



**HAL**  
open science

# Clas-Maze: An Edutainment Tool Combining Tangible Programming and Living Knowledge

Qian Xing, Danli Wang, Yanyan Zhao, Xueyu Wang

► **To cite this version:**

Qian Xing, Danli Wang, Yanyan Zhao, Xueyu Wang. Clas-Maze: An Edutainment Tool Combining Tangible Programming and Living Knowledge. 19th International Conference on Entertainment Computing (ICEC), Nov 2020, Xi'an, China. pp.353-368, 10.1007/978-3-030-65736-9\_32 . hal-03686018

**HAL Id: hal-03686018**

**<https://inria.hal.science/hal-03686018>**

Submitted on 2 Jun 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Clas-Maze: An Edutainment Tool Combining Tangible Programming and Living Knowledge

Qian Xing<sup>2,1</sup>, Danli Wang<sup>1,2,\*</sup>, Yanyan Zhao<sup>1</sup> and Xueyu Wang<sup>1</sup>

<sup>1</sup> The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China

<sup>2</sup> School of Artificial Intelligence, University of Chinese Academy of Sciences

\* danli.wang@ia.ac.cn

**ABSTRACT.** With the development of computer technology, children's programming technology has become increasingly mature. However, the current content and form of children's programming tools are too singular and do not integrate well with daily life skills. In the era of vigorously advocating environmental protection, it is particularly important to cultivate children's environmental awareness and ability from an early age. The paper described a new edutainment tool named Clas-Maze, for children in 5-9 years old, which combines tangible programming and living knowledge, such as garbage classification. Clas-Maze consists of three parts: Programming blocks which can be used to construct a program to control the garbage's route, external camera which can be used to collect the information of programming blocks, and the virtual environment which is the execution interface of the system on the computer. We wanted to explore whether the system is helpful to children in learning garbage classification and programming, and the difference between single and collaborative learning. So we conducted a user experiment with 37 children. The results showed that Clas-Maze can help children learn programming and garbage classification. Single and cooperative programming have their own advantages. In order to cultivate children's ability of decision-making, communication and cooperation, people can choose cooperative programming.

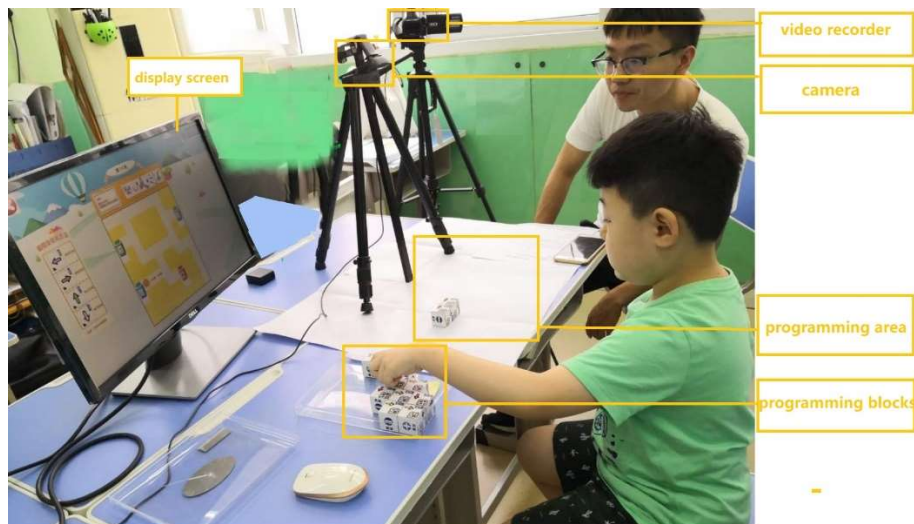
**Keywords:** Edutainment Tool ; Children's Programming; Computational Thinking; Garbage Classification; Learning Effects;

## 1 Introduction

With the rapid development of computer technology, training of children's computational thinking has gradually attracted people's attention in the education community [1]. At present, the international community believes that children's computational thinking is as important as "reading, writing, and computing" skills. It is also a cognitive skill that everyone should master. Cultivating this computational thinking mode from a young age can greatly improve their logical thinking ability, cognitive ability, creative ability, etc[2][3]. From American "programming one hour a day" campaign to adding programming education to the national curriculum in Britain [4][5], and then to China's

A New Generation of Artificial Intelligence Development Plan, programming education is gradually being extended [6][23]. Traditional programming languages are based on the form of text or symbols, and involve complex grammar and instructions. It is difficult for children to learn [7][12], so some simpler programming methods have been developed for children's programming learning, such as graphical programming language Scratch [8] and tangible programming language Tear [9][10].

