TanPro-Kit: a Tangible Programming Tool for Children

Danli Wang[1,2], Yunfeng Qi[1,3], Yang Zhang[1], Tingting Wang[1,3]
1. Institue of Software, Chinese Academy of Sciences, Beijing, China
2. Beijing Key Lab of Human-Computer Interaction, Beijing, China
3. Graduate University, Chinese Academy of Sciences, Beijing, China
{danliwang2009,qiyunfeng123, bennybeihang, kristin.wtt}@gmail.com

ABSTRACT
This paper describes a new tangible programming tool--- TanPro-Kit which was designed for children aged 5 to 9. It consists of programming blocks and a LED pad. The LED pad presents visual animations and audible feedback according to the arrangement of blocks with which children make program to play a maze game. Aiming at lowering the cost of TanPro-Kit, we adopted LED, RFID, wireless and infrared technology to develop the whole system. The system acquires the programming blocks' physical information which is then translated into programming semantic. TanPro-Kit is low-cost, which is more acceptable in developing countries. We ran a user study with 16 children involved, which showed TanPro-Kit to be attractive to children and easy to learn and use.

Categories and Subject Descriptors
K.3.1 [Computers and Education]: Computer Uses in Education, H.5.2 [Information Interfaces and Presentation]: User Interfaces

General Terms
Design, Experimentation, Human Factors

Keywords
Tangible Programming, Children, Programming Languages, Learning.

1. INTRODUCTION
Research by Resnick et al. indicates that computer programming intervention in children's education may help them develop a deeper understanding of how computers work and support their development of "computational thinking", providing experience with important problem-solving and design strategies [6]. Early studies with Logo also showed that when introduced in a structured way, computer programming can help children improve visual memories and basic numerical senses, as well as develop language skills [2]. Therefore, lowering the barriers to programming and offering children proper programming tools is highly valuable in education.

Most of the existing programming tools, however, are based on texts and symbols which are difficult for children to understand [4, 7]. Fortunately, tangible interface and graphical interface provide layered scaffolding to children as they progress toward increasingly authentic programming environment. Horn described the programming learning process of children as starting with a tangible system and then transiting to a graphical system with increased capabilities and complexity [3]. And tangible programming is thought to be a developmentally appropriate practice for young children [1]. Compared with operating computer directly by using keyboard and mouse, using physical objects to interact with computer is much easier to involve children in the process [5]. Children can write programs by assembling the physical objects without keystrokes [5, 3].

There is some prominent work in tangible programming, which enlightens us a lot. For instance, *Electronic Block* [10] uses building blocks to program. It consists of three types of building blocks: sensor block as input; logic block to conduct logic computation; behavior block as output. It is designed for preschoolers so that the syntax is simple, easy to manipulate and free from spatial limitation. *Tangible Programming Bricks* [5] proposed by MIT communicates with the computer via a card slot, with PIC microprocessor built in each physical programming brick. Children put the bricks into the card slot in different sequences to control the game objects like toy trains and household appliances. This work is highly functional, containing lots of different programming concepts. *Tangicons 3.0* [8] is an educational game for children, which focuses on developing children's abstract thinking and collaborative ability. It allows children to solve problems together by manipulating physical objects to move virtual characters on a map on the computer screen. Its programming blocks are built with Sifteo cubes, which cost much. *Tern* [3] is a tangible programming language with which children assemble puzzle pieces to express certain meanings. After finishing programming, children need to manually use a camera to capture the sequence's image which is transferred to computer to control virtual characters or a real walking robot. *T-Maze* [9] also uses cameras to identify physical programs. Children play maze games shown on a computer screen by assembling tangible blocks in sequences. Real-time feedback is presented on the screen to show the path children have programmed for the characters in the game.

Though many good tools such as the works above were presented, there still exist improvements in this field. First, some programming tools are based on computers, which require certain conditions of usage. Second, the high cost of programming blocks and feedback devices including computer screen is not affordable in some kindergartens in developing countries.

Trying to solve the problems, we developed a low-cost programming tool---TanPro-Kit, which uses LED pad as feedback device (Figure 1). TanPro-Kit's games are in two levels: the introductory level, and the further level. It enables two modes of feedback, focusing on helping children learn programming as beginners. Besides, it does not have special requirements on the usage condition, which is convenient for its wide application in kindergartens.

Figure1. Children with TanPro-Kit

2. IMPLEMENTATION

TanPro-Kit is composed of two parts: the programming blocks and the LED pad (see Figure 2).

