

Evaluation of an Eye Tracking Selection Technique with Progressive Refinement

by

Yunhan Wang

Department of Mechanical Engineering and Materials Science  
Duke University

Date: \_\_\_\_\_

Approved:

\_\_\_\_\_  
Regis Kopper, Supervisor

\_\_\_\_\_  
Mary Cummings

\_\_\_\_\_  
Michael Zavlanos

\_\_\_\_\_  
Leila Bridgeman

Thesis submitted in partial fulfillment of  
the requirements for the degree of  
Master of Science in the Department of  
Mechanical Engineering and Materials Science in the Graduate School  
of Duke University

2018

ABSTRACT

Evaluation of an Eye Tracking Selection Technique with Progressive Refinement

by

Yunhan Wang

Department of Mechanical Engineering and Materials Science  
Duke University

Date: \_\_\_\_\_

Approved:

\_\_\_\_\_  
Regis Kopper, Supervisor

\_\_\_\_\_  
Mary Cummings

\_\_\_\_\_  
Michael Zavlanos

\_\_\_\_\_  
Leila Bridgeman

Thesis submitted in partial fulfillment of  
the requirements for the degree of  
Master of Science in the Department of  
Mechanical Engineering and Materials Science in the Graduate School  
of Duke University

2018

Copyright by  
Yunhan Wang  
2018

## **Abstract**

We designed a novel eye tracking selection technique with progressive refinement - eye-controlled sphere-casting refined by quad-menu (EyeSQUAD) selection technique. We provided the closest target approximation and space partition methods to obtain stabilized eye movement control with relatively low computational cost. Through a user study, we evaluated the performance of this new technique with comparison of two previous selection techniques - ray-casting and SQUAD under different target size and distractor density scenarios. Results show that the EyeSQUAD technique can achieve similar selection speed as ray-casting and is more accurate than ray-casting although less precise and slower than SQUAD. Finally, we summarized several insights for designing interaction techniques with eye tracking.

# Contents

|   |     |
|---|-----|
| Abstract .....  | iv  |
| List of Figures .....   | vii |
| Acknowledgements .....  | ix  |
| 1. Introduction .....   | 1   |
| 2. Related Works .....  | 6   |
| 2.1 Previous selection techniques.....                          | 6   |
| 2.2 Eye tracking techniques in human-computer interaction ..... | 8   |
| 2.3 Eye tracking techniques in virtual reality .....            | 10  |
| 3. EyeSQUAD Selection .....                                     | 12  |
| 3.1 Closest target approximation .....                          | 16  |
| 3.2 Space partition .....                                       | 18  |
| 4. Methods.....   | 21  |
| 4.1 Experiment design .....                                     | 21  |
| 4.1.1 Goals and Hypotheses.....                                 | 21  |
| 4.1.2 Design .....  | 22  |
| 4.2 Apparatus.....  | 23  |
| 4.3 Participants.....   | 25  |
| 4.4 Procedure.....  | 25  |
| 4.5 Pilot Study .....   | 27  |
| 5. Results .....  | 29  |

|  |    |
|--|----|
| 5.1 Mean Error Rate .....                          | 29 |
| 5.2 Average Selection Time.....                    | 38 |
| 5.3 User Preference.....                           | 47 |
| 6. Discussion .....                                | 50 |
| 7. Conclusion .....                                | 54 |
| Appendix A Informed Consent Form.....              | 57 |
| Appendix B Color Blindness Assessment.....         | 60 |
| Appendix C Background Survey .....                 | 61 |
| Appendix D Ray-casting Questionnaire .....         | 63 |
| Appendix E SQUAD Questionnaire .....               | 64 |
| Appendix F EyeSQUAD Questionnaire .....            | 65 |
| Appendix G Overall Performance Questionnaire ..... | 66 |
| Appendix H Experiment Oral instructions.....       | 67 |
| References .....                                   | 71 |

## List of Figures

|   |    |
|---|----|
| Figure 1: Main scene (left) and QUAD-menu selection scene (right) .....                                       | 14 |
| Figure 2: Schematic of the closest target approximation method.....   | 16 |
| Figure 3: Schematic of the space partition method .....   | 18 |
| Figure 4: Experiment participant and FOVE headset (with an HTC Vive tracker) and<br>HTC Vive controller. .... | 24 |
| Figure 5: Mean error rate with different techniques.....  | 31 |
| Figure 6: Mean error rate with different target sizes.....  | 32 |
| Figure 7: Mean error rate with different distractor densities. ....   | 33 |
| Figure 8: The interaction between technique and target size when fixing technique. ....                       | 34 |
| Figure 9: The interaction between technique and target size when fixing target size. ....                     | 35 |
| Figure 10: The interaction between technique and distractor density when fixing<br>technique.....             | 36 |
| Figure 11: The interaction between technique and distractor density when fixing<br>distractor density. ....   | 37 |
| Figure 12: Average selection time with different techniques.....  | 39 |
| Figure 13: Average selection time with different target sizes.....  | 40 |
| Figure 14: Average selection time with different distractor densities. ....                                   | 41 |
| Figure 15: Interaction between technique and target size when fixing technique.....                           | 42 |
| Figure 16: Interaction between technique and target size when fixing target size.....                         | 43 |
| Figure 17: Interaction between technique and distractor density when fixing technique.<br>.....               | 44 |
| Figure 18: Interaction between technique and distractor density when fixing distractor<br>density.....        | 46 |

Figure 19: Overall user preference for the selection techniques. ....47

Figure 20: Overall average ratings of techniques. ....48

## **Acknowledgements**

First, I need to thank my parents and friends. Without supports and encouragements from them, I cannot conquer all the difficulties successfully. I want to thank my advisor Dr. Regis Kopper for his guidance and encouragements through most of my master's life. From his course, I began to meander in the field of virtual reality and be captivated by interaction techniques. Then I want to thank all professors in my thesis defense committee – Dr. Mary Cummings, Dr. Michael Zavlanos and Dr. Leila Bridgeman for their unconditional help and advice along my thesis project. Besides, I want to thank David J. Zielinski and Zekun Cao for their invaluable suggestions and helps during my development period. Last but not least, I want to thank all participants in the experiment for their voluntarily help for the study.

# 1. Introduction

3D interaction techniques are methods that provide users with good experience in virtual reality which allow them not only to see but also interact with virtual contents. Manipulation is the prerequisite for many interaction techniques which consists of several subtasks such as selection, positioning, rotation and scaling (Bowman et al. 2004). As one of the most basic manipulation tasks, selection requires users to accomplish “target acquisition task” (Bowman et al. 2004). Without selection techniques, virtual contents are impossible to be interacted with in the first place.

Ray-casting is the most commonly used selection technique in virtual reality due to its easiness of implementation and usage. With ray-casting, a ray is casted by a hand controller into the environment, and the object hit by the ray is selected when user presses a button. It is easy to learn as well as easy to perform. However, its performance is poisoned by hand jitter especially selecting small or remote targets (Poupyrev et al., 1998; Bowman et al. 2004, Kopper et al. 2011). Researchers have presented many possible solutions and new techniques in order to solve this problem of ray-casting (Haan et al., 2005; Frees et al., 2007; Vanacken et al., 2007; Kopper et al., 2011).

Kopper et al. (2011) proposed the concept of selection with progressive refinement and designed the sphere-casting refined by quad-menu (SQUAD) technique based on this concept. Contrary to direct selection, performing selection with the SQUAD technique takes multiple steps with each step not requiring high precision

while all steps combined can obtain decent results. By SQUAD, user first selects a bunch of objects in the environment with a selection bubble. The objects selection by the bubble is further evenly and randomly distributed on an out-of-context quad-menu. Then user needs to select the quadrant part that contains the target and press a button. After several similar refinement steps on the quad-menu, user can capture the target and be transferred back to the main scene. Allowing users to complete selection with “lazy” steps, SQUAD can achieve almost zero error rate and similar speed compared with ray-casting.

With the appearance of built-in eye tracking headsets in virtual reality, taking advantages of eye tracking could potentially save time to perform tasks and provide completely hands-free experience for users. Besides, high precision required tasks can be solved by eye movements without urging eye tracking devices if progressive refinement is also involved.