In recent years, with the rapid development of economy and society, environmental problems have become increasingly serious, and people's awareness of environmental protection has been constantly enhanced. New regulations on garbage classification, which are more scientific and meticulous, have also been strongly promoted in most urban areas of China. Garbage classification refers that people put the same kind of garbage into the same dustbin in the designated place. For children, learning garbage classification not only helps them to establish environmental awareness, but also helps them to improve their ability of learning and classification. *Notice on Promoting the Management of Domestic Garbage Classification in Schools* [11] was issued by Chinese Ministry of Education on January 16, 2018, which proposed to widely adopt the form of telling stories, playing games, knowledge competition and other activities to carry out a variety of garbage classification themed education activities. And people should make full use of wall map, blackboard newspaper, publicity window, campus website and other publicity positions to vigorously promote garbage classification.



**Fig. 1.** Overview of Clas-Maze system

The paper presented a new edutainment tool, Clas-Maze combining tangible programming and knowledge of garbage classification (Figure 1). The system is designed for children 5-9 years old, so that they can not only cultivate computational thinking, but also improve their ability of garbage classification when they learn to program. This paper explored three questions through the experiment: 1) Can children learn the

knowledge of garbage classification through Clas-Maze, and whether there is a difference in the learning effect between single and double groups? 2) Can children learn the programming knowledge through Clas-Maze, and whether there is a difference in the learning effect between single and double groups; 3) Whether there is a difference in the learning performance between single and double group. Our study showed that the children's two abilities were improved significantly after the experiment, but there was no significant difference between two groups.

In the next sections, the related work, description of system and user study are presented. Result analysis, limitation and future work are also discussed.

## **2 Related Work**

### **2.1 Children's Programming Education**

In recent years, children's programming education and programming tools have been deeply studied. Relevant research proved that using interactive programming tools in computer teaching can increase students' enthusiasm and participation [28]. Numerous human-computer interaction laboratories and scientific research institutions have conducted research on tangible programming, such as Northwestern University, Massachusetts Institute of Technology (MIT), University of Colorado, Carnegie Mellon University (CMU), and Google. There are some great tangible programming tools, for example Storytelling Alice [13] is based on Alice 2.0 programming tool for storytelling. roBlock [14] supports user to set a robot by connecting different blocks, and robot can do specific actions by logical calculation. TurTan [15] is a programming tool based on computer vision technology, which uses a camera to capture and identify the location of fingers and objects on the desktop. T-Maze [16] is a tangible programming tool, in which children play the role of escaping the maze, and choose the way forward by placing programming blocks. TanProStory[22] is programming tool for storytelling, in which children can tell a story by arranging programming blocks to control a character. Tern [25], Strawbies[26], CoProStory [29] and Loopo [12] combine tangible language and virtual display interface. Users connect wood blocks or electronic slices to form program control flow, and get programming feedback through screen display, which can teach children computer programming knowledge [24].

The above tangible programming tools all use physical interaction technology, but they are all presented in the form of games, rarely combining practical skills in life. This paper proposed the programming system that combines the garbage classification knowledge and programming knowledge. It not only can develop children's computational thinking and promote cognitive development of children, but also can help them to improve the ability of garbage classification.

### **2.2 Cultivate children's garbage classification**

The cultivation of children's living ability is an important aspect of all-round development education for children. Today, with the continuous promotion of quality education, it is very important to improve children's living ability [17]. Garbage classification

is a very frequently used item in living capacity. Many countries in the world have cultivated children's garbage classification ability since childhood, such as Sweden, Belgium, Japan and so on. The most typical country is Japan. Japanese kindergartens take garbage classification behavior cultivation as an important part of their education. They cultivate children's garbage classification ability from daily life, on-site visits, watching related cartoons, playing games and some other ways[18]. With the increasing efforts of environmental protection in our country, people also pay more and more attention to cultivate children's ability and awareness of garbage classification, we believe it would be very beneficial to carry out garbage classification theme education in kindergartens.

### 2.3 Summary of related work

Analysis of the above related work shows that children's tangible programming tools and related technologies are relatively mature, and young children can learn programming through physical operations. However, previous programming tools only teach children programming knowledge, not combine living knowledge. Therefore, we implement Clas-Maze, which is aimed to improve children's multi-faceted abilities.

## 3 Clas-Maze Description

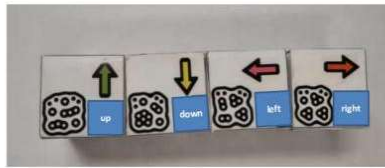


Fig. 2. Tangible programming blocks

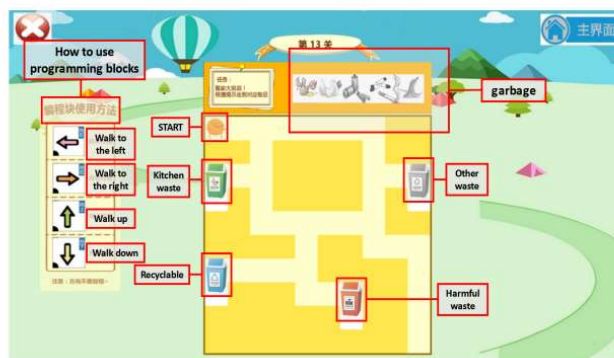


Fig. 3. The virtual environment of garbage classification

Clas-Maze contains three parts: tangible programming blocks, external camera and the virtual environment of garbage classification. Programming blocks are made by 3cm wooden brick cube as shown in Figure 2. There are four faces in each block, which can be used to express four distinct semantics, and to reduce the number of programming blocks, thus lowering cost of the system. In the process of programming, the image of tangible programming block is collected by external camera, and the collected image is converted into programming language through ReacTIVision visual recognition library.

