

Research on the Input Methods of Cardboard

Zhiyi Ma[1,2(✉)], Hongjie Chen[1,2], Yanwei Bai[1,2], and Ye Qiu[1,2]

[1] School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China
{mazhiyi, chen.hj, byw, qiuye2014}@pku.edu.cn
[2] Key Laboratory of High Confidence Software Technologies, Peking University, Beijing 100871, China

Abstract. Virtual Reality (VR) is a new technology developed in the 20th century and the demands for VR from all walks of life are increasing, and Cardboard plays an important role in the demands. Facing the current situation that the existing Cardboard input methods cannot meet the actual needs well, aiming at the four typical input methods of Cardboard: buttons, gaze, voice and gestures, the paper conducts an experiment on typical object selection tasks and analyzes the experimental results in the sparse distribution, dense distribution, and sheltered dense distribution. The research results have important practical value for the design and development of the input of Cardboard based applications.

Keywords: Cardboard · Input method · Virtual Reality

1 Introduction

Virtual reality is characterized by immersion, which means that users feel like they are there. In virtual reality applications, output devices and input devices are necessarily. For output devices, displays are the most core parts of virtual reality applications; for input devices, the usual mouses and keyboards have become almost unusable. For this problem, the researchers have proposed many solutions.

Google introduces the concept of Cardboard VR in 2014, and soon afterwards, Cardboard becomes a very cheap head-mounted virtual reality device that can be used with most smartphones. Because of its versatility, this paper chooses it as the support tool for building VR applications. Although Cardboard has the advantages of low price and wide applicability, the disadvantages are also prominent. Because Cardboard's built-in input method uses only one button, the interaction of users with it is very limited. Most Cardboard-based VR applications only provide the viewing function [1], and it is difficult to have further interaction, and therefore the experience of users with it much worse than that of using traditional virtual reality devices.

Although there are currently some studies trying to provide new input methods for Cardboard or compare performance indicators of the existing input methods, these studies all have their own shortcomings. Some studies focus on the design of input methods and lacks research on user feedback [2–4], and others only compare time and accuracy of several input methods in simpler object selection tasks [5, 6].

© Springer Nature Switzerland AG 2020
O. Gervasi et al. (Eds.): ICCSA 2020, LNCS 12250, pp. 18–29, 2020.
https://doi.org/10.1007/978-3-030-58802-1_2

Aiming at the lack of Cardboard input methods and the deficiencies of related researches, the paper conducts an experiment to study the variables related to input methods based on the research [7].

2 Related Work

The researches on Cardboard input methods can be divided into two categories: new interaction of Cardboard, and empirical researches on existing interaction.

2.1 Design of Cardboard Input Methods

Among all the input methods of Cardboard, in addition to the built-in key input, the most popular and successful input method is gaze [2]. Its input process is that a user first uses the focus ray to find an object to be interacted with, then a timer called as a gaze ring appears at this time, and finally the selection operation is triggered when the gaze ring is filled. This method brings convenience for input, but also has some disadvantages. For examples, using the countdown method to let the user choose whether to input or not, it will cause a certain degree of tension, which will reduce the user experience; the user may be mistakenly input due to unfamiliar how to use it or inattention, and it is easy to maintain their concentration so as to fatigue; in virtual reality with high density of objects, frequent appearance of gaze rings will also affect the user's operations; the countdown delays the user's input process, causing unnecessary waits, and it is also difficult to fix the specific delays.

Majed, A.Z. et al. proposed a Cardboard input method, PAWdio [3], which tracks the relative positions of two earbuds for input. Compared with ordinary key input, PAWdio allows users to have a stronger sense of immersion with better accuracy and efficiency. However, this method has only one degree of freedom in operation, and is still affected by the arm span and the length of the headphone cable when in use. In addition, the operation is very different from the ways people operate in the real world.

Based on RGB cameras, Akira, I. et al. proposed an input method [4], which uses the camera in a smart phone for tracking a thumb. When the thumb is raised, the cursor movement in the virtual reality can be controlled by the thumb, and the selection is made when the thumb is pressed. The authors research the method on an object selection task and find that the method has better accuracy and efficiency when the object is larger, but the error rate increases significantly when the object is smaller. The advantage of this method is that gesture input can be completed without using an additional input device, and it works well in some specific scenarios. However, because the control is performed only through a thumb, the input is still limited to the simulation of clicks, and more complex information cannot be input.