In this study, with combination of eye tracking and progressive refinement, we present a novel selection technique - EyeSQUAD, stands for eye-controlled sphere-casting refined by quad-menu. With EyeSQUAD, users first roughly select a bunch of objects containing the target with a selection sphere whose center is determined by eye movements. We used an approximate method to stabilize the point-of-regard that calculated from the eye ray data. After that, similar to the SQUAD technique, the initial objects selected by the selection sphere are evenly and randomly distributed on a quad-

menu which consists of four quadrants. Then users need to fixate on the quadrant that contains the target and select. After several refinement steps, users can obtain the target.

We performed a user study for examining the performance of this new technique with comparison of ray-casting and SQUAD techniques under different target size and distractor density scenarios. Results indicated that SQUAD is significantly precise than EyeSQUAD and EyeSQUAD is significantly precise than ray-casting in general. In view of selection speed, there is no significant difference among these techniques on overall except SQUAD is significantly faster than EyeSQUAD. To be specific, coinciding with previous study (Kopper et al., 2011), target size has significant effects on the performance of ray-casting; however, it is not significant for either SQUAD and EyeSQUAD techniques. Only when the size of target is large, the ray-casting technique is significantly faster than the other two techniques. With consideration of distractor density, as expected, the selection time of SQUAD and EyeSQUAD techniques (i.e. techniques with progressive refinement) significantly increases with varying the distractor density from sparse to dense. Contrary to progressive refinement techniques, the distractor density has no main effect on the selection speed of the ray-casting technique.

According to the results and feedbacks from the user study, we found several design insights that are necessary to be paid attention to for interaction techniques with eye movements. Although controlling by eye movements can potentially provide users

with more intuitive and faster approaches to interact with virtual contents and liberate hands, implementing eye movements as both input and output from a closed-loop design perspective could bring several problems. For instance, eyes can be easily stimulated by sudden changes in frames and bright objects in virtual environment which can act as a negative feedback to poison its performance of interaction techniques that controlled by eye movements. These problems are negligible in traditional distal pointing techniques since the aimed point determined by the direction of a controller will not be influenced by visual stimulus even eyes close for few seconds. Besides, determining commands with eye movements can be catastrophic since unneglectable amount of eye movements are involuntary, which is so-called Midas touch problem.

All in all, in this study, we provided a new selection technique (EyeSQUAD) with the eye tracking metaphor which is rarely implemented for manipulation tasks in virtual reality. With selection by progressive refinement, EyeSQUAD can obtain better accuracy and precision than standard ray-casting technique when target is small. Additionally, transferring some workload from hands to eyes, we surprisingly found that similar performance can be achieved via controlling by eye movements as controlling with hands even with relatively low-cost eye tracking devices. This infers that implementing eye tracking into human-computer interaction techniques is possible to achieve similar performance as usual hand controls which could potentially extends the applicability and the population of the human-computer interactions.

We will explore the possibilities of implementing interaction techniques with eye tracking and dive into these tradeoffs in the following sections. In the next section, we will discuss several selection techniques, show implementations of eye tracking in human-computer interaction and review eye tracking techniques in virtual reality. In the third section, we will present a new eye tracking selection technique. After that, we will show our experiment design, apparatus, and details of a user study that we performed to evaluate the performance of our new technique. Subsequently, in the “Results” section, we will discuss the results from both mean error rate and average selection time aspects. Finally, we draw conclusions of the study and discuss future works in the last section.

## **2. Related Works**

### ***2.1 Previous selection techniques***

Many scholars addressed solutions and presented new techniques that could solve hand jitter problem of ray-casting.

Frees et al. (2007) designed the Precise and Rapid Interaction through Scaled Manipulation (PRISM) interaction technique that can improve the accuracy and precision of standard ray-casting by changing the control/display ratio with respect to hand speed. With implementation of PRISM, user can obtain a different mapping of movement in virtual world which is slower than the actual hand movement in the real world when the hand speed is under certain scaling constant while recover the mapping when hand moves fast. By this design, depending on user's hand speed, a two-way control (PRISM enhanced ray-casting and direct ray-casting) can be utilized to achieve both high precision when speed is not essential and quick speed when accuracy is not interested.

Vanacken et al. (2007) presented the depth ray and 3D bubble cursor techniques that can select even invisible target occluded by other objects. The 3D bubble cursor technique will cast a 3D bubble cursor into the environment which can resize its size to catch one closest target at one time. Also, once one target is caught by the bubble, the target becomes semi-transparent leaves opportunity to select objects behind. However, 3D bubble cursor can be difficult and tiring to use within a highly cluttered environment

which requires high accuracy to catch the exact target among many close distractors. Different from the 3D bubble cursor, basing on the ray-casting technique, the depth ray technique contains a depth marker controlled by movement of hand which can select multiple targets in the meanwhile. One possible limitation of this depth ray technique could be caused by the control of depth marker with same mapping of hand movement. Because this technique would suffer if the actual target is far behind the objects that occludes it whose distance may not be possible to reach with hand movement. However, the transparent function that turns captured targets into semi-transparent in their study is important for implementing techniques that can select fully-occluded target.

Haan et al. (2005) provided a revised version of ray-casting – IntenSelect technique combined with a dynamic rating system for all objects in the environment which can dynamically generate a bended ray between the highest score target and the controller. They performed user study for both static and dynamic to evaluate the performance of this intenSelect technique which showed the InternSelect outperforms ray-casting. Their dynamic rating system for assisting selection could not only be augmented in ray-casting but also other techniques lack of accuracy under specific scenarios with the price of computational power with respect to the number of objects.

Kopper et al. (2011) introduced a new concept of selection by progressive refinement. The idea of progressive refinement is to first select a set of objects that contains the target and then refines them until finally captures the target without

requiring user to be precise in each step while all steps combined together can achieve a decent performance. Based on this concept of progressive refinement. They designed the sphere-casting refined by quad-menu (SQUAD) selection technique. The SQUAD technique outperforms ray-casting when target is small and is precise and easy to use. The presence of SQUAD shows another metaphor to perform tasks that requires high precision and accuracy which is the progressive refinement concept.

## ***2.2 Eye tracking techniques in human-computer interaction***

Eye tracking applications can be divided into two categories, diagnostic and interactive, depending on whether eye tracking is regarded as an input (Duchowski, 2007). Most early-age eye tracking applications serve as diagnostic purposes since data could only be recorded and analyzed retrospectively due to lack of computational capability (Duchowski, 2007). On the contrary, interactive eye tracking techniques did not begin to “flourish” until the minicomputers became widespread when robust real-time computational powers were accessible (Jacob and Karn, 2003). Interactive eye tracking techniques can be further categorized into selective and gaze-contingent subtypes (Duchowski, 2007). Selective type of eye tracking techniques, as the main focus in this study, replace usual inputs such as mouse and controller with the point-of-regard. Gaze-contingent type of eye tracking applications utilize the gaze data to improve the quality of display rendering.

Eye tracking technology is widely used in a broad variety of disciplines such as neuroscience, psychology, marketing/advertising and human factors (Duchowski, 2007), providing objective diagnostic evidence. Eye Movements can also be utilized in usability research. Jacob and Karn (2003) described eye movements in usability research as “Rising from the Ashes” rather than “Taking Off Like Wildfire”. They noted that eye tracking had been impeded from being widely implemented due to technical problems with eye tracking in usability studies, labor-intensive data extraction and difficulties in data interpretation.

Directly implementing eye tracking as a human-computer interaction metaphor would suffer because even current technology could not ensure a robust performance of both tracking and calculation of correct point-of-regard without constantly calibration. Progressive refinement could take a step back from pushing eye tracking always maintains a reliable and precise performance. Combining progressive refinement idea with eye tracking could be a good compromise to achieve high performance without enforcing eye tracking consistently.

Jacob and Karn (2003) introduced one of main issues with control by eye movements as “Midas Touch” problem which revealed the conflict in determining whether the user is “purely” looking or wants to activate certain input. This problem is caused by the closed-loop design when using eyes as both input and output nodes.

Combining eye tracking with speech control could potentially solve this “Midas Touch” problem (Kaur et al., 2003).

“Once eye tracking achieves these improvements in technology (increase validity and reliability), methodology (decrease the intrusiveness of equipment), and cost (cheaper), it can take its place as part of a standard HCI toolkit.” (Poole and Ball, 2006)

Selecting with eye movements is more intuitive and convenient for at least liberating one hand. Several researchers (Jacob and Karn, 2003; Poole and Ball, 2006; Duchowski, 2007) mentioned that using eye tracking as control signals can save time and replace traditional input devices such as mouse, keyboard and controller.