The virtual environment of garbage classification includes garbage, maze, and dustbins, as shown in Figure 3. There are four kinds of dustbins namely kitchen waste, recyclable, harmful waste and other waste, as shown in Figure 4. While playing the game, children firstly need to judge the type of a garbage and then help the garbage get back “home” by placing the programming blocks. In order to make children better understand the function of programming blocks, Clas-Maze also provides feedback on programming. If programmed correctly, a forward arrow will appear, otherwise no arrow will appear and the route will stay at the previous step.



Fig. 4. Four kinds of dustbins

## 4 User Study

To evaluate children’s learning effect and performance, we conducted a lab-based user experiment with children.

### 4.1 Goals of Study

In this experiment, by analyzing the performance of children in the experimental video and questionnaires, we mainly explored the following three questions:

1. *Can children learn the knowledge of garbage classification through Clas-Maze, and whether there is a difference in the learning effect between single and double groups?*
2. *Can children learn the programming knowledge through Clas-Maze, and whether there is a difference in the learning effect between single and double groups?*
3. *Whether there is a difference in the learning performance between single and double group?*

## 4.2 Participants and Environment

The experiment was conducted on August 6th and 8th, 2019 at Beijing Huacai Education Center. The whole experiment was conducted in a spacious and quiet classroom in Huacai Education Center, and the temperature and humidity met the requirements of natural comfort. In addition to the necessary tables and chairs, the room also includes a video recorder, two notebooks, a camera, and a display.

**Table 1.** The specific information of all participants

Group	Age	Gender	Tangible program	Garbage classification
G1 (single)	Range=6-7	7 girls	2 Had learned	5 Had learned
	Mean=6.19	6 boys	11 No learned	8 No learned
	SD=0.24			
G2 (double)	Range=5-8	12 girls	2 Had learned	16 Had learned
	Mean=6.48	12 boys	22 No learned	8 No learned
	SD=0.76			

A total of 37 healthy children participated in the experiment. There are 19 boys and 18 girls, and their mean age is 6.38 years old ( $SD = 0.64$ ). The participants were divided into Group G1, which completed the experiment independently, and Group G2 which completed the experiment by two children's cooperative programming. The specific information of all participants is shown in Table 1. To protect the participants' privacy, the researchers assigned the children a test code before they did the experiment.

## 4.3 Experimental Design

**Tasks Design.** Participants in this experiment need to complete two tasks. During the experiment, unless the children ask the researchers for help or there is a problem with the experimental equipment, the researchers cannot provide them with any help.

- Task 1: By placing the programming block, move the first type of garbage, the battery, to the position of "Harmful Trash can" (simple)
- Task 2: By placing the programming block, move the second type of garbage, cigarette butts, to the position of "Other Trash can" (more complicated)

**Process.** The process of the whole experiment is as follows:

- Firstly, we taught children knowledge about garbage classification and programming, including the meaning of programming, sequence, garbage classification and classification standards.
- Then the researchers introduced the Clas-Maze system to the children, including the each part of the virtual environment, the name of each target garbage, the type of each dustbins, rules for programming and designing route, and the use of

programming blocks. After getting familiar with the system, children need to complete a pre-test questionnaire, but they can't get any tips or correct answers from the researchers.

- Next was the practice stage. Researchers led Children to complete a simple exercise task (put the egg shell in the kitchen waste dustbin). And children completed a complex exercise task independently (put plastic bottles in recyclable dustbin), in which they can ask researchers a variety of questions about system.
- Finally, they needed to complete the two formal tasks mentioned in Section Tasks design. During this period, the researchers used the experimental record table to record participants' behavior. At the same time, a video recorder was recording the programming performance and facial expression of them. After the experiment, children still needed to complete a post-test questionnaire.

#### 4.4 Data Acquisition

**Questionnaire.** Using the questionnaires had two main purposes: 1) compare children's ability to deal with programming problems and garbage classification problems in pre-test and pos-test; 2) investigate children's subjective feelings about the system. So they were divided into pre-test questionnaire and post-test questionnaire, which aimed to collect children's personal information, experience of learning garbage classification and using tangible programming, test children's ability and children's perception of system use. Considering children's cognitive habits, this paper used the improved smiley-o-meter rating scale [19] to collect subjective data.