Pfeuffer, K. et al. discuss the effect of the object distribution density on user input in gaze mode [8], and the paper takes the object distribution density as a control variable together with more input methods.

2.2 Comparison of Cardboard Input Methods

Tanriverdi,V. et al. study the performance of using focus rays and an arm-extended gripping tool on object selection tasks, and find that using focus rays has an efficiency advantage [5]. Cournia, N. et al. compare the efficiency of using focus rays with the use of handheld controllers in object selection tasks, and find that the use of controllers is more efficient when selecting long-range targets [6]. These efforts focus on the comparison between focus rays and virtual hands, but ignores the comparison between buttons and gaze, which are two most common input methods. In addition, the tasks do not consider the influence of factors such as the distribution density of objects and the shapes of objects on the users' selection of input methods.

Ganapathi, P. et al. first compare buttons and gaze in the different dense distributions of objects, and obtain the following conclusions: the needed time and error rate of gaze input are significantly higher than that of button input in the sheltered dense distribution; in the sparse distributions, users with gaze are more comfortable; in the sheltered dense distributions, button is easier to use. Taken together, they believe that gaze should be used in the sparse distribution and buttons should be used in dense or sheltered dense distributions [7]. Their research is limited to the traditional two input methods, lacking a comparison of other input methods that do not use additional controllers. In addition, their research is limited to square-shaped objects and lacks analysis of other shapes of objects.

3 Experimental Design

3.1 Subjects

There are 20 subjects in this experiment, all of them are undergraduates, and they have no obstacles in using Cardboard's key input and gesture input, etc.

3.2 Experimental Equipment

Two devices used in the experiment are a Cardboard[1] and an Honor 8 smartphone. The main performance parameters of the phone are as follows: the screen size is 5.2 in., the resolution is 1920 * 1080, the operating system is Android 8.0, the processor is Kirin 950, and the storage is 4G RAM and 64G ROM.

3.3 The Experimental Environment

The experimental tool for recording data is developed with Unity 2017.4[2] and Google VR SDK For Unity 1.4[3] [9]. Some changes ARE made based on the work [7].

[1] This device uses engineering plastics as the main structure and is functionally no different from a traditional Cardboard.

[2] https://unity3d.com/cn.

[3] https://developers.google.cn/vr/develop/unity/.

The whole experiment consists of two parts, the first part is training and the second part is testing. The training is relatively simple for subjects to familiarize four different input methods.

The test is used to collect experimental data and is complicated. It mainly includes 6 different scenarios, which are composed of squares or ellipsoids in the sparse distribution, dense distribution, or sheltered dense distribution, respectively, see Fig. 1.

The objects in each scenario are divided into two parts, the shape parameters of the two groups of objects on the upper side are the same, and the shape parameters of the group of objects on the lower side are different from those on the upper side. In the sparse distribution and dense distribution, the color of the objects is black and white; in the sheltered dense distribution, the color of the shelters is red, and the other colors are the same. All objects are located on the same plane in a scenario, 47 m from the camera (the default distance unit in Unity, the same below), the two groups of objects on the upper side are 62 m from the ground, and the objects on the lower side are 47 m from the ground. In addition, the relative squares and ellipsoids in the scenarios have the same positions on the three coordinate axes.

3.4 Experimental Task

Focusing on four typical input methods of Cardboard: buttons, gaze, voice and gestures in the sparse distribution, dense distribution, and sheltered dense distribution of objects (balls and cubes), the paper analyzes whether there are significant differences in the following indicators:

- the number of user errors?
- the user time to complete the tasks?
- the user convenience?
- the user satisfaction?

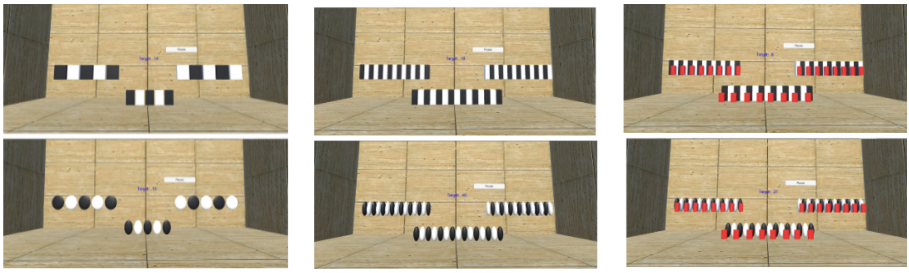
In the above analyses, if there are differences, it is necessary to analyze the impact of the variables.

