included a five-level Likert questionnaire according to the following semantic differentials emotions: happy-sad, confused-confident, boring-interesting, disappointed-satisfied, undetermined-determined. They also included computational thinking examination, with 9 multiple choice and 3 gap filling questions, on the programming concepts investigated during the study. The data were collected through online questionnaires and tests and then were analyzed with SPSS.

3.4 Procedure

Firstly, the students filled-in the pre-tests that recorded their demographics and attitudes at their own convenience. In the beginning of the experiment the students were informed that they were going to participate in a voluntary activity about computing and that the exams of the activity do not count towards the grade of their normal course of study. Students worked in (same-gender) pairs on each one of the activities, but answered the questions of the post-tests individually. The emotion post-test was filled-in first, it was followed by students' programming skills assessment. On different days (within the same week) the students were following the same procedure for the second and the third treatment of experiment.

4. RESULTS

4.1 Robotic computing is more exciting than desktop computing

According to the mean averages in Figure 1 it appears that students prefer robotic computing in all five categories of emotions under investigation. In addition, wearable programming has been found to provoke more positive emotions than desktop computing.

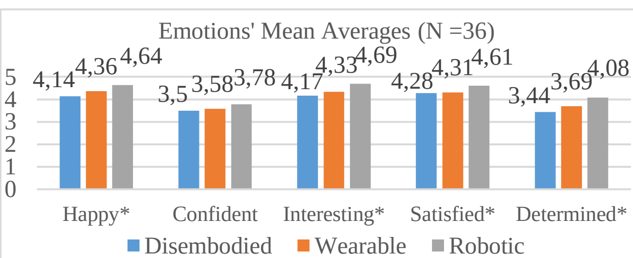

Figure 1. Emotions' mean average segregated according to emotions and target platforms.

The one way within – subjects' analysis of variances (ANOVA) was applied, in order to verify whether there was a significant statistical difference between the students' emotions. The results indicated that there was indeed significant difference in four out of five emotional categories. More specifically in categories: **happy** (Wilks' Lambda = 0.745, F(2, 34)=5.81, p=.007 < .05), **interesting** (Wilks' Lambda = 0.766, F(2, 34)=5.183, p=.011 < .05), **satisfied** (Wilks' Lambda = 0.766, F(2, 34)=5.183, p=.011 < .05) and **determined** (Wilks' Lambda = 0.577, F(2, 34)=12,466, p=.000 < .05). In the post – hoc analysis, the paired sampled t-test indicated that students felt happier and more interested, satisfied and determined with robotic computing than with desktop computing.

4.2 Tangible Computing and Students' Performance

According to the correct answers' percentage (Figure 2), it is obvious that in the **sequence** computational concept, students performed better in wearable and robotic computing than in desktop. In the **repeat** concept there were no major differences while in the **If – else** notion students performed better in desktop computing. Finally in the **extended program** node students gave more correct answers in robot computing.

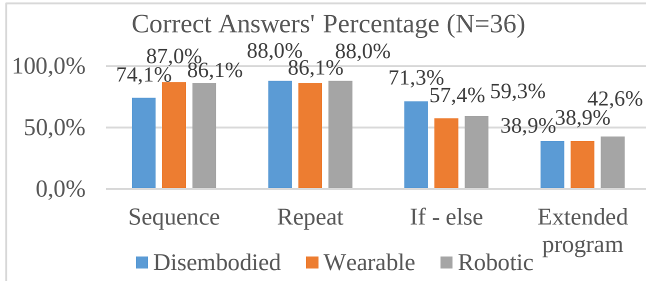

Figure 2. Correct answers' percentage segregated according to computational concepts and the target platforms.

Inductive statistical analysis showed no significant difference in students' performance in all computational concepts regardless of the target platform. It can be therefore inferred that the tangible computing platforms, employed in this survey, did not affect dramatically the student's performance in programming.

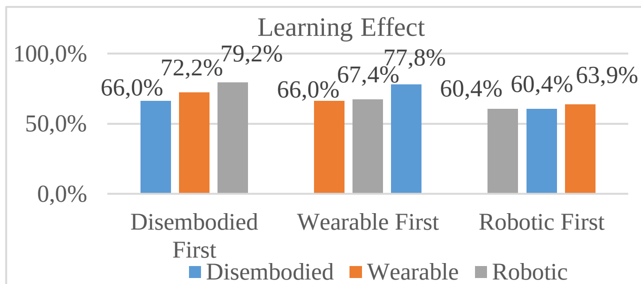

Figure 3. Correct answers' percentage segregated according to order of treatment and the target platform

Moreover, since a within groups experiment was applied in our research, as it was expected, students' programming skills improved after each programing activity. According to Figure 3, *Disembodied – First* and *Wearable – First* students' subgroups had the highest and smoothest improvement in their performance. Surprisingly, *Robotic – First* students' subgroup, showed minor improvement on their programming skills, although a similar smooth increase in the performance was anticipated. It seems that using robots as the introducing target platform had a neutral learning effect. Nevertheless, further investigation is required to verify this finding.

4.3 Gender and Tangible Computing

With regards to the emotions boys reported to have experienced, ANOVA test revealed significant difference in categories: **confident** (Wilks' Lambda = 0.641, F(2, 16)=4.477, p=.029 < .05) and **determined** (Wilks' Lambda = 0.418, F(2, 16)=11.124, p=.001 < .05). In the case of girls, the difference was found in the following categories: **happy** (Wilks' Lambda = 0.482, F(2, 16)=8.591, p=.003 < .05) and **determined** (Wilks' Lambda = 0.653, F(2, 16)=4.249, p=.033 < .05). The post hoc analysis showed that more positive emotions were reported in robotic computing than in the desktop for both boys and girls.