**Record Table and Video Recorder.** In order to facilitate more careful and comprehensive recording of children's performance, we recorded the data of the children in the experiment by using the record table and video recorder. The researcher used recorder to record the child's specific conditions, such as help, irritability, and interruptions, etc. Video recorder is used to record the operation, behavior and facial expression of children in the programming process.

#### 4.5 Video data coding and quantization

**Table 2.** Group G1 behavior coding scheme

	Key	Code	Type	Description
1	P	Programming time	State event	From put down the first programming block to complete both tasks
2	T	Thinking time	State event	No discussion, no program, time of duration is more than 5s
3	H	Happiness	Point event	Cheerful behaviors including applauding, laughing, jumping and so on



4	C	Confusion	Point event	Be frustrated or overwhelmed, including scratching the head, frowning, sighing and pouting.
5	A	Asking for help	Point event	Children asking researchers for help
6	E	Trial-and-error of garbage classification	Point event	Try different kinds of classification
7	W	Trial-and-error of programming	Point event	Try a variety of programming blocks

**Table 3.** Group G2 behavior coding scheme

	Key	Code	Type	Description
1	P	Programming time	State event	From put down the first programming block to complete both tasks
2	T	Thinking time, left child	State event	No discussion, no program, time of duration is more than 5s
3	S	Thinking time, right child	State event	No discussion, no program, time of duration is more than 5s
4	H	Happiness, left child	Point event	Cheerful behaviors including applauding, laughing, jumping and so on
5	I	Happiness, right child	Point event	Cheerful behaviors including applauding, laughing, jumping and so on
6	C	Confusion, left child	Point event	Be frustrated or overwhelmed, including scratching the head, frowning, sighing and pouting.
7	D	Confusion, right child	Point event	Be frustrated or overwhelmed, including scratching the head, frowning, sighing and pouting.
8	A	Asking for help, left child	Point event	Children asking researchers for help
9	B	Asking for help, right child	Point event	Children asking researchers for help
10	E	Trial-and-error of garbage classification, left child	Point event	Try different kinds of classification
11	F	Trial-and-error of garbage classification, right child	Point event	Try different kinds of classification

12	W	Trial-and-error of programming, left child	Point event	Try a variety of programming blocks
13	Y	Trial-and-error of garbage classification, right child	Point event	Try a variety of programming blocks
14	R	Relevant discussion time	State event	Discuss something about task
15	Z	Irrelevant discussion time	State event	Communicate something except relevant discussion

Firstly, we extracted three groups' video data. In these three videos, the behaviors of the subjects were classified and coded, and labeled with BORIS, a software for video annotation [20]. The first was behavioral classification. In order to ensure the accuracy and reliability of classification, two researchers watched video data and recorded children's behavior categories in the experiment. Through watching video for many times and discussing the classification results, the kappa coefficient [21] of the two researchers was 0.78. The main classification of behavior included relevant discussion, irrelevant discussion, asking for help, programming, trial-and-error of programming, trial-and-error of garbage classification, thinking, confusion, happiness, etc.

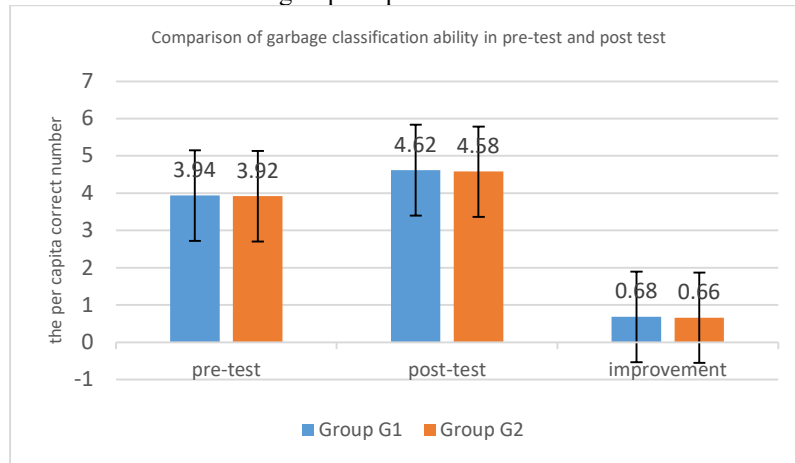
The second was behavior coding. According to the behavior classification scheme, the behavior coding scheme was designed by using BORIS, including behavior definition (Description), marking target objects, setting shortcut operations (Key), annotation event type (Type). Event types were divided into point event and state event. Because the number of children in two groups were different and the behavior categories were also different, the videos of Group G1 and Group G2 are also coded separately. The behavior coding map of Group G1 is shown in Table 2. On the basis of G1 behavior coding map, iterative coding is carried out to generate Group G2 coding map as shown in Table 3. Finally, according to the behavior coding map, we annotated the video of Group G1 and Group G2 respectively by using the BORIS software. We counted the frequency and duration of all kinds of behaviors and completed the quantitative analysis of all children's video data in the experiment. In the process, we found that there were four children in Group G1 whose video data had partial defects. We discarded their video data, so there were 11 children in Group G1 and 22 children in Group G2.