- What input methods do users prefer?

Among them, user time to complete the tasks and the number of user errors are automatically calculated by our experimental tool [9]; user convenience, user satisfaction, and user preference are obtained by filling in questionnaires after the experiment, and the questions are designed with the 5-point semantic differentiation scale method.

3.5 Design of Experimental Input Methods

The Cardboard input methods involve in the paper include buttons, gaze, voice and gestures. The button input is Cardboard's own function, and is completed by clicking the button on the right of Cardboard to simulate a click on the smartphone screen; gaze input is to fill the gaze ring that appears after the focus ray collides with the target to confirm the user's selection intention, and the system automatically completes the selection when the gaze ring is filled. These two input methods are more common and



(a) Sparse distribution (b) Dense distribution (c)Sheltered dense distribution

Fig. 1. Three distributions of two kinds of objects

will not be described in the paper. The following mainly focuses on voice input and gesture input.

In virtual reality, voice inputs are mainly divided into two categories. One is to simply trigger the corresponding operation by keywords; another is to perform the corresponding operation after semantic analysis of the speech, and the analysis is relatively more complicated. In order to compare with the traditional input methods, the paper uses a combination of focus rays and voice input to select objects. Specifically, the selection process is to find the object to be selected with the focus ray, and then say the command "Pick" to complete the selection. The command word detection uses the voice wake-up function in the voice SDK provided by iFlytek[4].

As far as we know, on the Cardboard platform, there is no related work on analyzing the performance and user feedback of object selection tasks in the different dense distributions of objects, combined gesture input based on RGB cameras with traditional input methods. In order to study this problem, the paper designs and implements an input method combining focus rays with gesture recognition based on RGB cameras. The application process is to first find the object to be interacted with the focus ray, and then make a corresponding gesture to select the object (or execute the instruction bound to the gesture).

Based on typical gesture recognition algorithms and our simple experiment, we design a gesture recognition algorithm and implements it using Opencv For Unity[5]. The algorithm has a good recognition effect in the experiment, and there is no sense of delay at a frame rate of 30 frames/s [9].

4 Experimental Results and Analysis

In the following experiment, in order to further explore the relationship between the indicators in Sect. 3.4 and the effect variables, the paper uses a two-factor analysis of variance with a significance level of 0.05 to compare whether the difference in each

[4] https://www.xfyun.cn/.

[5] https://assetstore.unity.com/packages/tools/integration/opencv-for-unity-21088.

indicator is statistically significant when selecting different shapes of objects with different input methods in the different dense distributions of objects.

4.1 Number of Errors

The selection error refers to that the object selected by the user does not match the target. The mean of the number of errors is shown in Fig. 2.

It can be intuitively seen from the Fig. 2 that the average number of errors of gestures and gaze is more, and the average number of errors of buttons and voice is less.

In three distributions,

1) The interaction of input methods and shapes is not significant. Their F and P values, respectively, are ($F(3, 152) = 2.586$, $P > 0.05$), ($F(3,152) = 0.792$, $P > 0.05$), and ($F(3,152) = 0.795$, $P > 0.05$).
2) The main effects of the input methods are significant. Their F and P values, respectively, are ($F(3,152) = 9.929$, $P < 0.05$), ($F(3, 152) = 15.851$, $P < 0.05$), and ($F(3, 152) = 6.575$, $P < 0.05$), and this indicates that there is a significant difference in the number of errors in different input methods in the dense distribution. Turkey's post hoc tests show that the number of gesture input errors is significantly higher than the other three input methods. In addition, there are no significant differences between the other input methods.

When the objects are sparsely distributed, the main effect of the object shapes is also significant ($F(1,152) = 9.571$, $P < 0.05$). According to its mean values, it can be seen that the number of errors when selecting ellipsoids is significantly smaller than the number of errors when selecting blocks. The main effect of object shapes is not significant when the objects are densely distributed and sheltered densely distributed (($F(1,152) = 0.090$, $P > 0.05$), ($F(1,152) = 0.429$, $P > 0.05$)).

There are two usual reasons of errors. One is that the user may inadvertently move the cursor over a non-target object and the selection is triggered automatically after staying because of the user's inattention, which is often found in gaze input method. Another reason for an error is mainly caused by the "Heisenberg effect". For example, when the user uses gesture input to focus on the gesture, and ignore the cursor has moved to the wrong target, resulting in wrong selection.

