



애니메이션 제작 학습 모형을 통한 초등학교 코딩교육에 관한 연구 - 스크래치 활용 중심으로

서갑열¹·박희현^{2*}

¹홍익대학교 게임학부

²한서대학교 영상애니메이션학과

Study on Elementary Coding Education through Animation Production Learning Model - Focused on the utilization of Scratch

Gapyuel Seo¹ · Heehyeon Park^{2*}

¹Department of Games, Hongik University, Sejong-si, 2639, Korea

²Department of Motion Graphics & Animation, Hanseo University, Seosan-si, Chungcheongnam-do, Korea

[요 약]

오늘날 디지털기술이 빠르게 발전되면서 초등학생들을 위한 코딩 및 프로그래밍 교육의 중요성도 함께 강조되고 있다. 하지만 어린이를 위한 코딩 교육은 프로그래밍 언어를 교육하는 것에 집중되어 있어서 그것을 효율적으로 교육하고 활용하는 방법에 대한 연구는 많지 않은 것이 현실이다. 이 연구에서는 학습자가 신속하고 흥미롭게 학습 할 수 있는 코딩 교육 방법으로 스크래치를 활용한 애니메이션 제작방법을 사용하였다. 본 연구 결과를 위해 총 10명의 학생들을 대상으로 7주에 걸쳐 교육하고 실험해본 결과, 학습자들은 스크래치를 활용한 애니메이션 제작 과정을 통해 자연스럽게 프로그래밍 언어를 학습하며 흥미롭게 코딩을 학습하는 것을 확인할 수 있었다.

[Abstract]

Today, digital technology is rapidly evolving and the importance of coding and programming education is emphasized. However, coding education for children is focused on teaching programming languages and there is not much research on how to efficiently educate and utilize them. In this study, we used a scratch - based animation production method as a coding education method for learners to learn quickly and interestingly. As a result of this study, 10 students were trained and experimented for 7 weeks. As a result, it was confirmed that the learners learn the programming language naturally through the process of animation using scratch and learn interesting coding.

색인어 : 애니메이션, 코딩, 프로그래밍, 스크래치, 코딩 교육 모델

Key word : Animation, Coding Education, Programing Education, Scratch

<http://dx.doi.org/10.9728/dcs.2018.19.12.2323>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 28 November 2018; **Revised** 16 December 2018

Accepted 23 December 2018

***Corresponding Author; Heehyeon Park**

Tel: +82-41-660-1477

E-mail: hpark@hanseo.ac.kr

I . Introduction

The development of digital technology has changed our culture and daily lives rapidly. In the field of education also has been changed by using digital technology rather than traditional cramming method of education. Prensky claimed that today's students are no longer the people our educational system was designed to teach, which means that they need a suitable tools and methods in learning for digital generation[1]. In recent year there is increasing recognition that the learning of computer language is the fundamental education in order to live in digital age.

According to the announcement of the curriculum reform from ministry of education in 2015, "Information" subject will be the mandatory course of junior high school from 2018, and "Basic Education of Software" will be the mandatory course of elementary school from 2019. Based on this announcement, the need of software education from elementary school have recently emerged, and this will introduces the experience of programming in early education. However, there is a lack of instructional models in programming learning that can give students interests and motivations. The purpose of this study is to examine the possibility of animation production in coding education and to suggest a coding education model that utilizes the convergence characteristics of animation rather than existing coding education of functional learning.

In this paper, we propose a instructional model that will create an animation project in order to learn a programming language for primary students. The goal of the project was to design an integrated system for the 5th grade students that combined digital animation with a coding-based learning utilizing "Scratch" to promote creative thinking, computational thinking, logical thinking and problem-solving ability. Furthermore, we suggest the learning model for self-directed learning geared towards primary students.

II . Background

2-1 Coding Education

In recent years, coding education has been carried out based on the basic common knowledge in order to live the digital world rather than simply teaching programming to be a programmer. In particular, various methods have been developed to make them interested in coding from an early age.

Coding education for children is focused on helping to

understand algorithms principle naturally by play and game. It is a method that tells a loop or calculation principle without using programming terminology. Most of them are doing coding education using easy-to-use games like character-based games or drag-and-drop methods.