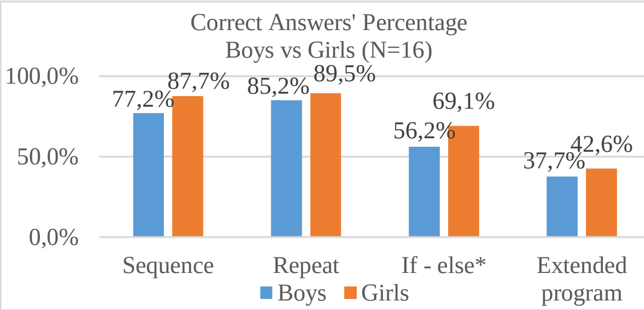

Figure 4: Correct answers' percentage boys vs girls segregated according to computational concepts and the target platforms.

According to the students' responses analysis in the computational thinking tests, girls performed better in all programming concept categories in this study. Statistical difference between boys' and girls' performance was confirmed by the independent samples T-test in the **If – else** programming notion; $t(106) = -2.109$, $p = 0.037$. Unexpectedly, these results demonstrated that boys did not acquire more programming skills than girls.

5. CONCLUSION

The results of the experiment have confirmed that learning computer programing with ubiquitous target platforms is more effective than working only on the desktop. In particular, students expressed more positive feelings towards the robotic computing treatment. Although the wearable computing treatment has not been as favorable as the robotic one, it has been preferable to the desktop target platform. One possible explanation is that the wearable computing treatment in this experiment is based on the LilyPad platform, which is not as refined as the Lego Mindstorms one. Therefore, further research should consider again the wearable computing treatment with a more malleable implementation of a wearable target platform. Moreover, the results of the experiment have shown that there was no gender difference in the interest toward the type of the ubiquitous computing platform. Previous research has promoted wearable computing platforms as more engaging for girls (e.g., [2, 13]), but our findings indicate that girls were emotionally engaged in robots as much as boys do. In addition, it was found that girls were more interested in robotic than wearable computing. In further research, it is intended to repeat the experiment with other groups of students and additional activities following the student initiative in order to confirm our findings.

6. REFERENCES

[1] Fabiane Barreto Vavassori Benitti. 2012. Exploring the educational potential of robotics in schools. *Comput. Educ.* 58, 3 (April 2012), 978-988.

[2] Beisser, S. R. (Jan 2006). An Examination of Gender Differences in Elementary Constructionist Classrooms Using Lego/Logo Instruction. Computers in the Schools, 22, 7- 19.

[3] Leah Buechley, Mike Eisenberg, Jaime Catchen, and Ali Crockett. 2008. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '08). ACM, New York, NY, USA, 423-432.

[4] Polina Charters, Michael J. Lee, Andrew J. Ko, and Dastyni Loksa. 2014. Challenging stereotypes and changing attitudes: the effect of a brief programming encounter on adults' attitudes toward programming. In *Proceedings of the 45th ACM technical symposium on Computer science education* (SIGCSE '14). ACM, New York, NY, USA, 653-658.

[5] Stephen Cooper, Wanda Dann, and John Harrison. 2010. A k-12 college partnership. In *Proceedings of the 41st ACM technical symposium on Computer science education* (SIGCSE '10). ACM, New York, NY, USA, 320-324.

[6] Yasmin B. Kafai, Eunkyoung Lee, Kristin Searle, Deborah Fields, Eliot Kaplan, and Debora Lui. 2014. A Crafts-Oriented Approach to Computing in High School: Introducing Computational Concepts, Practices, and Perspectives with Electronic Textiles. *Trans. Comput. Educ.* 14, 1, Article 1 (March 2014), 20 pages.

[7] Caitlin Kelleher and Randy Pausch. 2005. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. ACM Comput. Surv. 37, 2 (June 2005), 83-137.

[8] Caitlin Kelleher, Randy Pausch, and Sara Kiesler. 2007. Storytelling alice motivates middle school girls to learn computer programming. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07). ACM, New York, NY, USA, 1455-1464.

[9] Colleen M. Lewis. 2010. How programming environment shapes perception, learning and goals: logo vs. scratch. In *Proceedings of the 41st ACM technical symposium on Computer science education* (SIGCSE '10). ACM, New York, NY, USA, 346-350.

[10] Amon Millner and Edward Baafi. 2011. Modkit: blending and extending approachable platforms for creating computer programs and interactive objects. In *Proceedings of the 10th International Conference on Interaction Design and Children* (IDC '11). ACM, New York, NY, USA, 250-253.

[11] Omar Mubin, Catherine J Stevens, Suleman Shahid, Abdullah Al Mahmud,and Jian-Jie Dong. A review of the applicability of robots in education. *Journal of Technology in Education and Learning*, 1, 2013

[12] Papert, S. *Mindstorms: children, computers, and powerful ideas*. New York, NY: Basic Books, 1980.

[13] Kanjun Qiu, Leah Buechley, Edward Baafi, and Wendy Dubow. 2013. A curriculum for teaching computer science through computational textiles. In *Proceedings of the 12th International Conference on Interaction Design and Children* (IDC '13). ACM, New York, NY, USA, 20-27.

[14] Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, and Yasmin Kafai. 2009. Scratch: programming for all. *Commun. ACM* 52, 11 (November 2009), 60-67.

[15] Resnick, M.; Martin, F.; Sargent, R.; Silverman, B., "Programmable Bricks: Toys to think with," *IBM Systems Journal* , vol.35, no.3.4, pp.443,452, 1996

[16] Snyder, T. D. (2014). Mobile Digest of Education Statistics, 2013 (NCES 2014-085). Washington, DC: National Center for Education Statistics, Institute of Educational Sciences, U.S. Department of Education.