4.2 Time to Complete the Tasks

The time to complete the task refers to the time from the start of countdown to the user's correct choice in a scenario. The longer it takes, the less convenient it is for the user to use the input method. The data distribution of the average time to complete the tasks is shown in Fig. 3.

As can be seen from the Fig. 3, the time to complete the tasks gradually rises in the sparse, dense, and sheltered dense distributions. In terms of each denseness, the time required for gesture input is the longest, and the gap between the other three input methods is not large.

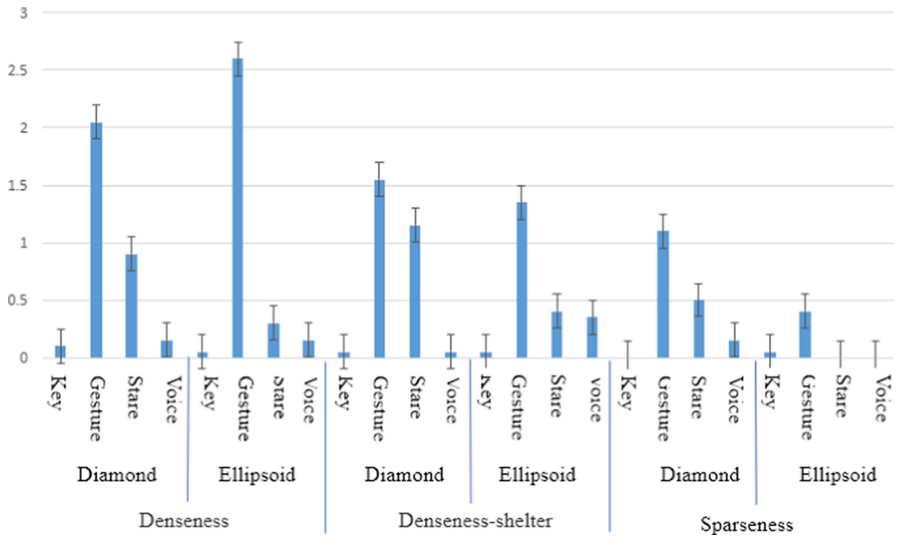


Fig. 2. Average number of errors

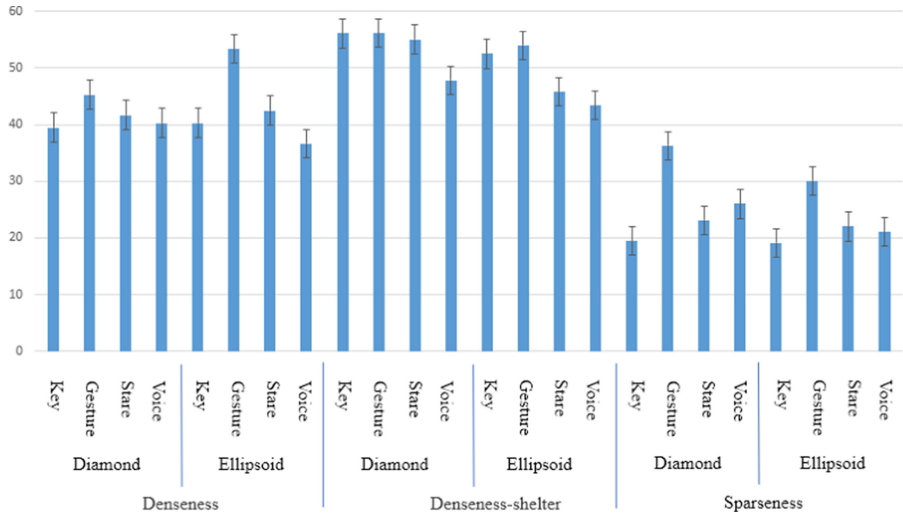


Fig. 3. Average time to complete the tasks

In terms of the interaction between the input methods and the shapes of the objects, the analysis of variance shows that the interactions in the three distributions are not significant, and their F and P values, respectively, are ($F(3, 152) = 2.082$, $P > 0.05$), ($F(3, 152) = 0.988$, $P > 0.05$), and ($F(3, 152) = 0.226$, $P > 0.05$).