### ***2.3 Eye tracking techniques in virtual reality***

Interacting with eye movements in virtual reality has great potential. Poole and Ball (2006) mentioned that controlling by eye movement can be implemented in virtual reality. VR usually have far-away objects spread out in a large three-dimensional space. Compared with other input devices, control by eye movement could easily move from an object to another one.

Tanriverdi and Jacob (2000) designed an eye tracking navigation technique and evaluated the effect of this technique on spatial memory compared with a hand-pointing technique. They found that the technique with eye tracking could be faster than the hand-pointing technique especially in distant virtual environments although the hand-pointing technique would yield better spatial memory impacts. Their study proved that

interaction with eye movements can be feasible even with relatively low accurate and reliable eye tracking hardware. However, if they had included other inputs for commands rather than eye movements alone, their eye tracking technique would have avoided the Midas Touch problem.

Pfeiffer (2008) tested the accuracy of an eye tracking device in a CAVE-type VR system. They achieved an accuracy of about 1° horizontally and 2° vertically which might be accurate enough at that time while these accuracies may not be enough in some studies (Kopper et al., 2011) where the largest target size is around 1°. Also, the Midas touch problem can interrupt the use of the technique if controlling completely with eye movements.

Although there are few eye tracking interaction techniques in virtual reality, none of them can achieve high accuracy in a cluttered environment and well address the Midas touch problem.

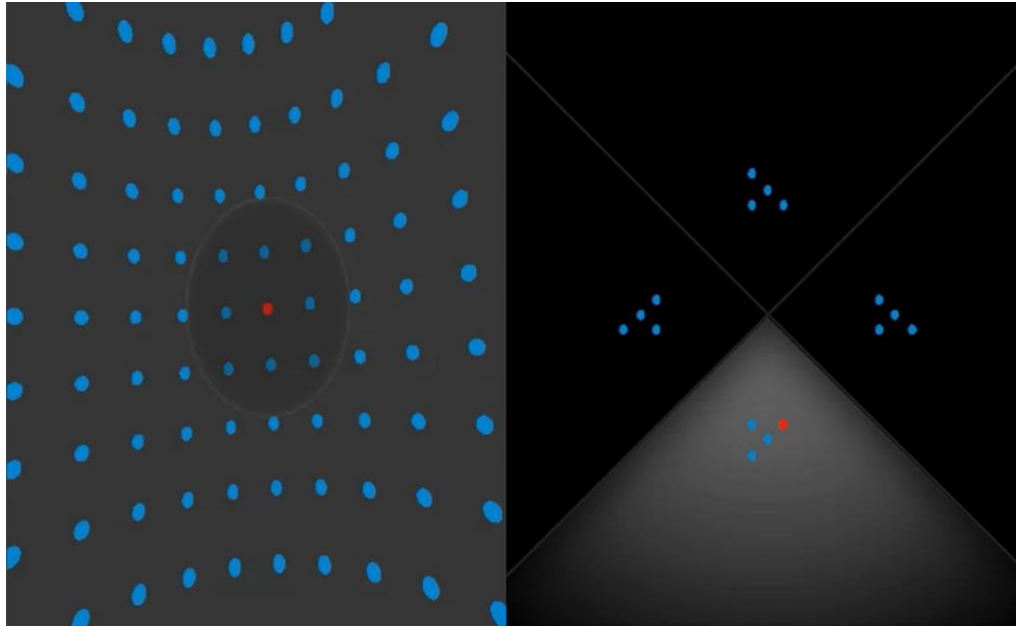
### 3. EyeSQUAD Selection

We designed a novel selection technique with eye tracking - eye-controlled sphere-casting refined by quad-menu (EyeSQUAD) selection technique which combines the progressive refinement idea from a previous selection technique - SQUAD (Kopper et al. 2011). Progressive refinement is an indirect method of selection which allows users to first select a bunch of objects including the target and then refine them steps by steps until finally obtain the target. Performing progressive refinement only requires “lazy” selection in each step while all steps combined together could obtain a decent performance.

We can basically divide the EyeSQUAD technique into two subtasks: sphere-casting subtask and quad-menu refinement subtask.

For the sphere-casting subtask, instead of casting a sphere by a controller, EyeSQUAD allows user to control the selection sphere with eyes by calculating the convergence point from the user’ eye rays data. We set the diameter of the selection sphere to be  $26.3^\circ$  which is consistent with the angular size of the selection bubble in SQUAD. The size of selection sphere is visually constant to prevent the sphere from being invisible or oversize when it is far away or close enough. The objects inside of the selection sphere will be chosen as the initial set of objects that need to be further refined when user performs selection (see Figure 1 left).

Then this set of objects are evenly and randomly distributed on a quad-menu with out-of-content display (Figure 1 right). After that users need to fixate on the quadrant part that contains the target that they want and perform further selection. The objects on that quadrant will be distributed again on the quad-menu. Progressive refinement continues until the user gets the target or exit in the middle if lost the target. Once the quad-menu selection process is finished, the user will be transferred back to the original scene. One limitation of this out-of-context selection design is difficult to select two similar objects in different depths since the depth information of all selected objects is ignored once arranged on the quad-menu. Another potential limitation is that the details for differentiating different objects may be lost after resized by the quad-menu. Although spheres are used as objects in the experiment for visual consistency by any viewpoints, these two limitations should be noticed when implementing EyeSQUAD in realistic situations.



**Figure 1: Main scene (left) and QUAD-menu selection scene (right)**

We found that the calculated point-of-regard was precise and stable when mapped in a 2D plane which was orthogonal to the user (e.g. the quad-menu) while was greatly contaminated once implemented in 3D with the depth involved. Mismatches between the position of the convergence point from the recorded eye rays and the physical position of point-of-regard would appear after ran eye tracking for a while or moved the device a little bit. One possible method to optimize the performance of the technique in 3D could be separating the depth control from the technique and determining the depth by other user data such as pupil size and extent of squinting. By this way, the point-of-regard can be casted onto an invisible plane which is orthogonal to the user and is changed with the extent of squinting or pupil size. Unfortunately, however, by the end of this study, we have not found a way to access to users' data of

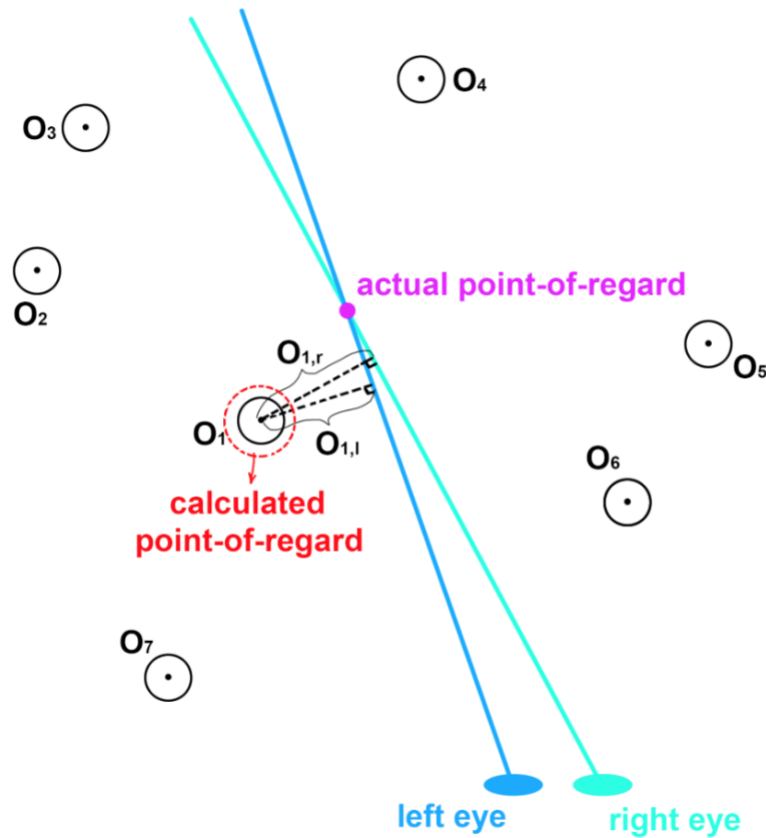
pupil size or extent of squinting. Nevertheless, we mentioned this approach since it can be possible once the data of pupil size or extent of squinting become available for future improvements.