## 5 Result Analysis

In the following part, we will analyze the results of the questionnaire, record table, and video data. And we will answer the three questions raised in the section of Goals of Experiment.

- Question 1: Can children learn the knowledge of garbage classification through Clas-Maze, and whether there is a difference in the learning effect between single(G1) and double(G2) groups?

The pre-test questionnaire and post-test questionnaire both contain 5 questions for garbage classification ability, which test result is shown in Figure 5. In Group G1, the per capita correct number of garbage classification is 3.94 ( $SD = 0.67$ ) in pre-test; that is 4.62 ( $SD = 0.26$ ) in post-test. In Group G2, the per capita correct number of garbage classification is 3.92 ( $SD = 0.86$ ) in pre-test; that is 4.58 ( $SD = 0.60$ ) in post-test. The data comparison between the two groups is that the Group G1 improvement of the garbage classification ability is 0.02 higher than Group G2. We use One-way ANOVA to analyze the data between the two groups, with the grouping situation as the variable factor and the correct number of questions as the dependent variable. The results showed that the significance level  $P = 0.76$  between pre-test Group G1 and Group G2, that is greater than 0.05. So there is no significant difference in garbage classification ability between two groups in pre-test. The significance level between two groups is  $P = 0.91$  in post-test, that is greater than 0.05. So there also is no significant difference between two groups in post-test.



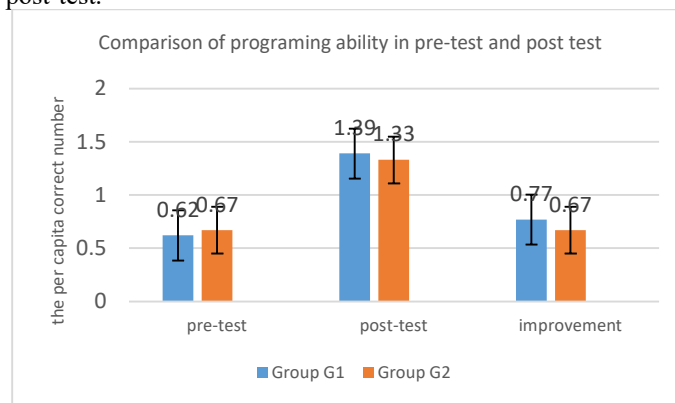
**Fig. 5.** comparison of garbage classification ability in pre-test and post test

There is a subjective question in the post-test questionnaire "Does this game help you to learn garbage classification?". 2 children think that the effect learning garbage classification is the same with or without this system. 13 children think that the system is a bit helpful. 22 children think it is very helpful. About pre-test and post-test data of all participants, we use the SPSS software to perform Non-parametric Test on relevant samples. The result is  $P = 3.815E - 6$ , which proved that the children's garbage classification ability has a very significant difference between pre-test and post-test.

Combining the date of garbage classification ability and subjective question, it is easy to know that children can learn garbage classification knowledge through this programming system. During complete tasks, children in Group G1 got more help and guidance from researchers, so the learning effect of the Group G1 is a little better.

- Question 2: Can children learn the programming knowledge through Clas-Maze, and whether there is a difference in the learning effect between single and double groups?

The pre-test questionnaire and post-test questionnaire both contain 2 questions for programming ability, whose test result is shown in Figure 6. In Group G1, the per capita correct number of programming is 0.62 ( $SD = 0.42$ ) in pre-test and is 1.39 ( $SD = 0.59$ ) in post-test. In Group G2, the per capita correct number of programming is 0.67 ( $SD = 0.41$ ) in pre-test and is 1.33 ( $SD = 0.32$ ) in post-test. The improvement of Group G2 is 0.10 higher than that of Group G2. We also use One-way ANOVA to analyze the data between the two groups. The results showed that the significance level,  $P = 0.82$  between Group G1 and Group G2 is greater than 0.05, that is, there is no significant difference in garbage classification ability between two groups, neither in pre-test or post-test. About pre-test and post-test data of all participants, we performed Non-parametric Test. The result showed that  $P = 3.815E - 6$  is far less than 0.05, that is, the children's programming ability has a very significant difference between pre-test and post-test.