For input methods, when the objects are sparsely distributed and the objects are densely distributed, the main effects of the input methods are significant, and their F

and P values, respectively, are (F (3,152) = 35.781, P < 0.05), and (F (3,152) = 3.914, P < 0.05). This indicates that the time to complete the task of different input methods are significant differences, and Turkey's post hoc multiple tests also show that gesture input takes significantly longer to complete the task than the other three input methods. The main effect of input methods is not significant in the sheltered dense distribution (F (3,152) = 1.783, P > 0.05).

For object shapes in the sparse distribution, the main effect of them is significant (F (1,152) = 9.836, P < 0.05). According to their means, it can be found that the time for selecting an ellipsoid is significantly less than the time for selecting a block; when the objects are densely distributed and sheltered densely distributed, the main effect of the object shapes is not significant, and their F and P values are (F (1,152) = 0.386, P > 0.05) and (F (1,152) = 2.247, P > 0.05), respectively.

4.3 Comfort

Comfort refers to whether the user will feel uncomfortable when using a specific input method to make a selection, such as feeling tired when using button or gesture input, feeling nervousness when using gaze input, and feeling awkward when using voice input. After the experiment, comfort information was collected by filling in a questionnaire, which related questions are designed using a 5-point semantic differentiation scale (a score of 0 indicates uncomfortable and a score of 5 indicates very comfortable). The average comfort of various input methods is shown in Fig. 4.

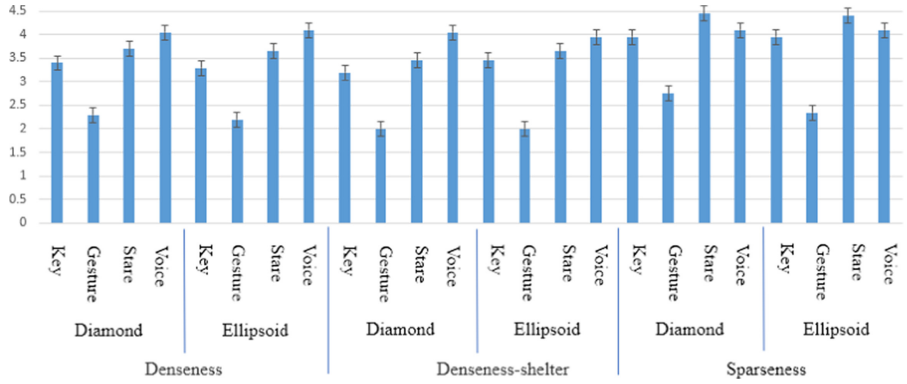


Fig. 4. Average comfort

It can be seen from Fig. 4 that users using gaze input in the sparse distribution have the highest comfort level, and voice input comfort is higher in the other two density distributions. In addition, there is a small difference in comfort when selecting different objects.

In three distributions, the analysis of variance shows that the interaction between the input methods and the shapes of the objects is not significant. Their F and P values, respectively, are (F (3, 152) = 0.431, P > 0.05), (F (3, 152) = 0.069, P > 0.05), and (F (3, 152) = 0.228, P > 0.05).

For input methods, in three distributions, the main effects of them are significant. Their F and P values, respectively, are ($F(3,152) = 31.705$, $P < 0.05$), ($F(3,152) = 33.943$, $P < 0.05$), and ($F(3,152) = 24.746$, $P < 0.05$). This indicates that there are significant differences in the time required to complete the tasks with different input methods. Turkey's post hoc multiple tests also show that the user comfort of gesture input is significantly lower than that of the other three inputs, the comfort of button input is significantly lower than that of voice input, and there is no significant difference between the other input method pairs.

For object shapes, the main effects of object shapes are not significant in three distributions, and their F and P values, respectively, are ($F(1,152) = 0.585$, $P > 0.05$), ($F(1,152) = 0.138$, $P > 0.05$), and ($F(1,152) = 0.256$, $P > 0.05$).

4.4 Convenience

Convenience refers to whether the user is convenient when making a selection using a specific input method. For example, both button input and gesture input require the user to move his hand, gesture input also requires the user to focus on both the hand and the target to be selected, gaze input requires the user to locate quickly when moving on the object, and voice input has certain requirements for the user's command word pronunciation. After the experiment, convenience information was collected by filling in a questionnaire, which related questions are designed using a 5-point semantic differentiation scale (a score of 0 indicates inconvenience and a score of 5 indicates convenience). The average convenience of various input methods is shown in Fig. 5.