We found another way to stabilize the calculated point-of-regard which is also called “convergence point” in this paper. Instead of directly controlling the selection sphere by the calculated convergence point, with a list of target positions, the selection sphere will always move to the closest target in the environment which is determined by calculating the minimum summation of the distance from the target to the two eye rays (see the closest target approximation subsection). This requires knowing target positions and much computational power when the number of target is large. However, the positions of targets in most interaction tasks are reasonably accessible. Besides, under relatively numerous targets scenario, the whole space can be further divided into several parts depending on the number of potential targets in the environments and only the target positions in the part where the convergence point locates will be taken into consideration (see the space partition subsection). By this way, the concern of computational cost can be solved as well. This approximation of controlling the selection sphere could ensure the user can catch at least one potential target at a time and would be efficient in an aggregated objects environment.

We separated the select commands from the whole selection process by pressing a button on a controller in case of a complete closed-loop design and the Midas touch

problem. In the future implementation, we plan to use a “tongue click” sound as a sound input to control the selection in order to support a completely hands-free experience.

### 3.1 Closest target approximation



**Figure 2: Schematic of the closest target approximation method**

As shown in Figure 2, we suppose the positions of two eyes in the environment are  $M_l\{x_l, y_l, z_l\}$  and  $M_r\{x_r, y_r, z_r\}$ , respectively while the directions of two eyes are  $S_l\{\alpha_l, \beta_l, \gamma_l\}$  and  $S_r\{\alpha_r, \beta_r, \gamma_r\}$ , respectively (“l” represents the left eye, “r” indicates the

right eye). Besides, the position of a certain object  $i$  in the environment is written as

$M_{O_i}\{x_i, y_i, z_i\}$ .

$$\overline{M_{O_i}M_l} = \{x_i - x_l, y_i - y_l, z_i - z_l\} \quad (1)$$

$$\overline{M_{O_i}M_l} \times \overline{S_l} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ x_i - x_l & y_i - y_l & z_i - z_l \\ \alpha_l & \beta_l & \gamma_l \end{vmatrix} \quad (2)$$

Then we can calculate the distance from the object  $i$  to the left eye ray as follows,

$$d_{O_i,l} = \frac{|\overline{M_{O_i}M_l} \times \overline{S_l}|}{|\overline{S_l}|} \quad (3)$$

Similar for the distance from the object  $i$  to the right eye ray. We can get the sum of the distances from object  $i$  to two eye rays.

$$d_{O_i,sum} = d_{O_i,l} + d_{O_i,r} = \frac{|\overline{M_{O_i}M_l} \times \overline{S_l}|}{|\overline{S_l}|} + \frac{|\overline{M_{O_i}M_r} \times \overline{S_r}|}{|\overline{S_r}|} \quad (4)$$

Finally, we can find the closest target by

$$index = \underset{i \in \{1, \dots, n\}}{\operatorname{argmin}} d_{O_i,sum} \quad (5)$$

The calculated point-of-regard is the position of the closest object

$M_{O_{index}}\{x_{index}, y_{index}, z_{index}\}$ . The selection will always move to this calculated point-of-regard with a constant speed of 6 m/s. Since the movements of eyes are saccade (Deubel and Schneider, 1996), we decided to transfer the bubble with continuous motion rather than instant transform to the temporal calculated point-of-regard in case of catastrophic simulator sickness.

### 3.2 Space partition

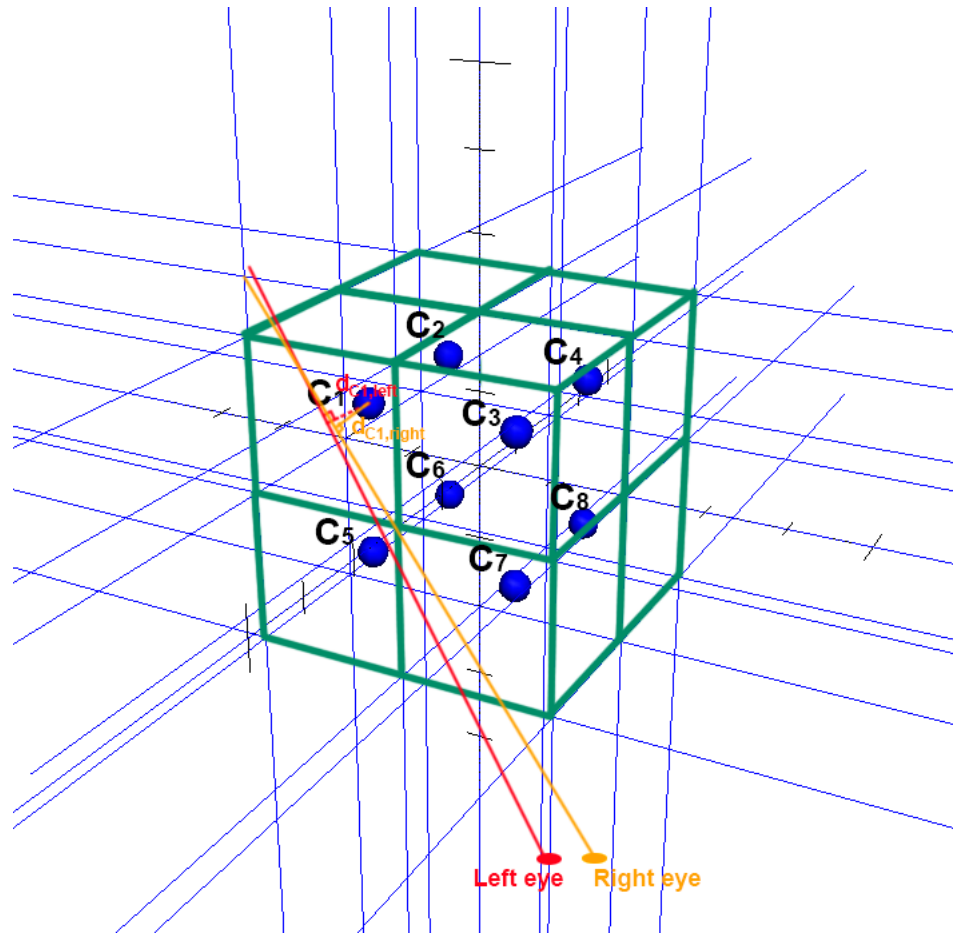


Figure 3: Schematic of the space partition method

Algorithm:

SpacePartition(Space C)

1. Partition space C into several parts
2. Find the closest center  $C^*$  in all partitioned parts that contain objects with the Closest Target Method.
3. If only one object left in  $C^*$ , return the position of this object;

else, return SpacePartition( $C^*$ ).

Space Complexity:  $O(N)$ , where  $N$  is the total number of objects in the space  $C$ .

Time Complexity: With the assumption of evenly distributed objects in the space, the expected worst-case running time is  $O(N \log N)$ .

The closest target approximation method may appear awkward and computationally expensive especially within a cluttered environment. In order to save computational power and optimize the approximation algorithm, the space partition (Figure 3) can be included which first partitions the whole space into several parts depending on the magnitude of target positions in the environment and then find the closest center of a partitioned part that contains targets (e.g.  $C_1$  in the Figure 3). Empty partitioned parts are just ignored. After that, only applying the closest target method in that closest part (e.g.  $C_1$  part in the Figure 3) rather than directly calculating all the distances from possible target positions to eye rays. If there is only one object left in the closest partitioned part (where closest center locates), the method returns the position of this object which will be regarded as the current calculated point-of-regard. Otherwise, the method will recursively call this space partition algorithm using the closest partition part as the input. By this recursion algorithm, the whole space can be partitioned into smaller ones until finally catches the closest target. A lot of computational cost can be

saved through recursion especially when the number of initial selected objects is large instead of brute force.

## **4. Methods**

### **4.1 Experiment design**

We evaluated the performance of the EyeSQUAD technique with comparison of ray-casting technique and SQUAD technique based on a selection task – acquire a target surrounded by several distractors in a virtual environment. The size of these objects and density of the distractors vary among different conditions in the experiment.

#### **4.1.1 Goals and Hypotheses**

The purpose of the experiment is to determine the tradeoffs between the EyeSQUAD technique and two other previous selection techniques – ray-casting and SQUAD. Ray-casting only requires a single but precise click during the selection while SQUAD and EyeSQUAD allow user to select with little precision in each step but requires several steps. EyeSQUAD is able to enhance the accuracy and speed of the SQUAD technique since selection with eyes is more intuitive and saves time. Besides, EyeSQUAD can avoid several selection issues introduced by hand controller such as hand jitter and occupation of at least one hand during selection. This study aims to answer two research questions:

- 1) Can the EyeSQUAD selection technique outperform previous selection techniques such as the ray-casting technique and the SQUAD technique?
- 2) Will target size or distractor density have influence on the performance of selection techniques?

