COMPARISON OF TEXT-BASED AND VISUAL-BASED PROGRAMMING INPUT METHODS FOR FIRST-TIME LEARNERS

Responding to comments

Daisuke Saito, Hironori Washizaki, Yoshiaki Fukazawa

To Prof. Tian Lau, the reviewers, and editors

Thank you for your careful review, comments, and advice. They were extremely beneficial and have greatly improved our manuscript. Below are the responses to your comments.

Thank you for your support.

Comment 1

I am pleased to accept your revisions. However, there are still a number of language issues that need to be addressed before the article can be accepted unconditionally. Even though I gave you a go-ahead with your paper, the publisher/copy editor is likely to reject the paper based on the English quality. We ask you to consider revising the paper using Grammarly Professional.

Answer to comment 1

The revised manuscript was proofread by Grammarly Professional and a native English speaker. As a result, the paper is improved. An example of the modifications is shown below:

Revised contents

ABSTRACT

The following results are revealed: (1) The visual input method induces a larger change in attitude toward programming. (2) The number of operations and input quantity influence both groups. (3) The overall results suggest that a visual input is advantageous in a programming implementation environment for first learners.

INTRODUCTION

Both visual input and text input methods have been used to teach programming to beginners. In programming, first-time learners (first learners) tend to employ visual input methods instead of text methods. Although some studies have found that visual inputs are superior, others have shown that text inputs can be adapted to first learners (Saito, & Yamaura, 2013). In addition, several tools have been developed for both types of programming in the same environment (Bau, Bau, Dawson, & Pickens, 2015; Fraser, 2013). However, few studies have compared the learning effects by input method. Hence, whether one programming method (text or visual) is more suitable for introductory education remains unknown (Price, & Barnes, 2015; Weintrop, & Holbert, 2017).

Herein the differences between visual and text input methods are investigated in the same Lua programming environment to determine if the input method influences the learning effect. Specifically, this research examines the following Research Questions (RQs):
Research Question 1 (RQ1): Does a visual-based input method induce a different attitude toward programming than a text-based input method?

Research Question 2 (RQ2): Does the understanding of programming between visual-based and text-based input methods differ?

RQ1 assesses whether the programming method is suitable for an introductory environment. This RQ can elucidate the attitude of first learners toward programming by input method, which should assist in selecting the most suitable method for introductory programming. RQ2 evaluates the understanding of programming basics. Furthermore, it examines the understanding of programming concepts by focusing on a sequential execution, conditional branching, and repetition. This RQ can reveal which method is more suitable for learning. These RQs should help determine the more appropriate programming method and environment for introductory education because increasing the learning efficiency should enhance the learning effect. In addition, the proper learning environment should improve first learners' motivation to learn.

The rest of this paper organized as follows. The Background Chapter explains the background and previous works. The Lecture Design Chapter discusses the lecture design used to compare the two methods. The Experiment Chapter and the Analysis Method Chapter describe the experiment using the lecture design and the analytical techniques, respectively. The Results Chapter overviews the results. The Discussion Chapter evaluates the results. Finally, the Conclusion and Limitations Chapter provides the conclusions, threats to validity, and future work.

BACKGROUND

Programming Learning for First Learners

It is often noted that beginners have difficulty learning to program (Kelleher, & Pausch, 2005; Gross, & Powers, 2005). Several studies have been conducted to address this issue. Some studies used a visual method like Scratch developed by MIT (Maloney, Resnick, Rusk, Silverman, & Eastmond, 2010; Chiu, 2015; Sáez-López, Román-González, & Vázquez-Cano, 2016). A different study used a text method such as C (Saito & Yamaura, 2013). Other studies used both visual and text methods for Project-based Learning for programming based on problem-solving (O'Kelly & Gibson, 2006) and Game-based Learning (Long, 2007; Jiau, Chen, & Ssu, 2009; Vasilateanu, Wyrazic, & Pavaloiu, 2016). In addition, some works investigated attitudes toward programming (Du, 2016).

Each method has its own learning effect. Some success has been reported for first learners using these methods. However, which programming input method (visual or text) is more suitable for first learners remains unknown. In addition, the learning effect for each method is unclear. Based on this situation, this paper focuses on the input method and compares the learning effect in the same programming environment. It is intended to serve as a reference for educators when selecting an input method to teach first learners.

Two input methods

In this paper, we evaluate the learning effect of text inputs and visual inputs in the same programming language. A comparison of the learning effects in text and visual methods can be traced back to the Dual-coding theory (DCT) raised by Paivio (Paivio, 2013; Clark, & Paivio, 1991). In this theory, human information processing can be divided into two systems: language and non-language. Language systems are character information such as characters and voices, while non-language systems are sensory information such as images. These affect human recognition (Clark, & Paivio, 1991).

Studies have examined characters and images using DCT. One study investigated the influence of student's prior knowledge on a computer-based physical lesson learning due to differences in the
A programming language can be expressed as text representations or visual representations. For example, visual programming languages like Scratch (Maloney et al., 2010) or Alice (Dann, Cooper, & Pausch, 2011) use drag and drop of visual inputs for program learning. A visual language is suitable as a language initially performed by first learners who are unfamiliar with a programming language. Furthermore, text programming languages such as Python and JavaScript use a keyboard to type inputs. Text languages can be more sophisticated than visual languages. Although a text language is better suited if the purpose is clear, learners must possess sufficient typing skills. In addition, some researchers have investigated the transition to text-based programming from visual-based programming (Kölling, Brown, & Altadmri, 2015; Lahtinen, Ala-Mutka, & Järvinen, 2015; Bau et al., 2015). Hence, the research results implemented in the field of multimedia are applicable for first learners in programming.