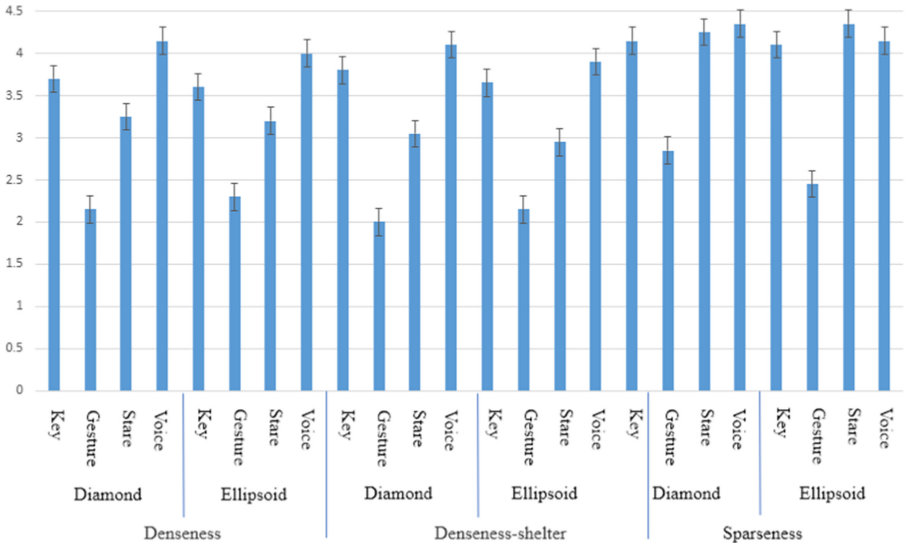


Fig. 5. Average convenience

It can be seen from Fig. 5 that the convenience of gesture input is the lowest, gaze input is the most convenient when the objects are sparsely distributed, and the convenience of voice input is more convenient in other two distributions. In addition, there are small differences in convenience when selecting different objects.

In three distributions, the analysis of variance shows that the interaction between the input methods and the shapes of the objects is not significant. Their F and P values are ($F(3, 152) = 0.447$, $P > 0.05$), ($F(3, 152) = 0.207$, $P > 0.05$), and ($F(3, 152) = 0.202$, $P > 0.05$), respectively.

For input methods, the main effects are significant in three distributions, and their F and P values are ($F(3,152) = 26.186$, $P < 0.05$), ($F(3,152) = 30.009$, $P < 0.05$), and ($F(3,152) = 24.762$, $P < 0.05$), respectively. This indicates that there are significant differences in convenience in different input methods. Turkey's post hoc tests also show that the convenience of gesture input is significantly lower than that of the other three input methods, and there is no significant difference between the other input methods.

For object shapes, the main effects are not significant in three distributions, and their F and P values are ($F(1,152) = 0.791$, $P > 0.05$), ($F(1,152) = 0.021$, $P > 0.05$), and ($F(1,152) = 0.188$, $P > 0.05$), respectively.

4.5 User Preference

User preference refers to that if the user chooses only one input method, which one he will choose. Considering that it is unlikely to design different input methods for objects of different shapes in an application, the questionnaire is designed only for user preferences in three distributions. The statistics are shown in Fig. 6.

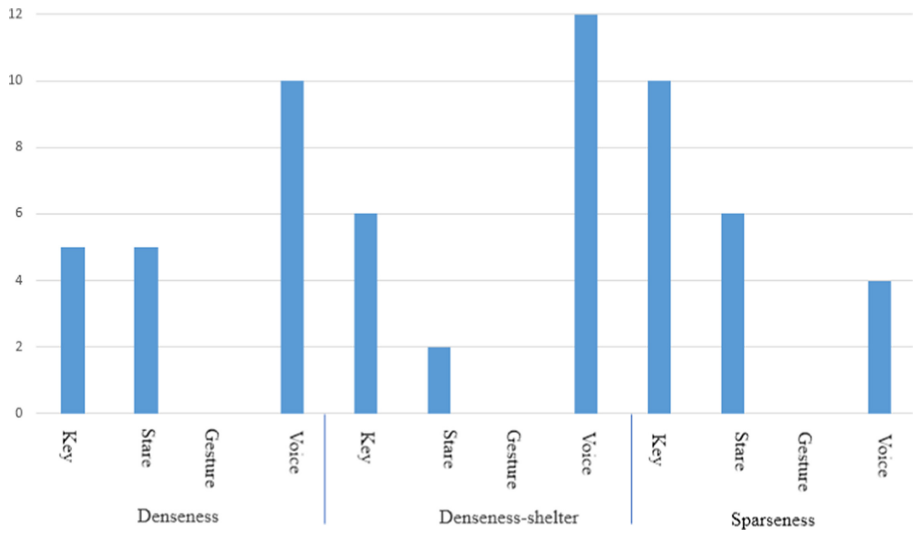


Fig. 6. User preferences in three distributions

It can be seen from Fig. 6 that no subjects choose gesture input, most of the subjects choose key input in the sparse distribution of objects, and most of the subjects choose voice input in the other two distributions.