With the consideration of the tradeoffs and the research questions, we hypothesized that

(H1) The time of selecting a target with SQUAD or EyeSQUAD will not be affected by the target size while ray-casting will be slow with small targets and fast with large targets.

(H2) The time of selecting a target with SQUAD or EyeSQUAD will be proportional to the number of distractors in the virtual environment while the performance of ray-casting will not be influenced by the distractor density.

(H3) The EyeSQUAD will outperform ray-casting when number of distractors is small.

(H4) The EyeSQUAD will outperform SQUAD in all conditions.

(H5) The SQUAD and EyeSQUAD techniques will have virtually no errors due to their low requirement of precision while ray-casting will increase errors with decreasing the target size.

#### **4.1.2 Design**

Since individual difference with eye tracking methods is significant (Goldberg and Wichansky, 2003). We used a factorial within-subject design with repeated measures. There are three independent variables: technique (ray-casting, SQUAD, EyeSQUAD), target size (small: radius 0.01m or  $0.26^\circ$ , medium: radius 0.015m or  $0.40^\circ$ , large: radius 0.04m or  $1.06^\circ$ ), and the distractor density (sparse: 16, medium: 64, dense:

256). This design is 3x3x3. There are two dependent variables: time to complete a task, mean number of errors per trial.

The order of the presentation of technique was counterbalanced while each of the nine conditions of target size vs. distractor density was repeated 8 times and presented in random order.

## **4.2 Apparatus**

We used the FOVE Eye tracking headset (weight: 520 g) which is the first head-mounted display headset in virtual reality has built-in eye tracking functionality. Since our research requires subjects to stand at a fixed point within a room tracking space during the experiment. Besides, a controller tracking is necessary for two experiment conditions (i.e. ray-casting and SQUAD). With the consideration of tracking consistency and reasons above, we used HTC Vive room space tracking for both an HTC Vive controller and headset tracking by muting original head tracking of the FOVE and mounting an HTC Vive tracker (weight: 300 g) at the FOVE headset (see Figure 4). To achieve that, one laptop (PC1) connected with the FOVE headset provides the display of virtual contents and eye tracking while one desktop (PC2) connected to the HTC headset provides headset and controller tracking.

FOVE Unity plugin v1.3.0 was used driven by FOVE Version 0.13.0 on an ASUS GL502V (PC1) Quad-Core Processor (2.8 GHz), 16.0 GB RAM, with NVIDIA GeForce GTX 1070 running Windows 10. An Alienware X51 R3 Edition (PC2), with Quad Core

Processor (2.7 GHz), 8GB RAM, NVIDIA GeForce GTX 970 running Windows 10, drives HTC SteamVR plugin v1.2.3 to support a HTC Vive with 6DOF position and orientation tracking. A local server was built for supporting real-time data transfer between the two PCs (PC1 and PC2) through UDP.



**Figure 4: Experiment participant and FOVE headset (with an HTC Vive tracker) and HTC Vive controller.**

The virtual environment was made with Unity 3D Engine (version 2017.1) and the scripts were written with C#. All virtual objects including a target and other distractors were circular and located in a sphere with a radius of 2.155m whose center was the position of the participant. The target was chosen within an inner sphere (radius 1.1m) which ensured that the target located within the original visual range in each selection and the selection bubble would at least cover certain number of objects (e.g. 16, 64, 256) in one selection. This fixed the refinement steps (e.g. 2, 3, 4) under each target density condition.

Before each selection task, a reset session was included to ensure that participants began from a same position since the difficulty of a pointing task is positive correlated with the amplitude of movement to accomplish the task (Fitts, 1954). Participants could take a small break in the reset session if they needed. During the reset session, they used the controller or eyes to move a dot into a large circle in the center of the screen and press the touchpad on the controller to proceed. Timing would only start after the reset session.

### **4.3 Participants**

24 voluntary unpaid participants (12 male, 12 female) were recruited for the experiment whose age from 21 to 32 years old with a median age of 24 years old. All of the participants are graduate students except one post-doc scholar.

### **4.4 Procedure**

Participants were first welcomed by the experimenter and given background information of the study. Then participants needed to read and sign informed consent form which includes experiment details such as procedure, benefits and risks. After that, they were asked to complete color blindness assessment and background survey online. Since none of them were identified as people with color blindness, none of participants were excluded from the experiment.

Participants were emphasized to perform the trials as quickly as possible while making as few mistakes as they can and making fewer mistakes is more important than

being quick. Then the experimenter would explain how to complete the selection task. They were notified that they should hold the controller with dominant hand and could not use the other hand to steady the controller through all trials. Once the experimenter finished explanation of usage of the Vive controller, they were asked to move to the experiment area which is a fixed point in the room tracking space and wore the FOVE headset. There was also a red starting point in the virtual environment which was consistent with the starting point in the real world and would change to green if the participant was close enough to it (smaller than 0.1m).

After that they would start learning their first technique in a corresponding training session which would teach them how to use the technique and allow them to try all nine combinations of target size and target density conditions once. During the training session, they would be told to accomplish at least one correct selection and one error selection to see both results. (For correct selection, a checkmark would be displayed. Otherwise a cross would appear.) After the training session, they would perform the corresponding experiment condition of the technique which contains 72 trials with all 9 combinations repeated 8 times in random order.

Once completed all of these trials, they filled a technique rating questionnaire for the technique they just performed. All participants accomplished all three techniques in a specific order that was counterbalanced. After finishing all of these conditions, they filled a post-study overall performance questionnaire.

## **4.5 Pilot Study**

We first performed a pilot study for 12 participants (9 males, 3 females), age between 21 to 28, all right-handed users.

During the pilot study, many participants mentioned that they had a hard time in the quad-menu selection process. Even when they were fixating on a point on the quad-menu, the highlighted screen quickly changed between two quadrants back and forth which could bring another distraction in the experiment.

This was mainly because they were looking at the target (red dot) which could locate at the margins between two quadrant parts. The worst case would be the target located near the center of the quad-menu which resulted in highlighted quadrant shaking among all four parts. The best way to choose among these quadrants is to look at the button part of each quadrant which is far from the margin and can easily identify which quadrant was looked at instead of focusing on the target.

Some stated that their eyes were stimulated by sudden appearance of large number of light distractors which would influence the use of the technique.

According to these feedbacks, we rearranged the distribution of all objects on the quad-menu to a place further away from margins and set the color of highlighted quadrant darker in case of stimulating users' eyes too much. The redistribution on quad-menu could narrow down the maximum number of objects that can be placed on the quad-menu compared with the arrangement of objects before the redistribution. This

could potentially be a limitation of the redistribution. However, with current arrangement of objects in the experiment, around thousand objects could be displayed on the quad-menu at once which might be the extremely dense case in real world applications. (Note: These thousand objects are the objects selected by the selection bubble rather than the total number of objects in the environment.)

## 5. Results

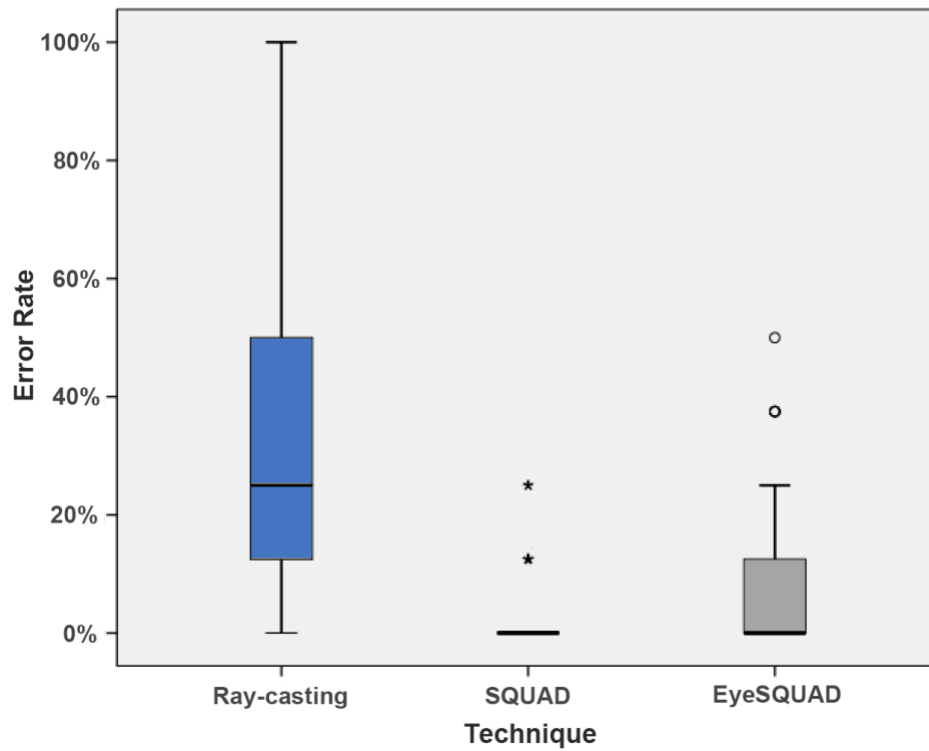
We used a repeated measure multi-variate ANOVA (MANOVA) model with a significant level of 0.05 to evaluate impacts on two dependent variables: 1) mean error rate and 2) average time to complete selection. There are three independent variables: 1) technique (Ray-casting, SQUAD, EyeSQUAD), 2) target size (small, medium, large) and 3) distractor density (sparse, medium, dense). We performed repeated measure since all nine combinations of target size and distractor density are randomized with counterbalanced order of technique and the repeated measure could provide more statistical power with fewer subjects.

