Programming Lessons for Kindergarten Children in Japan

Takeshi Watanabe, *watanabe@viscuit.com*
The University of Electro-Communications, LLC. Digital-pocket, Japan

Yuriko Nakayama, *nakayama@uec.ac.jp*
Kagawa-fujimigaoka Kindergarten, Japan

Yasunori Harada, *hakase@viscuit.com*
LLC. Digital-pocket, Japan

Yasushi Kuno, *y-kuno@uec.ac.jp*
The University of Electro-Communications, Japan

Abstract

Viscuit is a programming language developed by the one of the author. Distinguishing feature of Viscuit is that the programs are made of pictures only, and no characters are required to understand or make programs. This feature makes it possible to experience programming for pre-school children. Actual, LLC. Digital-pocket cooperates with Kagawa-fujimigaoka kindergarten in Japan for regular Viscuit classes since November 2015. In this poster, we analyzed children 's programs and videos taken during the lessons held for them in 2017. In addition, we took questionnaire on the difference between the usual state and the state of the Viscuit lessons of the children to the teacher in charge of each class, and examined the features of the programming lesson at the kindergarten.

Keywords

keyword; programming for kindergarten; visual programming language; Viscuit

Preface

Background

In Japan, programming education is going to start from elementary school year 2020. As the Ministry of Education of Japan states, the aims for programming education in elementary school is not to acquire coding skills, but to acquire logical thinking abilities and problem solving skills or computational thinking skills (Wing, 2006). As for the programming experiences, children are expected to organize or combine primitive actions or symbols to construct the movement they want. As the result of a survey in 2016, the number of Japanese private programming classes are increasing, along with the children's age decreasing. Particularly, programming workshops or classes for preschool children are getting popular, perhaps parents are willing for their children to be get ready for elementary schools programming classes.

We observed that children who joined full-year lessons during 2016 had apparently changed in their programming abilities. And programs made by them became more complex and rich. In this poster, we show results from our analysis, along with actual programs made by children.

What is Viscuit

Viscuit is a programming language created by Yasunori Harada who is one of the authors, in Japan 2003 (Harada & Richard, 2003) (Harada, 2010), which is a rule-based visual programming language like KIDSIM (Smith, Cypher, & Spohrer, 1994). The most distinguish character of this language is user can make program without using any letters or numbers, but only with drawing. On Viscuit, you can make programs by putting your drawing on the rules that are similar to glasses (actually we call those rules glasses). Right lens of the glasses means "before" situation and left means "after" situation. If you put a picture upper on the right lens than picture that is in left, a picture on the stage get to move up direction.

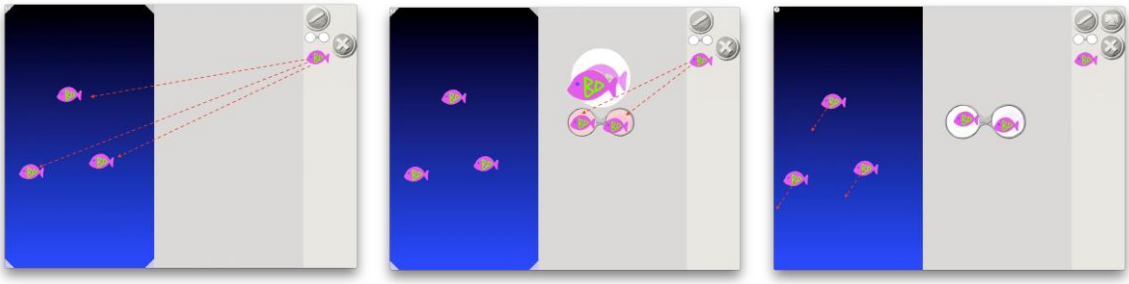

Figure 1. How to make program by Viscuit

Viscuit has a function of sharing program in a group. This function is called “Viscuit Land”. Programs created on each student’s tablet are gathered at another tablet through the network. For example, if the theme of a lesson is “sea”, participants will draw something that moves in the sea, make drawings move, send network and share programs in a group. In kindergarten, we have lessons using group work, not individual production.

Figure 2. Recital time by watching “Viscuit Land”

Kagawa-fujimigaoka Kindergarten

Programming lessons are held by kindergarten’ teacher. Nakayama, who is also one of the authors, is in charge of Viscuit lessons. Initially, the equipment has been brought by LLC. Digital-pocket, but then they bought iPad mini more than 30 units, also prepared Wi-Fi environment. So now, lesson has been carried out using all kindergarten equipment.

Purpose

In this research, we will clarify the growth of children that is shown in the continuing lesson of kindergarten. Also, clarify what preschool children are learning through programming lessons. Specifically, their programs that run in the computer are made as whether or not they wish. Also, we will clarify the features and important points of programming practice in kindergarten.

Research Method

Subject

The subjects are 56 children (5 and 6 years old) (28 people, 2 classes). In addition, a teacher in charge of each class is conducted a questionnaire survey on the state of the child at the end of each lesson. In the questionnaire, we asked the teacher how different between the child 's usual state and the state of the programming lesson, and who shows characteristic behaviour in the programming lesson.