In this way, since it is easy to learn from a young age, the coding education has a great effect on the improvement of comprehensive thinking ability, and many countries around the world tend to introduce curriculum into regular subjects. These are the current state of coding education in each country. Coding education is expected to become a key factor in enhancing national competitiveness, and coding will become a basic education essential for the digital age.

2-2 Scratch

Scratch program is developed by the Lifelong Kindergarten Group at the MIT Media Lab. Scratch is a visual programming language that provides a rich learning environment for people of all ages. It allows you to create interactive, media-rich projects, including animated stories, book reports, science projects, games, and simulations. The scratch is very convenient tool for learning coding in primary school students, also is effective tool to learn quickly and express student's thoughts easily. User's can drag and drop the block palette and attaching them to other blocks like a jigsaw puzzle. Structures of multiple blocks are called scripts[2]. This method of programming (building code with blocks) is referred to as "drag-and-drop programming"[3]. Programming is done by dragging command blocks from a palette into the scripting panel and assembling them, like puzzle pieces, to create "stacks" of blocks.

There have been several studies on the utilization of Scratch. The most of studies were related to education purpose in various areas. The results of previous studies related to Scratch are shown in Table 1.

Scratch is an intuitive language that can be programmed through block building like the LEGO block. Unlike other educational languages, Scratch makes programming easier for learners who want to start learning a programming language. So beginners are easy to understand, easy to learn, so they can learn algorithmic ways of motivating learners and approaching problems in a variety of ways to create animations.

Table 1. A Previous Study on Scratch

Researcher	Research Result
Seungki Shin et al (2014)	improving problem solving capacity from logical problem solving.
JungCheol Oh, et al (2013)	positive effect on the creativity and affective characteristics related with science
Burke, Quinn, et al. (2012)	embody both the technical and the creative elements of composition and offer a broader conception of what “writing” with computers may look like in the 21st century.
Olabe, J. C., et al.(2011)	improving the goal setting, team working, problem resolution and communication with other students
Maloney, John, et al. (2010)	new forms of self-expression, producing a diverse range of projects while deepening their understanding of a core set of computational ideas
S. Garner,(2009)	gain knowledge of data and control structures, and basic algorithms used in business systems
Fesakis, Georgios, et al.(2009)	positive effect on students' opinions and attitudes towards computer programming

2-3 Animation

The animation production process has been studied very deeply for a long time. It has a very systematic process from brainstorming to story composition, creating animation, and presentation. Since it is possible to make animation with Scratch, we think that coding education through creation of animation using scratch is very effective. Scratch allows users to use event-driven programming with multiple active objects called sprites[4]. Sprites are created as vector or bitmap images and can be created directly through the scratch editor, or it can be imported from other sources. Scratch contains 10 groups of blocks such as motion, looks, sounds, pen, data, control, events, sensing, operators and more blocks. These blocks are essential elements for animation production.

III. Learning Model

2-1 Learning model of animation production

This research is implemented in five stages of the animation creation based on coding. Fig. 1 shows the stages for the process of animation production based on coding.

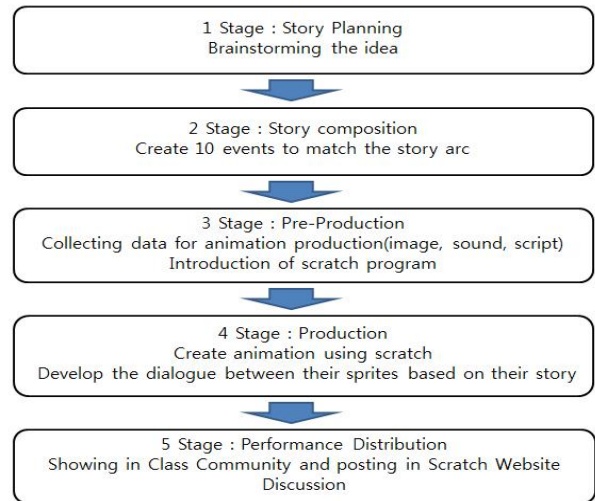


Fig. 1. Learning model of animation production

For the test of this learning model, we have 5 male and female elementary students aged 11, in South Korea. The students were familiar with computer-based education, since the use of laptops is an essential part of their curriculum. They had never learned coding before, but some had prior experience of an animation and story creation. This program proceeded for 7 weeks to make an animation from scratch based on their own story creation.