Due to the small sample size of 24 subjects compared with 3 independent variables and potential interactions, the power of results will be checked in case of exaggerating the significance of the effect of a factor which might not exist. In the study, the significance is examined by identifying the power of test whether is larger than 0.8 which indicates “sufficient power to detect effects” (Field, 2013). From the results, we find that our sample size is enough since sufficient powers are observed in the tests.

### 5.1 Mean Error Rate

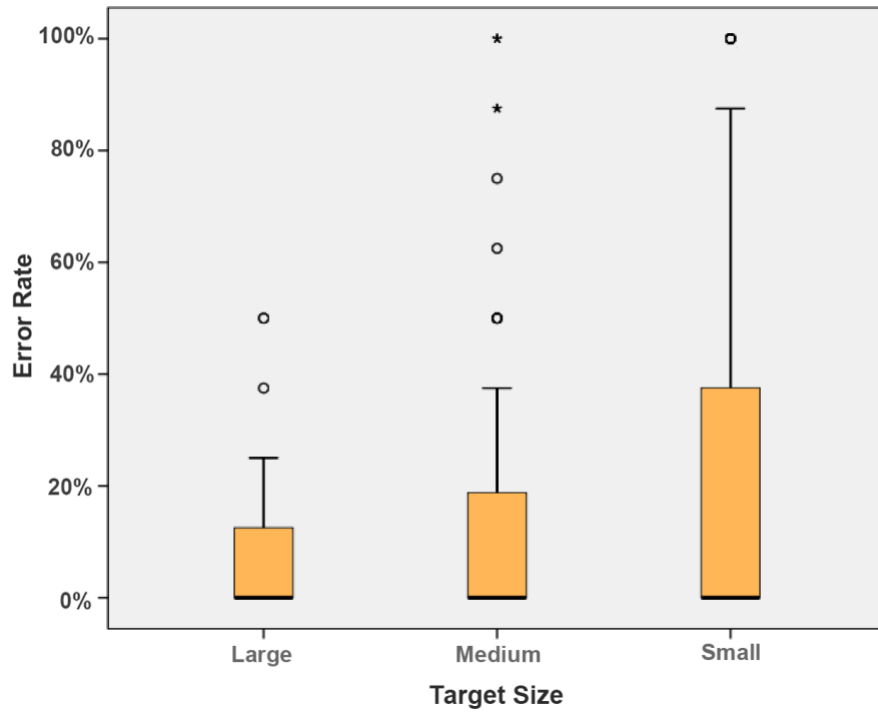
Overall, technique has a significant effect ( $F(0.95; 2,36) = 97.986$ ,  $p < 0.001$ , power  $> 0.999$ ) on the error. Because the technique independent variable has 3 levels: ray-casting, SQUAD and EyeSQUAD, the degree of freedom (DOF) for this independent variable is thus 2. The total number of subjects in the experiment is 24. Therefore, the

total DOF is 46 (i.e.  $(3 - 1) * (24 - 1) = 46$ ). The DOF of error for the technique is 36 obtained by subtracting the DOF of effects (i.e.  $(3 - 1) * (6 - 1) = 10$ , where 6 is the total number of orders) from the total DOF. Since there are three independent comparisons performed, with consideration of the family-wise error rate, the alpha value for each comparison can be calculated through the Bonferroni correction ( $\alpha_{\text{per comparison}} = \frac{\alpha}{m} = \frac{0.05}{3} = 0.0167$ , where m is the number of independent comparisons performed). As shown in Figure 5, ray-casting leads a significantly higher mean error rate of 34.2% compared with the mean error rate of 0.9% of SQUAD ( $p < 0.001$ ) and the mean error rate of 6.2% of EyeSQUAD ( $p < 0.001$ ) with the Bonferroni test. SQUAD has a significantly lower mean error rate than EyeSQUAD ( $p < 0.001$ ). This contradicts the hypothesis (4) and the former part of the hypothesis (5) (i.e. EyeSQUAD virtually shows no errors). In the discussion section, we will discuss several aspects that we have not taken into consideration when designing the experiment.

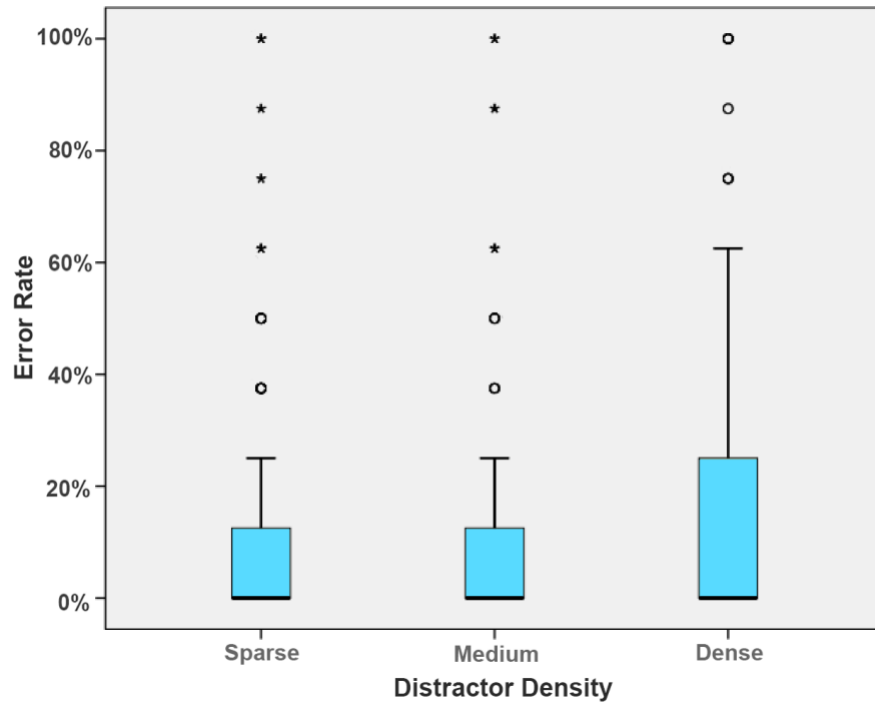


**Figure 5: Mean error rate with different techniques.**

Apart from technique, target size (Figure 6) has a significant effect on the error ( $F(0.95; 2,36) = 36.762, p < 0.001, \text{power} > 0.999$ ) while distractor density (Figure 7) is not significant ( $F(0.95; 2,36) = 0.432, p = 0.654, \text{power} = 0.112$ ). This shows that the effect of target size on ray-casting is so significant that even averaged on overall techniques it is still significant. Performing pairwise comparisons with the Bonferroni test, large target size has significant lower error rate than medium target size ( $p < 0.0167$ ) and medium target size has significant lower error rate than small target size ( $p < 0.001$ ). Contrary to target size, the effect of distractor density is not significant once averaged on overall techniques.