5 Discussion

From the experimental results in Sect. 4, when the objects are sparsely distributed, the input methods and object shapes have a significant impact on the indicators recorded by the tool [9], such as the number of errors and task completion time, and the shapes of the objects have no significant effect on the comfort and convenience.

When objects are densely distributed, gesture input performs the worst and voice input does best, and there is no significant difference between button input and gaze input. The conclusion is similar in the sheltered dense distribution, and the gesture input performs the worst and the speech input performs best. The reason for the poor performance of gesture input is that users not only focus their attention on the objects to be selected, but also pay attention to the positions and postures of the hands. This affects the users' immersion, and it is also easy for the users to feel fatigue during use; that is, users' convenience and comfort are affected.

In addition, the research results show that there is no significant interaction between object shapes and input methods.

It can also be seen from the experiment that when the objects are sparsely distributed, if users want to perform simple object selection tasks, Cardboard's built-in button input method is enough. When the object is densely distributed or densely distributed with shelters, voice input should be used instead of button input. In addition, if the interactions are complex, voice input is preferred over gesture input.

From the interviews after the experiment, almost no subjects feel that the shapes of the objects have any effect on their choice, but according to the data recorded by our tool [9], the shapes of the objects still have a significant effect. The specific reason for this needs to be further studied.

The subjects in the experiment consisted of 9 boys and 11 girls, and the gender distribution is relatively even. The subjects are all college students and their ages range from 20 to 28 years old, and the age distribution is relatively narrow, but they are more representative among the people using VR applications.

6 Conclusion

The paper studies the input methods of Cardboard, including the experimental design and implementation, the experimental results and analysis, and the corresponding conclusions.

At present, there are other gesture input methods based on smart phone's RGB cameras. For these input methods, further experiments are needed for research. Moreover, the paper only explores the effect of object shapes on the study variables, and does not consider the effect of object size, therefore, it is necessary to further explore whether there is the interaction between object sizes and shapes, and how the interaction affects the tasks of object selections.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (No. 61672046).

References

1. Barbagallo, R.: How to Support Gear and google cardboard in one unity 3D project. http://ralphbarbagallo.com/2015/05/26/how-to-supportgear-vr-and-google-cardboard-in-one-unity3d-project/
2. Anthony, S., Simon, J.: Design and implementation of an immersive virtual reality system based on a smartphone platform. In: IEEE Symposium on 3D User Interfaces on Proceeding, pp. 43 – 46. IEEE (2013)
3. Majed, A.Z., Sam, T., Eelke F.: PAWdio: hand input for mobile VR using acoustic sensing. In: 2016 Annual Symposium on Computer-Human Interaction in Play on Proceedings, pp. 154–158. ACM (2016)
4. Akira, I., Takuya, A., Keigo, S., Shuta, N.: FistPointer: target selection technique using mid-air interaction for mobile VR environment. In: 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems, pp. 474. ACM (2017)
5. Tanriverdi, V., Robert J.J.: Interacting with eye movements in virtual environments. In: The SIGCHI Conference on Human Factors in Computing Systems on Proceedings, pp. 265–272. ACM (2000)
6. Cournia, N., John. D.S., Andrew T.D.: Gaze- vs. hand-based pointing in virtual environments. In: ACM Conference on Human Factors in Computing Systems on Proceedings, pp. 772–773. ACM (2003)
7. Ganapathi, P., Keyur, S.: Investigating controller less input methods for smartphone based virtual reality platforms. In: 20th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct on Proceedings, pp. 166–173. ACM (2018)
8. Pfeuffer, K., Mayer, B., Mardanbegi, D., Gellersen, H.: Gaze + Pinch interaction in virtual reality. In: The 5th Symposium on Spatial User Interaction on Proceedings, pp. 99–108. ACM (2017)
9. Yanwei, B.: Design and implementation of a server cluster monitoring tool based on virtual reality. Master's thesis, Peking University (2018)