First stage is a story planning stage that students can brainstorming the idea for story creation. Some students have a experience about story creation but the other have a hard time to create their own story from scratch. We suggest to them to using Story Starters from Scholastic. Scholastic's Story Starters kids' writing activity generates creative writing prompts, from general fiction to adventure, fantasy, and science fiction. Students can choose a theme and then produce prompts using the Spin lever. They can adjust one piece of the prompt at a time with the Spin This Wheel buttons[5]. Fig. 2 shows the Story Starters from Scholastic.

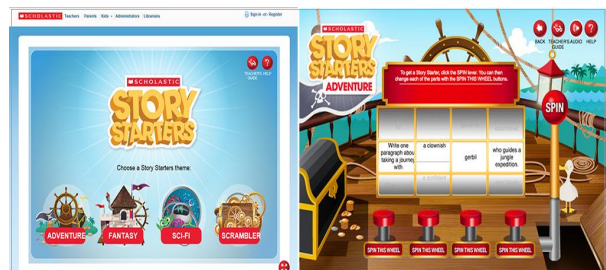


Fig. 2. Story Starters from Scholastic

Second stage is a story composition stage that students can create 7 to 8 events based on Freytag's pyramid. First, we introduce to them about Aristotle's three-act structure. Aristotle is a famous Greek philosopher, scientist and poet, could be the first person to bring up the idea of plot structure. As he said: "A plot must have, a beginning, a middle, and an end, and the events of the plot must causally relate to one another as being either necessary or probable." [6]. It consist of three stages which is beginning , middle and ending.

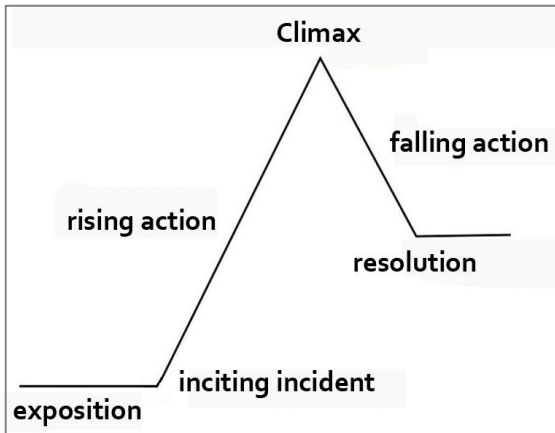


Fig. 3. Freytag's pyramid

The beginning stage is the set up stage and, we introduce to them how to set up the characters and setting. The middle stage is the confrontation stage and, we introduce to them to develop the story so that it leads up to an exciting climax. The ending stage is the resolution of the story and, we introduce to them how to solve the conflicts and end the story. As you build a story, you must set up the characters and make up the various events that the person makes, and reveal the thoughts, emotions, greed, and desires that are in their inner worlds in the ambassadors and trivial actions that are delivered through their mouths [7]. Second, we introduce to them about narrative structure based on Freytag's pyramid. Gustav Freytag is a contemporary German novelist and playwright. Freytag considered plot as a narrative structure that divided a story into five parts, which is known as Freytag's pyramid [8]. Fig. 3 shows the Freytag's pyramid. After demonstrating and explain Freytag's pyramid with many stories, students assigned their stories to the Freytag's pyramid and created seven to eight events. Fig. 4 shows the story structure setting from student work based on Freytag's pyramid.

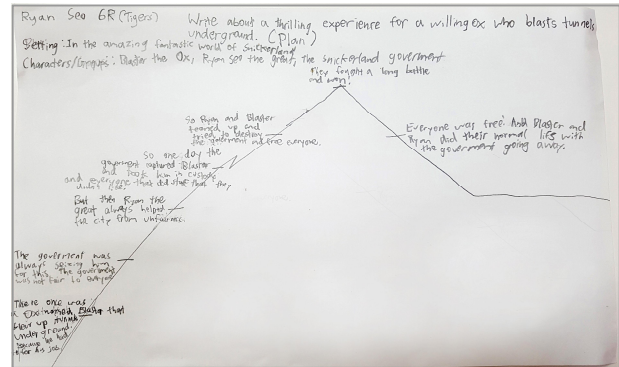


Fig. 4. Story events based on freytag's pyramid (student work)

Third stage is a pre-production stage that students collect data for animation production such as sound, image and script. We also explain basic information about what Scratch is. students are introduced to Scratch programming project by viewing a collection of example projects, so they can think what will be able to do in Scratch.