Figure 2. TanPro-Kit

2.1 Programming Blocks

Programming blocks send LED pad their physical information which is then translated into the program semantics. In TanPro-Kit, there are three kinds of programming blocks: start and end block, direction block and sensor block. Each block has a figure on the top surface and a text prompt at the top right corner of the surface. The left and right surfaces of the blocks are attached with magnets along with symbol graphics representing “positive” and “negative” poles respectively. They are designed to help children connect blocks correctly. As Figure 3 shows, each block has a single chip microcomputer (SCM), an infrared transmitter and receiver module, and a wireless module. When a new block is added to the sequence, the prior block’s infrared signal will activate the new added one which will then send its identity code to LED pad. At the same time, it will start its own infrared transmitter, being ready for triggering next added programming blocks.

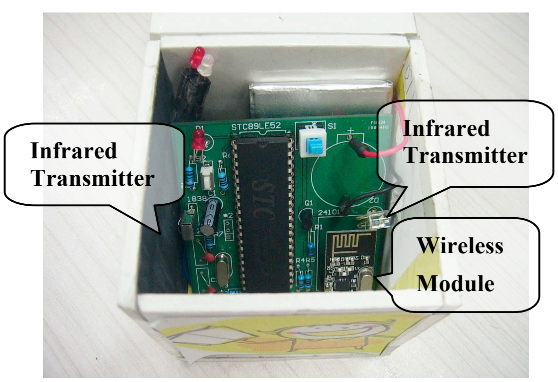

Figure 3. SCM in block

2.2 LED Pad

The LED pad implemented with SCM, RFID, sensors, wireless module and LED matrix, is the main feedback part of the whole system. It is composed of two parts: the Maze Game and the LED box (see Figure 4).

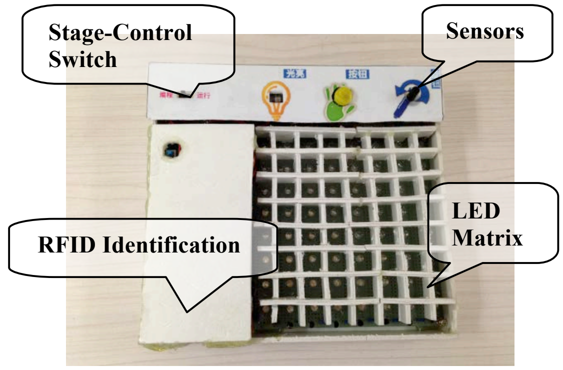

Figure 4. LED box

2.2.1 Maze Game

The Maze Games are generated from the printed maps with RFID tags attached on the lower-left corner of the back. Different RFID tags attached on different maps are used for LED box to identify the maze map and provide appropriate feedback accordingly. After putting a map on the LED box directly, the tag just covers the RFID receiver in the box, and sends information of the maze map to the SCM module, which then generates a virtual maze game ready to be played. The maze is composed of four kinds of cells: start cell, end cell, normal cell and sensor cell. The maze games are in two levels: the introductory level with real-time feedback both on programming stage and running stage; and the further level with real-time feedback in running stage only. For the programming on introductory levels, the LED lights along the path are lightened as the programming blocks are added. Once a wrong block is added, a “cross” meaning wrong will flicker in the LED matrix. While on the further levels, no feedback appears in the programming stage. For both of the two levels, the feedback in running stage is the same.

2.2.2 LED Box

The LED box mainly contains four parts: the stage-control switch, the RFID identification area, the sensors and the LED matrix (see Figure 4). The stage-control switch is used to switch between the programming stage and the running stage. The RFID reader at the top of the RFID identification area identifies the tags of different maze maps, so that children can change levels by simply putting a different paper map on the LED box. In running stage, the light in the LED matrix which occupies the start cell of the maze map is turned on, and other lights are turned on one by one, along the path constructed by tangible blocks. Thus, an animation of a moving light spot is presented to children to show the execution of their programs. When the light spot moves to the sensor cells, children need to trigger the relative sensors on the upper-right of the LED box, to keep the execution going.

There are three kinds of sensors as Figure 4 shows: light sensor, tangible button sensor and rotation sensor. The LED matrix consists of an 8×8 light matrix. The matrix can produce 2^{64} different patterns, which provides various visual outputs of the maze game.

3. USE CASE

In programming stage, firstly children need to find and program a path for the character to escape from the maze. Then, switch to running stage, and run the program constructed by tangible blocks, during which they may need to trigger sensors when the light spot is blocked at a sensor cell. In this part, a simple use case (see Figure 5) will be described.

Figure 5. A simple use case

3.1 Programming Stage

After choosing a maze map and putting it on the surface of LED box, children can start the programming stage by setting the stage-control switch on the LED Pad to the left. Children must place the start block as the first block of the program sequence. Then put the appropriate direction blocks and sensor blocks to construct a block sequence which indicates the path that the virtual character will go along. On the introductory levels, the LED lights along the path are turned on in real-time to indicate every move (see Figure 5). The sensor blocks will be needed when the path stretches to sensor cells on the map. If children place a wrong block in the sequence, a "cross" symbol composed of the LED lights will flicker in the LED matrix. The maze maps between introductory levels and further levels are basically the same. But on further levels, no real-time feedback is presented in the programming stage, even if children place a wrong programming block. Finally, an end block should be placed at the end of the program sequence, which will trigger a piece of music indicating the accomplishment of the programming stage.