**Fig. 6.** comparison of programming ability in pre-test and post-test

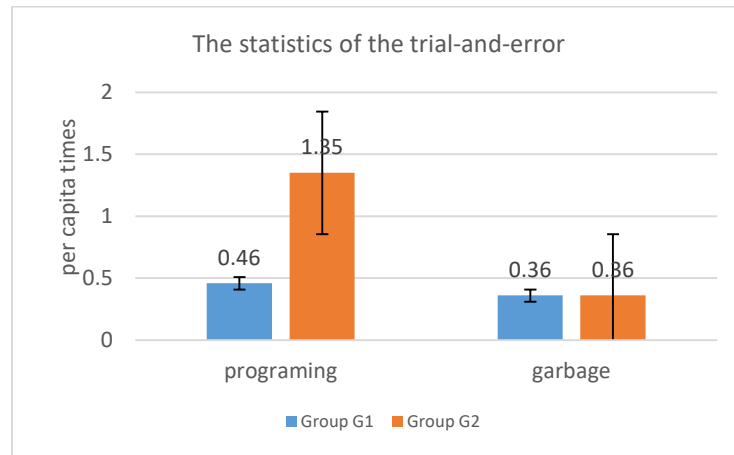
There is a subjective question in the post-test questionnaire "Does this game help you to learn programming?". 2 children think that this system is no helpful. 13 children think that it is a bit helpful. 22 children think that it is very helpful.

Combining the date of programming ability and subjective question, we found that children can learn programming knowledge through this programming system. In short period, children in Group G1 receive more instruction, so their performance is a little better at the aspect of learning effect.

- Question 3: Whether there is a difference in the learning performance between single and double groups?

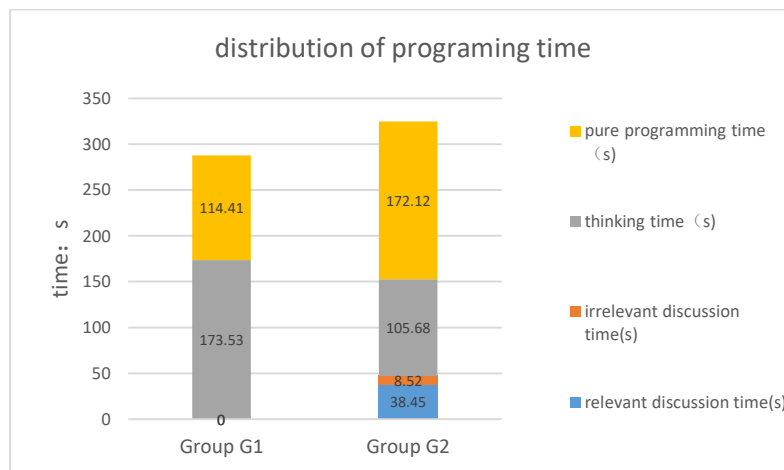
*Asking for Help.* According to the statistical analysis of the data in the record table, we can know that the per capita number of asking for help is 1.36 times in Group G1; it is 0.52 times in Group G2. Obviously, the number of G1 children asking researchers for help is significantly higher than that of Group G2.

*Trial-and-error of programming and garbage classification.* The definition of “programming trial-and-error” is that when child does not know which programming block to put, he tries a variety of programming blocks. The definition of “garbage classification trial-and-error” is that the child tries different kinds of classification, when he does not know a certain thing belong to what kind of garbage. The statistics of the trial-and-error is shown in Figure 7. In Group G1, the per capita number of programming trial-and-error is 0.46, and the per capita number of garbage classification trial-and-error is 0.36. In Group G2, the per capita number of programming trial-and-error is 1.35 (the No.6 child had 11 trial-and-errors), and the per capita number of garbage classification trial-and-error is 0.36. Obviously, the Group G2 has a higher number trial-and-error in programming and garbage classification. The reason is that a few children in Group G2 had higher number of garbage classification trial-and-error, which lowers the overall average.



**Fig. 7.** Statistics of the trial-and-error

*Programming time.* Programming time refers to the time from child puts down the first programming block to complete both tasks, so it consists of thinking time, relevant discussion time, irrelevant discussion time, and pure programming time. The thinking state is described as a state that child doesn't discuss or program and the duration is greater than 5s (we can sure that no child zone out during the experiment). Relevant discussion refers to the discussion among teammates about routes, garbage classification, use of programming blocks and other things related to the task. Irrelevant discussion refers to the communication between teammates except relevant discussion



**Fig.8.** Distribution of programming time

According to the video quantization data, we get the programming time distribution diagram as shown in Figure 8. In Group G1, averagely, the total programming time is 287.94s, and the thinking time is 173.53s, and the pure programming time is 114.41s. In Group G2, averagely, the programming total time is 324.77s, and the thinking time is 105.68s, and the pure programming time is 172.12s, and the relevant discussion time is 38.45s, and the irrelevant discussion time is 8.52s. In Group G2, there are three groups that their programming time is far higher than the per capita time. We made a detailed analysis of the three groups' video data and found that the participants had several disputes with their teammates during the programming process, such as scrambling for programming blocks and holding difference opinions about programming. Furthermore, we used Correlation Analysis to analyze the relationship between total programming time and other date. The results showed that the relevant discussion time and total programming time is extremely significant ( $\rho=0.63, P=0.002$ ). And the irrelevant discussion time and total programming time is significant ( $\rho=0.523, P=0.012 < 0.05$ ). Overall, the discussion among teammates lead the programming time of G2 to be longer than G1.