Content of Lessons

There are some challenges for children to study advance programming every lesson. One lesson is 40 minutes. Lessons consist of three parts. Two practice tasks are carried out in the first half, and a free program production concerning those tasks follows. In the practice tasks, we prepare pictures for children to focus on their tasks. In the free production time we let children draw as they like unless it has relation to the task. Lastly, we have recital time that we appreciate what they make in groups on the screen like Figure 2.

On the other hand, it was found that even if the level of lesson did not raise every lesson, it could be enjoyed enough just by changing the picture of the practice. So, some important parts of programming are repeated again and again by changing the picture to fix the study without progress.

The layout of the classroom is like Figure 3. In this layout there are clearly separate "teaching space" and "production space". And by only placing the tablets in "production space" and do not allow children to bring tablets in "teaching space", we can make children listen to what the teacher says in the "teaching space". When the teacher teaches something to children, they are gathered in “teaching space”, and after they listen what they should to do, then go “production space”. This will allow children who cannot concentrate when the tablet is in front of them make concentrate on what the teacher says. This way of going and coming is only done at the time of the practice. When they work on free program, they remain “production space” and focus on what they want to make.

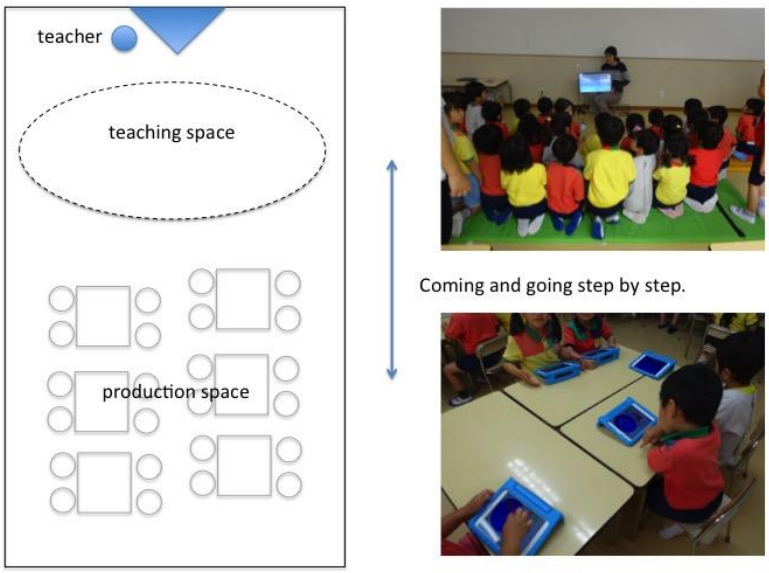

Figure 3. The layout of the classroom

Analysis

Analysis of program

From programs saved at each time, we counted the number of works saved, the number of pictures drawn, the number of pictures on the stage, the number of glasses. And we told whether those glasses were valid or not, whether the direction of drawings and movement matched and whether the task and the movement of the pictures they made matched.

Analysis of video

I felt it difficult to understand what the children were learning with only quantitative data of programs. Therefore, we executed video research. We chose 4 characteristic children and shot and analyzed the situation of free tasks in video. As a result of observing the video, it turns out that each of the four people shows different changes and attitude.

Figure 4. Lesson in kindergarten. Children make programs with their fingers.

Inquiry of teachers

Regarding the questionnaire of each class teacher, we got a glimpse of the difference between the ordinary activity of the children and the programming lesson using Viscuit. In the free description of questionnaire the following comments came out. ”Children's aggressiveness is high.” “I think in this lesson we provide a different experience from the drawing on paper.” “ I was surprised that they had been able to maintain their concentration for 40 minutes.” These are suggesting hints to clarify the features of programming lessons at kindergarten.

Conclusion

Now we have done analysis of the program of 9 children during May to July. As a result, 70% of children shows that a program that they made consistent with the tasks. In addition, we found that even if practicing tasks are completed in the lesson, children might start doing other challenges on the same stage and destroy previous program after they completed tasks. So, we could not understand whether they understand when they tried to practice tasks. Therefore, it necessary to take screenshots during lessons to check whether they manage to do tasks.

By observing each person about the video, we felt the possibility of being able to grasp the change of each person that cannot be found by program observation alone. Thus, we felt the possibility to grasp what kind of changes caused through programming lessons by combining log of program and observation of each person by video research.

References

Harada, Y. (2010). Computer Programming Education Using the Visual Programming Language Viscuit. NTT Technical Reiew , 8 (11), 1-5.

Harada, Y., & Richard, P. (2003). Fuzzy rewriting: soft program semantics for children. Proc. of the 2003 IEEE Symposium on Human Centric Computing Languages ans Enviroments , 1 (1), 39-46.

Smith, D. C., Cypher, A., & Spohrer, J. (1994). KidSim: programming agents without a programming language. Communications of the ACM , 37 (7), 54-67.

Wing, J. M. (2006). Computational Thinking. Communications of the ACM , 3 ed.:, 33-35.