Fourth stage is a production stage that students create animation using scratch. In the beginning of this stage, student need to create dialogue between their sprites based on their story. When students were ready to create animations, we taught coding for three weeks. Scratch consists of ten blocks of commands under the Scripts tab, which you drag blocks from Scripts tab into the screen area. Fig. 5 shows the main programming blocks in order to create an animation in Scratch. The fourth stage of instruction focused on the core blocks such as Motion, Looks, Sound, Events, Sensing and Control. The fourth stage of training was conducted for two hours per week. The first week consisted of learning the basics of scratch and sprites. The second week consisted of learning how to move the sprite and put the bubble in the dialogue between the characters. The third week consisted of event-driven programming to determine when actions was should occur. Students are encouraged to create animations based on what they have learned each week, and the video tutorial allowed them to continue to study scratches outside of class hours.

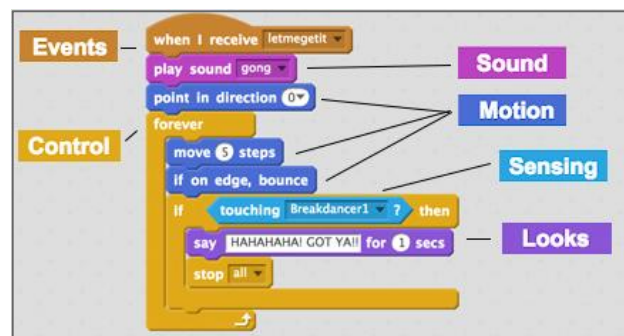


Fig. 5. The main programming blocks for animation

Table 2. Contents of training for three weeks

	Subjects	Education Contents
First week	-Scratch basics -Import and load Sprite	navigation Tools Sprite
Second week	-Moving Sprite -Dialog Bubble and background -Adding sound	Motion block Looks block Sound block
Third week	-Event : driven programming -Control : Conditional if-else statement -Sensing : interact with the surroundings	Event block Sensing block Control block

After three weeks of scratch learning, students set the characters and background images of the story. After collecting the image, create a dialog to be inserted into the animation and create the animation using scratch. Fig. 6 shows the student animation work utilizing the programming blocks in Scratch.



Fig. 6. Scratch animation (student’s work)

Fifth stage is performance distribution stage that students showing their work in class and posting in Scratch web-site. Students can upload their projects, share ideas with other users and look at each other's projects. Students can upload their work directly to the Scratch website, and community members can download the full source code so that they can create and edit it with other projects[9][10][11]. You can also develop your project by entering your ideas into other users' projects. Students can share and develop with each other.

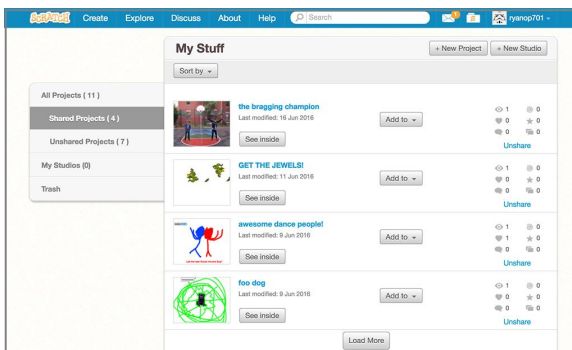


Fig. 7. Uploaded student work in Scratch Website

V. Conclusion

Existing programming education required a lot of time and effort to learn grammar of programming language and learn how to use tool. Thus, learners are primarily focused on completing executable coding without having to write errors and memorize the grammar of the language rather than solving problems through the programming language. Therefore, they can improve their original problem solving ability, logical thinking ability, and creativity[12]. This study aimed to develop and apply an educational programming learning that can be applied to elementary school education. We have designed and proposed a learning model in order to make it easier for students to understand the principles of programming so that students will be able to easily reach out to their learning goals by creating an animation that utilizes scratch [13].