**Figure 6: Mean error rate with different target sizes.**

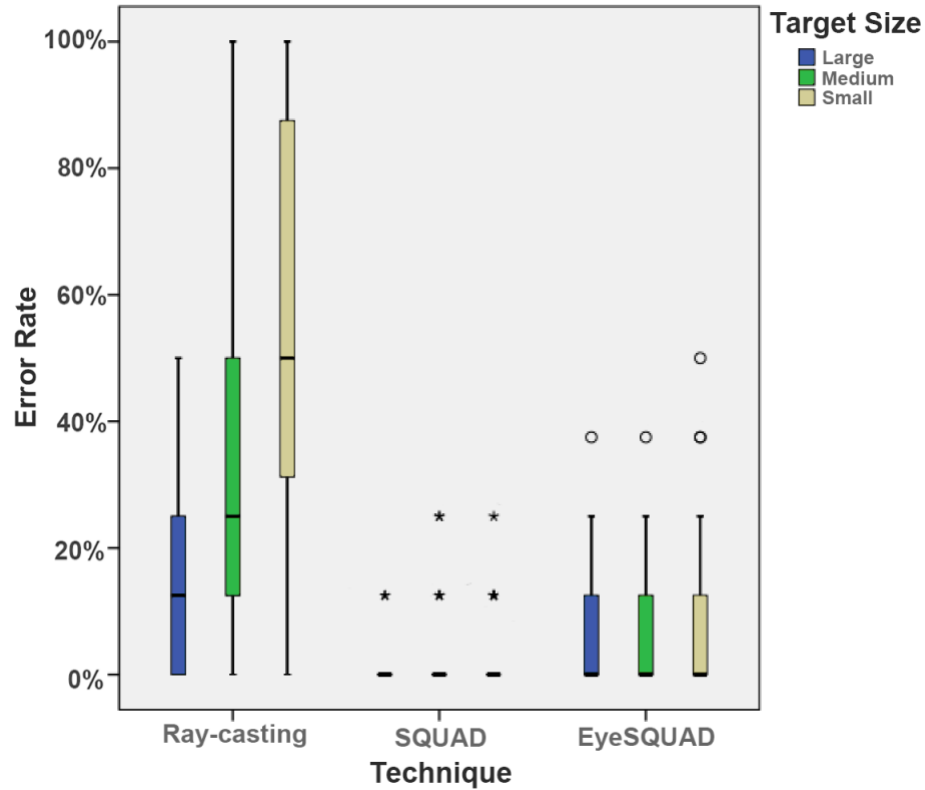


**Figure 7: Mean error rate with different distractor densities.**

Furthermore, the interaction of technique and target size ( $F(0.95; 4,72) = 29.874$ ,  $p < 0.001$ , power  $> 0.999$ ) has a significant effect. We can further explain the result by examining the interaction.

Figure 8 shows the interaction between technique and target size on the mean error rate when fixing the technique. Consistent with the results of the study of Kopper et al. (2011), there is a significant effect of target size with the ray-casting technique. The Bonferroni test indicates that the medium size has significantly higher error rate than the large size ( $p < 0.0167$ ) while the small size has significantly higher error rate than the medium size ( $p < 0.001$ ). As expected, however, the target size has no significant effects

on techniques with progressive refinements, that is, SQUAD ( $p = 0.534$ ) and EyeSQUAD ( $p = 0.817$ ) techniques. This coincides with the latter part of the hypothesis (5), which is that the ray-casting increases errors with decreasing the target size.



**Figure 8: The interaction between technique and target size when fixing technique.**

After examined the contribution of target size on the effect of interaction between technique and target size, the impact of the technique variable can be unveiled by fixing the target size as shown in the Figure 9. When the target size is medium and small, significant differences can be found among these techniques. The SQUAD technique is significantly more precise than the ray-casting technique ( $p < 0.001$ ) and the EyeSQUAD

technique ( $p < 0.0167$ ). EyeSQUAD is significantly more accurate than ray-casting when target is medium and small size ( $p < 0.001$ ). However, EyeSQUAD is less accurate than SQUAD regardless of target size because SQUAD yields lower error rate also when target is large ( $p < 0.0167$ ). This also disproves the hypothesis (4), which reveals that the SQUAD outperforms EyeSQUAD with respect to precision in all target sizes. When target is large, although ray-casting also has higher error rate than the SQUAD ( $p < 0.001$ ), there is no significance found between ray-casting and EyeSQUAD ( $p = 0.143$ ).

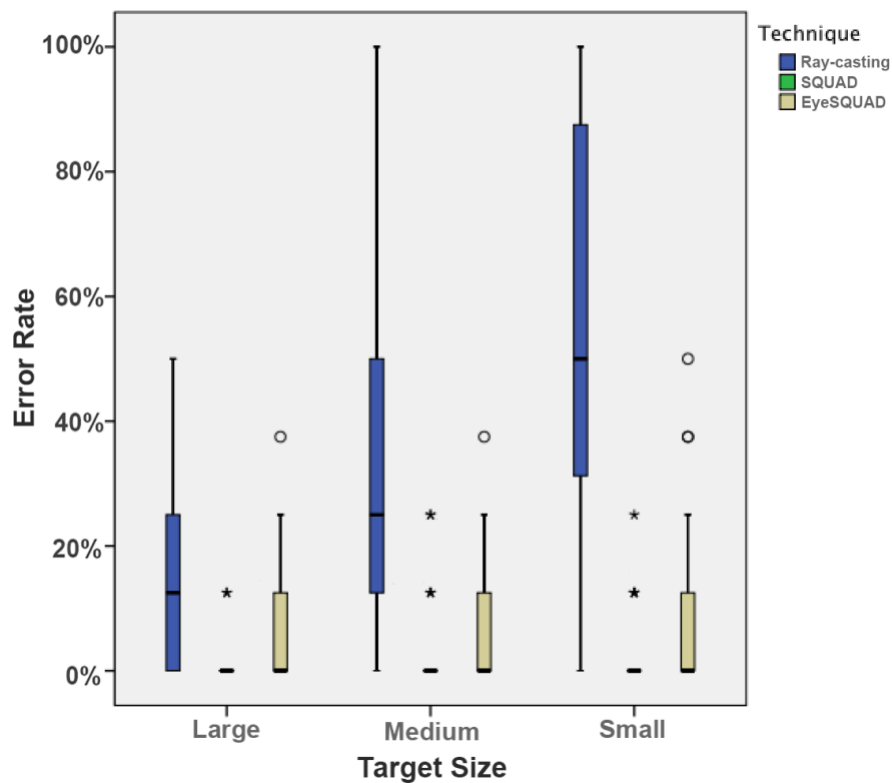
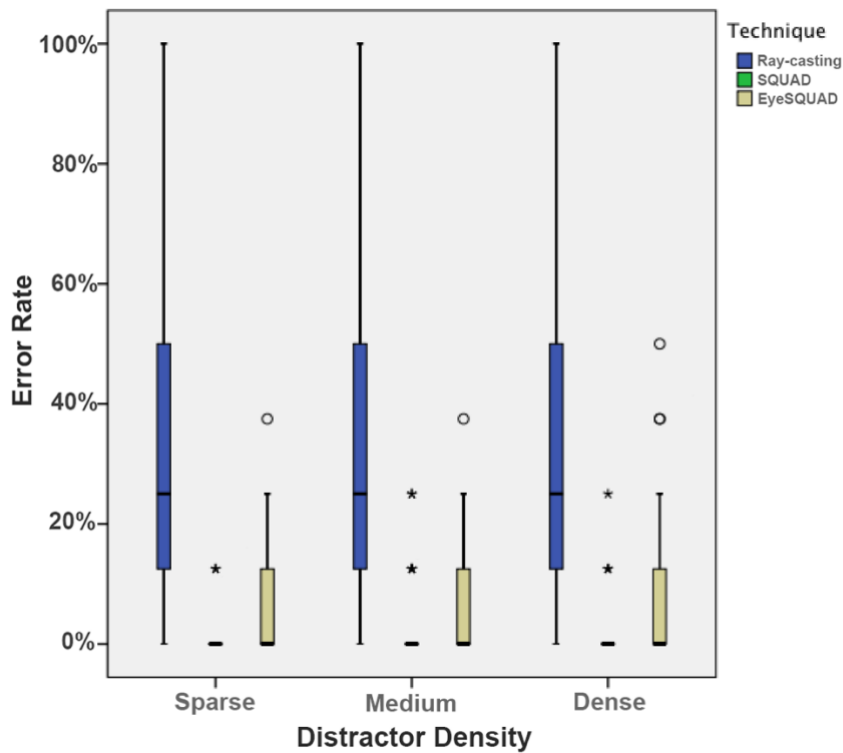


Figure 9: The interaction between technique and target size when fixing target size.