3.2 Running Stage

For both introductory and further levels, when children finish the task in programming stage, they need to set the stage-control switch to the right to enter the running stage. The light spots will reduce to one beneath the start cell of the map which stands for the character. It starts to move along the path which is programmed already. When the spot encounters a sensor cell, it will stop until children trigger the relative sensor. If the spot goes to the end cell of the maze, a check mark composed of green LED lights will appear and another piece of music will ring.

4. USER STUDY

4.1 Goals of User Study

The user study was designed for answering two following questions in order to evaluate TanPro-Kit:

- Whether TanPro-Kit is interesting, easy to learn and use.
- Whether children can learn concepts of programming by playing TanPro-Kit.

4.2 User Study Process

16 children (8 girls and 8 boys) aged 5 to 9 were involved in our research. We invited the participants to come separately. The study was conducted in a classroom of the kindergarten, where children felt more at ease and comfortable. We set up a video recorder on a tripod to capture children's behaviors and voice.

We introduced the TanPro-Kit to the protagonist child and demonstrated how to play it. Children were also invited to try programming by themselves. After the practicing, children played two introductory levels and two further levels in order. Two levels of maps were equally difficult, while the feedback mechanisms were not the same. Right after a child finishing the tasks, he/she was invited to finish a questionnaire and then interviewed by our researcher.

The questionnaire used the Likert-type scale on four questions with scores from one to five, one being the worse and five the best. The questions are about whether TanPro-Kit is playful, easy to learn and use. During the interview, children were asked three questions. Two were designed to test whether they can learn basic programming concepts after playing TanPro-Kit. The other one was a routine question "are you willing to play it again next time".

At last, we analyzed the videotapes, questionnaires and interview answers to evaluate TanPro-Kit.

4.3 Results Analysis

For the two questions raised in the Goals of User Study, we analyzed all the collected data and got the results as follows.

4.3.1 Playfulness, Ease of Learn and Use

The four questions in questionnaire were all given comparatively high scores. The one-sample t-test between the mean of the scores and the median of the 5-scale has significant difference, which shows a positive affirmation of TanPro-Kit.

For the playfulness, two questions of the questionnaire show that TanPro-Kit is acceptable by and attractive to the children. "Is it interesting to control light spots by placing blocks" got an average score of 4.69($SD=0.46$, $p<0.001$) and "how do you like the game" got 4.63($SD=0.48$, $p<0.001$). According to the videotapes, the music at the end of programming and running stage made children excited. They were happy, yelling "great" or clapping their hands, when seeing their program successfully guide the character out of the maze. In interview, when asked whether they want to play the game again, 15 children (93.75%) answered "yes". The child who did not want to play again was 5 year old, who showed some scatterbrained personality. She showed great interests in every first test of each level, but lost them soon when playing the second test.

For the ease of learn and use, other two questions of the questionnaire about "easy to understand" and "easy to play" scored averagely at 4.56 ($SD=0.61$, $p<0.001$) and 4.50 ($SD=0.61$, $p<0.001$). The result shows that children can understand the game and operate the tangible blocks easily. Though the magnets force between blocks resulted in some occasional inconvenience in younger children's manipulation, it did help a lot when they were programming. We found in the videotapes that when children found there is no attraction response after they added a block, they checked the programming block carefully and rotated it until the correct side of the block was connected. The changing of different levels of games is also easy to learn. After seeing the demonstration, children eagerly wanted to try to put a map on the box by themselves. And they did the changing without any help during the test. The switch of the programming stage and the running stage is also successfully mastered by children.

Together, the results from questionnaires and videotapes show that the tool is interesting, easy to learn and use for children.

4.3.2 Children's Understanding of Programming Concepts

In the interview, we asked two questions to evaluate how many programming concepts children could probably acquire from TanPro-Kit. The first question was to let children explain the two stages of the game and what they were doing during the stages. The second question was to ask children to recall the programming blocks they used and explain their functions. All children successfully explained the first question. In their words, the programming stage was to "find a path" and the running stage was to "watch the light spot moves along the path" and helped the little spot get through when it was stuck at a sensor cell by coving, twisting or pressing the sensors on the box. 11 children (69%) remembered all the programming blocks and the rest remembered most of them. They described the functions of direction blocks and sensor blocks as giving the light spot orders to move to the desired directions.