*Children's happiness.* The state of happiness is described as a set of emotionally cheerful behaviors including applauding, laughing, jumping and so on. Through the analysis of the video quantitative data, we got that the per capita number of happiness is 2.09( $SD = 0.89$ ) in Group G1; it is 3.86( $SD = 2.53$ ) in Group G2. The result showed that most children (regardless of the group) were in a happy state in the experiment. But the range of happy frequency in G2 is wider, and the per capita number of G2 is significantly higher than Group G1. We used the SPSS software to perform one-way ANOVA for the two groups' data, and there is a significant difference,  $P = 0.036$ . The children in Group G1 are happier in the cooperative learning.

*Children's confusion.* The state of confusion is described as children are frustrated or overwhelmed during programming, specific actions including but are not limited to scratching the head, frowning, sighing and pouting. The per capita number of confusion is 3.18( $SD = 2.96$ ) in Group G1 and 0.36( $SD = 2.17$ ) in Group G2 separately. It is easy to know that the number of confusion is lower in Group G2. It reflects that children in Group G2 can solve problems independently and timely, rather than rely on the experiment researchers.

*Children's subjective preference.* To investigate children's cooperative inclination in programming, the second question of the post-test questionnaire is "Do you prefer to program alone or cooperatively with a teammate?". We got the statistical analysis of data is shown that there was only 24.32% thought programming alone was better. But 75.68% of children thought programming with a teammate is better. Children who tend to program alone are generally older. They said that the game is simple and cooperating with others is unnecessary. Children who want a teammate said that it would be more fun if they had a partner to communicate. Further, we analyzed the personal data of children who were inclined to program alone and found that they had disagreements or disputes with their teammates during the cooperation. And they gave a low score to teammates, that is, the bad cooperative experience led them to want to program alone.

To sum up, children in Group G2 have a greater ability to solve problems through cooperation. They need less help from researchers, and they are more pleasant in the process of programming learning. But they averagely cost more time than those children in Group G1. So, we can judge the learning performance from two aspects. If the criterion was that child can get more knowledge in shorter time, the performance of Group G1 is better. If the criterion was that child can not only learn knowledge but also improve his or her decision-making and communication skills, and have a pleasant learning state, the performance of Group G2 is better.

## **6 Limitation and Discussion**

This experimental design also has some limitations. First, during the experiment, experimental equipment or children's arms may block the camera, which made the camera can't recognize the block. Although we can adjust the equipment quickly, it dispersed children's attention and commitment to learning in some degree. Secondly, variable control is not strict. Before the test, children in G1 were given one-to-one training, but children in G2 were one-to-two. So, this may lead children in G2 to learn knowledge more roughly. Asking a researcher can make children get the right answer more quickly than team discussion, which make G1 use a shorter time to finish tasks. In our study, 35 children were involved in. The small dataset has inherent limitations, and our results therefore more tend to hypotheses and inference rather than definite conclusions.

## 7 Conclusion and Future Work

The paper presented a new edutainment tool, Clas-Maze, which combines tangible programming and knowledge of garbage classification. Through user study, we evaluated the help of this programming system for children to learn programming and garbage classification knowledge and discussed the difference between Group G1 and Group G2.

All the children's ability of programming and garbage classification were improved significantly after the experiment, but there was no significant difference in the improvement effect between two groups. This paper evaluates the learning performance of children from six dimensions: trial and error, asking for help, programming time, happiness, confusion, children's subjective preference. The result showed that Group G1 and Group G2 have their own advantages. If the purpose of use is to let children obtain more knowledge in a short time, the system is more suitable for single use. If the purpose of use is not only to let children learn knowledge, but also to improve their ability of decision-making and communication in a pleasant learning environment, the system is more suitable for cooperative use.

Based on user feedback and analysis of experimental data, in the future, we will be to improve the programming system and the experimental design. First, we should control irrelevant variables more strictly, such as collective training at the stage of knowledge popularization. Second, Clas-Maze as an edutainment tool, we will integrate more living knowledge into it. Third, to better demonstrate the usefulness of the system, we will add a control group that use other learning methods, for example teaching by teacher or watching videos.

## ACKNOWLEDGMENTS

This research is supported by the National Natural Science Foundation of China under Grant No. 61872363, 61672507, the National Key Research and Development Program under Grant No. 2016YFB0401202, and the Research and Development Fund of Institute of Automation, CAS under Grant No. Y9J2FZ0801.