The use of scratch is capable of implementing interactivity with learners in various contexts such as graphics, sounds, and animation, and has the advantage of inducing the interest of learners and enhancing their concentration. In this study, we found that the expected effects are as follows ;

First, A learner create his or her own favorite story and directly produces an animation based on the story which it will induces interest and provides a learner-centered learning environment and a learning method capable of interacting through scratch. We expect to be established as a convenient self-directed learning tool that can reflect the needs of young learners.

Second, Students can combine commands to create an animation by moving character to improve logical thinking, deductive reasoning, and creativity.

Third, it is difficult to understand basic principles for beginners who are new to computer programming. However, it is possible to provide an experience to naturally acquire basic principles that are appropriate for learners' ability through the creation of animations using scratch.

We have been able to explain the utilization of educational programming language in the development of animation and suggested learning model. However, we found that we did not need to use various commands to create animation. As a future work, we will study educational programming learning through game creation so that students can build a project with various blocks in Scratch.

References

- [1] Mrac Prensky, Digital Natives, Digital Immigrants, *MCB University*, Vol.9 No.5, 2001.
- [2] Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M., & Rusk, N. Programming by choice: urban youth learning programming with scratch. Vol. 40, No. 1, pp. 367-371, 2008.
- [3] Scratch Wiki. [Internet]. Available: <https://wiki.scratch.mit.edu/wiki/Scratch>.
- [4] Wikipedia. Scratch (programming language). [Internet]. Available: [https://en.wikipedia.org/wiki/Scratch_\(programming_language\)](https://en.wikipedia.org/wiki/Scratch_(programming_language)).
- [5] Scholastic. Story Starters. [Internet]. Available: www.scholastic.com/teachers/story-starters.
- [6] Arnold B, Eddy B. Exploring Visual Storytelling. New York: Thomson Delmar Learning. 2007.
- [7] Mou, T. Y., Jeng, T. S., & Chen, C. H. From storyboard to story: animation content development. *Educational Research and Reviews*, 8(13), 1032. 2013.
- [8] Nockwon Song. Write scenario, Communicationbooks, ISBN 9791130400990. 2014
- [9] Olabe, J. C., et al. "Programming and robotics with Scratch in primary education." *Education in a Technological World: Communicating current and Emerging Research and Technological Efforts*: 356-363. 2011.
- [10] Monroy-Hernandez, Andres; Hill, Benjamin Mako; Gonzalez-Rivero, Jazmin; Boyd, Danah. Computers Can't Give Credit: How Automatic Attribution Falls Short in an Online Remixing Community. *Proceedings of the 29th International Conference on Human Factors in Computing Systems (CHI '11)*. ACM. pp. 3421-30. 2011.
- [11] Hill, B.M; Monroy-Hernández, A.; Olson, K.R. Responses to remixing on a social media sharing website. *ICWSM 2010 : Proceedings of the Fourth International Conference on Weblogs and Social Media*. 2010.
- [12] Seungki Shin, Phanwoo Park. A Study on the Effect affecting Problem Solving Ability of Primary Students through the Scratch Programming. *Advanced Science and Technology Letters* Vol.59, pp.117-120. 2014.
- [13] Gapyuel Seo, Eunkyung Choi. A Study of learning model for elementary school utilizing the development of animation based on Scratch, *Proceedings of the 18th Conference on Korean Animation Society* 14-15. 2016



Gapyuel Seo

1998 : Pratt Institute (B.F.A Degree)
2009 : Savannah College of Art and Design (M.A. Degree)

1999~2010: Activision Blizzard, Senior Artist
2010~2012: Department of Digital Arts, Springfield College, Assistant Professor
2012~now: Department of Games, Hongik University, Assistant Professor
※ Research Interests: : Game Graphics, Game Design, Game Storytelling.



Heehyeon Park

1998 : Pratt Institute (B.F.A Degree)
2001 : Pratt Institute (M.S. Degree)

2005~2010: GE (General Electric), Senior Designer
2010~2012: Department of Graphic+Media Design, SAGE College, Adjunct Professor
2015~now: Department of Motion Graphics & Animation, Hanseo University, Assistant Professor
※ Research Interests: : Motion Graphics, Animation, Visual Communication, UI/UX Design. Media Literacy.