In the analysis of videotapes, we found that children removed the last added block once they found the "cross" start to twinkle on the introductory levels. On the further levels, children switched between programming and running stage to modify their programs until the light spot finally reached at the end cell. Most children switched between stages only once before they succeeded.

Based on the analysis above, we could conclude that children can learn basic programming concepts by playing TanPro-Kit.

5. CONCLUSION AND FUTURE WORK

In this paper we described a new tangible programming tool---TanPro-Kit, which was designed for children aged 5 to 9. It consists of tangible programming blocks and a LED pad, which can be placed anywhere in the kindergarten or at home, with no space limitation. During the development of TanPro-Kit, we focused on the playfulness, learnability and handleability. In addition, we cut the development cost of the tool in order to make it affordable for children in the developing countries. We conducted a user study with 16 children involved and the results show that TanPro-Kit is attractive to children and easy to learn and use. Besides, children can get a preliminary understanding of programming by playing TanPro-Kit.

In the future, the tool must be improved to be more stable, which was a problem we found in the user study. And programming blocks in smaller size and with more functions such as "loop" and "variable" will be designed, which would enable TanPro-Kit to create a wider variety of application scenarios. In addition, we want to explore appropriate debugging methods for further levels. Also we will focus on how to provide appropriate feedback for children's learning. Finally, the improved system should preferably be able to support collaborations.

6. ACKNOWLEDGMENTS

We gratefully acknowledge financial support by the Major State Basic Research Development Program of China under Grant No. 2013CB328805, the National Natural Science Foundation of China under Grant No. 60970090 and No. 61272325, the Frontier Project of the Knowledge Innovation of Chinese Academy of Sciences under Grant No. ISCAS2009-QY03, and the Cooperation Project of Chinese Academy of Sciences and Foshan city under Grant No.2012YS04. We would like to acknowledge the support of Liang He, Ziyang Yuan, Muyan Li. Also we thank the teachers and children in Kindergarten that took part in the experimental study.

7. REFERENCES

[1] Bers, M.U., Horn, M.S. 2009. Tangible programming in early childhood: revisiting developmental assumptions through new technologies. In *High-tech tots: childhood in a digital world*, I.R. Berson, M.J. Berson, Ed. Information Age Publishing, Greenwich, 49-70.

[2] Clements, D. H. 1999. The future of educational computing research: The case of computer programming. *Information Technology in Childhood Education Annual*, *1999*, 1, 147-179.

[3] Horn, M.S., Crouser, R.J., Bers, M.U. 2012. Tangible interaction and learning: the case for a hybrid approach. *Personal and Ubiquitous Computing*. 16, 4 (April. 2012), 379-389.

[4] Kelleher, C. Pausch, R. 2005. Lowering the Barriers to Programming: A Taxonomy of Programming Environments And Languages for Novice Programmers. *ACM Computing Surveys (CSVR)*. 37, 2 (June. 2005), 83-137.

[5] McNerney T.S. 2004. From turtles to Tangible Programming Bricks: explorations in physical language design. *Personal and Ubiquitous Computing*. 8, 5(September. 2004), 326-337.

[6] Resniek, M., Flanagan, M., Kelleher, C., MacLaurin, M., Ohshima, V., Perlin, K., Torres, R. 2009. Growing Up Programming: Democratizing the Creation of Dynamic, Interactive Media. In *Proceeding of the CHI '09 Extended Abstracts on Human Factors in Computing Systems* (Boston, USA, April 04-09, 2009).CHI EA '09. ACM, New York, NY, 3293-3296.

[7] Revelle, G., Zuckerman, O., Druin, A. and Bolas, M. 2005. Tangible User Interfaces for Children. In *Proceeding of the* CHI '05 Extended Abstracts on Human Factors in Computing Systems (Portland, USA, April 02-07, 2005). CHI EA '05. ACM, New York, NY, 2051-2052.

[8] Scharf, F., Winkler, T., Hahn, C., Wolters, C., Herczeg, M., 2012. Tangicons 3.0: An Educational Non-Competitive Collaborative Game. In *Proceedings of the 11th International Conference on Interaction Design and Children* (Bremen, Germany, June 12-15, 2012). IDC '12. ACM, New York, NY, 144-151.

[9] Wang, D.L., Zhang, C., Wang, H.A. 2011. T-Maze: a tangible programming tool for children. In *Proceedings of the 10th International Conference on Interaction Design and Children* (Ann Arbor, USA, June 20-23, 2011). IDC'11. ACM, New York, NY, 127-135.

[10] Wyeth, P., Purchase, H. C., 2003. Using Developmental Theories to Inform the Design of Technology for Children. In *Proceedings of the 2003 conference on Interaction design and children* (Peston, UK, July 03, 2003). IDC '03. ACM, New York, NY, 93–100.