In a study comparing programming methods, visual programming methods are noted to be an easy educational method for educators (Weintrop & Wilensky 2015a). Studies on programming in higher education indicate that visual-based languages produce better results than alternative programs (Weintrop & Wilensky, 2015b). One study developed an extended function of Codeblock, which expands the visual programming function to Minecraft. It reported that this tool results in improved recognition of programming. Although this study compared the visual program function to text environments, a significant difference was not detected (Zorn, Wingrave, Charbonneau, & LaViola Jr, 2013).

Several reports in multimedia and programming learning have indicated that the visual method is suitable for first learners. However, they suggest that using a visual input method may be more advantageous for first learners. However, programming involves both visual information as well as behavioral aspects such as input of programs and confirmation of execution results. It is difficult to support all results in the multimedia field. In addition, there is no significant difference in recognition of programming compared to the text environment (Zorn et al., 2013). Consequently, the proper input method for first learners is not definitive. This study uses visual inputs and text inputs at the same abstraction level built in the same environment to provide clear answers. Hence, the comparison is based on the input differences without the effect of the environment. Additionally, this study strives to include younger participants. Table 1 highlights the difference from recent studies.

### RESULT

**Analysis of the results**

We analyzed the questionnaire results using the Wilcoxon signed-rank test and the Wilcoxon rank sum test (p-value < 0.05). Table 5 shows the results for A1, while Table 6 shows the results for A2 – A3. VG improved in all categories according to A1. In particular, the attitude towards turtle programming improved and the interest in turtle programming improved by about 0.6 points. However, some of the learners showed a lower value for attitude after the lecture. It is possible that some learners became bored with programming or were more absorbed in playing the game than programming. Some categories for TG also improved to a positive attitude, while others decreased to a negative attitude. The large amount of input necessary to program may induce a negative attitude. The attitudes for interest and difficulty of programming became positive. Furthermore, the attitudes for interest, difficulty, and fun of turtle programming improved.
Understanding Programming

Problem results and analyses

We used tests and questionnaires to confirm the comprehension level of first learners. There were six questions (Table 2) and one free problem. Each learner self-declared when a problem was complete and then took a screenshot to confirm the solution. In addition, we acquired the source code as the answer information. Figure 5 shows the response rate. The low response rate was a problem. There was not any difference in P1 by the group. For P2, the percentage of correct answers was higher for VG than TG. This difference is attributed to the amount of input required to program. TG provided a higher percentage of correct answers than VG for P3, which was about loop sentences, indicating that the operation amount (input amount) of VG increased. Consequently, the correct answer rate decreased for VG. The result of P4 was the same as P3. On the other hand, P5, which was about conditional statements, showed the opposite result; that is, VG had a higher percentage of correct responses than TG. Complicated condition expressions had to be inputted for TG. Hence, to obtain a correct response from TG is more difficult than VG. The result of P6 is the same as P5.

DISCUSSION

Result of RQ1

- **RQ1**: Does the visual-based input method induce a different attitude toward programming than the text-based input method?
  - **H1**: The visual input programming lecture induces a larger change in attitude toward programming.

In RQ1, there is a difference between VG and TG for visual expression. The results also differ from previous research. Zorn et al. used the Mod of CodeBlock for a student lecture course in 2013 (Zorn et al., 2013). Their research, which compared the learning effect of block programming to text programming, found that block programming increases student interest. In our research, VG shows statistically significant differences in "Interest" and "Usefulness" of turtle programming. VG also shows statistically significant differences in "Interest" in programming. These results indicate that visual inputs are likely to increase the interest in programming. Our results provide additional findings. VG also increases "Usefulness" and "Willingness". VG is more intuitive than TG. Because a keyboard was not used in VG, less time is necessary to see results.

TG has no statistically significant difference in some of the analysis. However, the arithmetic mean shows the difficulty programming in TG has a larger improvement than VG. A previous comparison study investigated programming difficulty (Price & Barnes, 2015). According to their research, a novice cannot distinguish the cause of programming difficulties because they do not recognize challenges due to interface differences. However, our study reveals a difference in attitude. This difference may be because the text input is a more realistic programming method than the visual input. Learners may have a prejudice that text is more representative of programming and it is perceived as more difficult. However, the lecture interposed games, creating the possibility that learners felt programming is easy. Hence, programming difficulty TG shows a larger improvement.

The visual input method improves attitudes towards programming more than the text input method. Although the results are not statistically significant, we can confirm H1. Regarding programming attitude for first learners, the visual input method is more suitable. However, the text input method should reduce the difficulty level more than the visual input method. Consequently, the text input method can be adapted to first learners. Accumulating more data in the future should further distinguish between the two input methods.
CONCLUSION AND LIMITATIONS

We examined whether text input method or visual input method is better for first learners. In the field of DCT and multimedia, it has been reported that visual expressions as well as applying text and images in a balanced manner are effective for first learners. Hence, it may be beneficial to teach programming visually. However, programming involves behavioral aspects, such as entering and executing programs. Because information is acquired by more than just site, programming has different aspects from multimedia learning. Furthermore, some studies have applied and compared programming learning methods for first learners, but it is unclear whether a visual input or a text input is more suitable for first learners. Therefore, we investigated the difference in the learning effect of two input methods using ComputerCraftEdu in Minecraft for programming involving first learners. The visual input method results in a larger change in attitude. Significant differences are noted, especially in the interest in programming (including turtle programming). Although the text input seems to make programming less difficult, the difference is not significant.