## References

1. Wing J M. Computational thinking[J]. *Communications of the ACM*, 2006, 49(3):33-35.
2. Council N. Report of a Workshop on the Scope and Nature of Computational Thinking[J]. National Academies Press, (2010).
3. Xiaozhou D. Design and implementation of tangible programming system based on distributed cognition [D]. Institute of Automation, University of Chinese Academy of Sciences. (2019).
4. Alvarado C. CS Ed Week 2013: the hour of code[J]. *ACM Sigcse Bulletin*, 46(1):2-4(2014).
5. Hongyan W, Yuhe T. UK: Programming education into the national Curriculum [J]. *Shanghai Education*, 2016(2):20-23(2016).



6. New Generation Artificial Intelligence Development Planning. Guofa No. 35. China's State Council(2017).
7. Tingting W, Danli W, Lu L, et al. Children-oriented graphical programming language and tools [J]. *Journal of Computer-Aided Design & Computer Graphics* (4): 154-161(2013).
8. Mitchel R, John M, Andrés Monroy-Hernández, et al. Scratch: Programming for All[J]. *Communications of the ACM*, 52(11):60-67(2009).
9. Horn, M., Jacob, R.J.K. Tangible Programming in the Classroom with Tern. In Proc SIGCHI 2007, ACM Press (2007).
10. Revelle, G., Zuckerman, O., Druin, A. and Bolas, M. Tangible User Interfaces for Children. In Proc. CHI EA 2005, ACM Press, 2051-2052 (2005).
11. Notice on promoting the management of domestic garbage classification in schools. General Office of Chinese Ministry of Education (2018).
12. Kelleher, C. and Pausch, R. Lowering the Barriers to Programming: A Taxonomy of Programming Environments And Languages for Novice Programmers. *ACM Computing Surveys*, 83-137 (2005).
13. Kelleher, C, and Pausch, R. Using storytelling to motivate programming. *Communications of the ACM* 50, 7, 58-64 (2007).
14. Wing J M. Computational thinking and thinking about computing[J]. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 366(1881):3717-3725(2008).
15. Gallardo, D., Juli'a, C.F. and Jord'a, S. TurTan: a Tangible Programming Language for Creative Exploration. In Proc. Third annual IEEE international workshop on horizontal human-computer systems. IEEE Press (2008), 412-420(2008).
16. Wang, D.L., Zhang, C., Wang, H.A. T-Maze: a tangible programming tool for children. In Proc. IDC 2011. ACM Press, 127-135 (2011).
17. Xiuqin Z. How to Cultivate Children's Self-care Ability [J]. *Examination Weekly*, (87):193-193(2016).
18. Miao Z. Educational Analysis on the Development of Garbage Sorting in Kindergartens in Japan [J]. *Education Guide (Second Half)*, (10): 87-90(2015).
19. Qiao J. Research and implementation of children's physical programming system based on augmented reality [D]. Institute of Automation, University of Chinese Academy of Sciences (2019).
20. Friard, O. Gamba, M. BORIS-Behavioral Observation Research Interactive Software: Italy. (2015).
21. Viera A J, Garrett J M. Understanding interobserver agreement: the kappa statistic[J]. *Fam med*, 37(5): 360-363(2005).
22. Qi Y, Wang D, Zhang L, et al. TanProStory: A Tangible Programming System for Children's Storytelling.[C]// *ACM Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, (2015).
23. Lihui S, Danhua Z. Current situation and Action Path of International Children programming Education research [J]. *Open Education Research*, 25(02):25-37(2019).
24. Junnan Y and Ricarose R. A survey of computational kits for young children. In *Proceedings of the 17th ACM Conference on Interaction Design and Children (IDC '18)*. Association for Computing Machinery, New York, NY, USA, 289–299(2018). DOI:<https://doi.org/10.1145/3202185.3202738>
25. Fails J A, Druin A, Guha M L, et al. Child's play: a comparison of desktop and tangible interactive environments. *Proceedings of the 2005 conference on Interaction design and children*. ACM: 48-55(2005).

26. Felix H, Ariel Z, Horn, M, and Judd, F. Strawbies: explorations in tangible programming. In Proceedings of the 14th International Conference on Interaction Design and Children (IDC '15). Association for Computing Machinery, New York, NY, USA, 410–413(2015).
27. Manches A, O'Malley C. Tangibles for learning: a representational analysis of physical manipulation[J]. *Personal & Ubiquitous Computing*, 16(4):405-419(2012).
28. Md M R, Monir S, and Roshan P. 2020. Impact of Infusing Interactive and Collaborative Learning in Teaching Introductory Programming in a Dynamic Class. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (SIGCSE '20). Association for Computing Machinery, New York, NY, USA, 1315. DOI:<https://doi.org/10.1145/3328778.3372608>
29. Deng X , Wang D , Jin Q . CoProStory: A Tangible Programming Tool for Children's Collaboration[C]// International Conference on Computer Supported Collaborative Learning(CSCL2019). 2019.