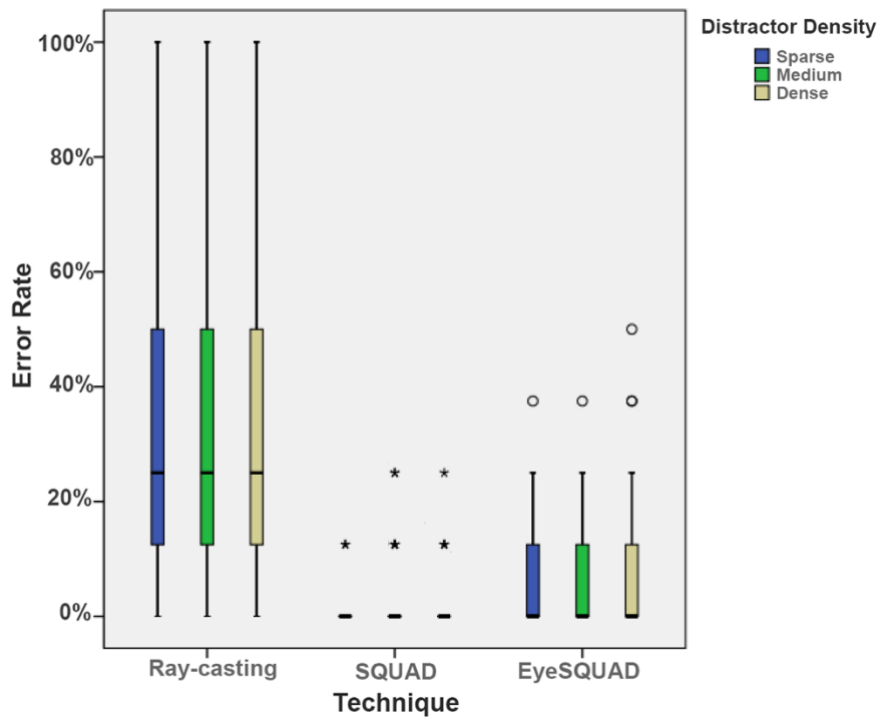
Besides, the effect of the interaction of technique and distractor density ( $F(0.95; 4,72) = 3.745, p < 0.0167, \text{power} = 0.854$ ) is found significant on overall. Examining the interaction of technique and distractor density (Figure 10), when fixing the distractor density, the Bonferroni test shows that ray-casting has significant higher error rate than both SQUAD ( $p < 0.001$ ) and EyeSQUAD ( $p < 0.001$ ), and SQUAD is significantly accurate than EyeSQUAD ( $p < 0.0167$ ) for any distractor density.



**Figure 10: The interaction between technique and distractor density when fixing technique.**

Nevertheless, when fixing the technique (Figure 11), there is no significant effect found with either ray-casting or SQUAD. With EyeSQUAD technique, the sparse

distractor density yields significantly lower error rate than medium distractor density ( $p < 0.0167$ ), although no other significant difference is found. This suggests that errors are more possibly to yield when increasing the number of refinement steps from 2 to 3 during selection with EyeSQUAD. In other words, the quad-menu selection process also has an effect on the errors which contributes potential amount of system errors.



**Figure 11: The interaction between technique and distractor density when fixing distractor density.**

A constant amount of error rate can be observed from the interaction between technique and target size when fixing technique (Figure 8) and the interaction between technique and distractor density when fixing distractor density (Figure 11). This

constant amount of error rate is mainly caused by system aspects such as losing accuracy of eye tracking with time proceeds, or by human aspects such as losing attention, or by design aspects such as distractions on the quad-menu. For instance, the accuracy of EyeSQUAD has been greatly improved from the pilot study to the actual experiment (17.4% to 6.2%) mainly by carefully removing some distractions on the quad-menu (e.g. move the distribution of objects away from the margin of the quadrants on quad-menu to prevent the calculated point-of-regard from shaking around margins).

Finally, under our experimental settings, no significance is found in the interaction of technique, target size and distractor density ( $F(0.95; 8,144) = 0.935$ ,  $p = 0.492$ , power = 0.412). No other significant differences can be detected on the error rate.

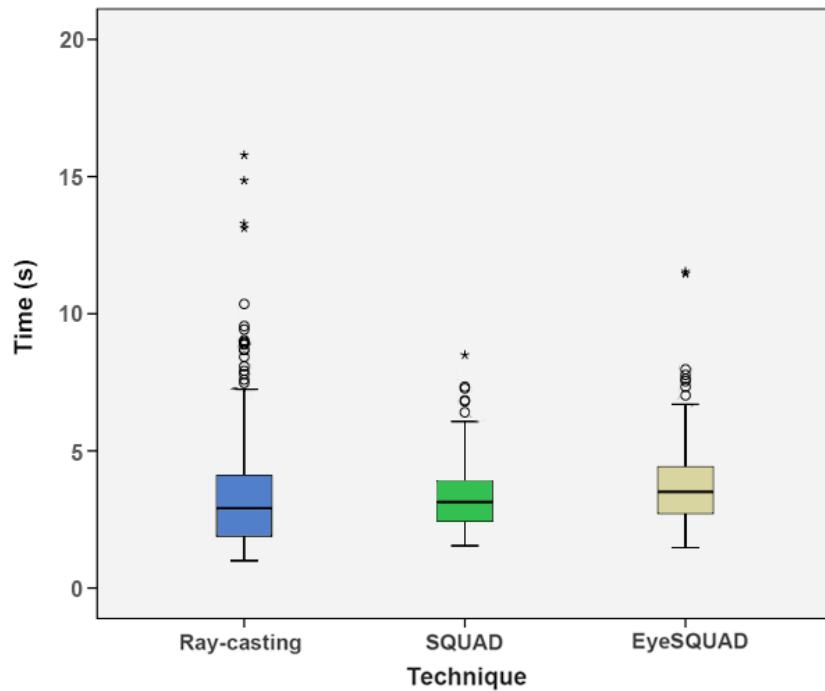
We checked whether there is a significant order effect in the study which attributes to the completely within-subject experiment design. As a between-subjects variable, the order is found not significant in the study on the error ( $F(0.95; 5,18) = 1.915$ ,  $p = 0.165$ , power = 0.451).

## **5.2 Average Selection Time**

Overall, target size ( $F(0.95; 2,36) = 35.768$ ,  $p < 0.001$ , power > 0.999) and distractor density ( $F(0.95; 2,36) = 67.810$ ,  $p < 0.001$ , power > 0.999) have significant effects, compared with no significance of technique ( $F(0.95; 2,36) = 2.404$ ,  $p = 0.112$ , power = 0.437). These suggest the effect of target size on the selection time of ray-casting and the

effect of distractor density on the speed of progressive refinement techniques are so significant which are still significant even averaged on overall.

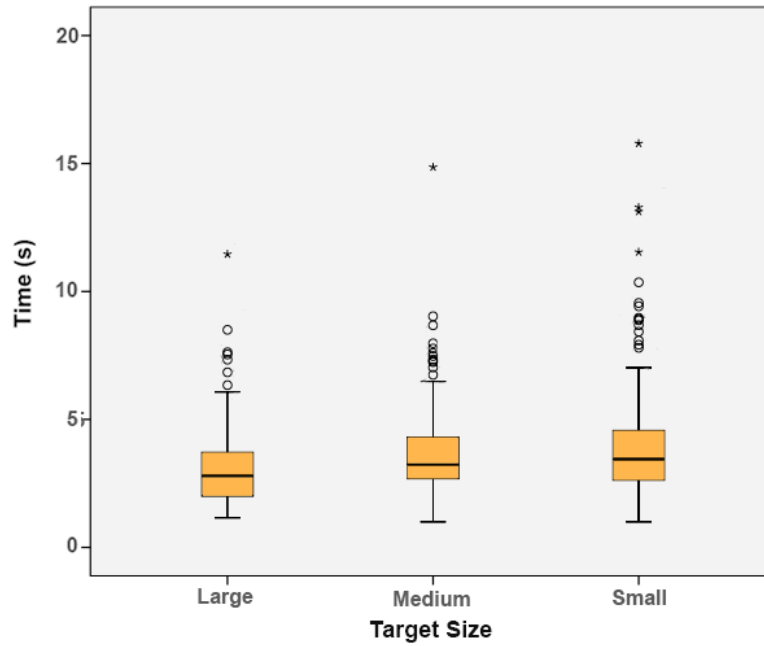
Delving into the pairwise comparisons of techniques first, the Bonferroni test suggests that EyeSQUAD has significantly lower selection speed than SQUAD ( $p < 0.0167$ ) although no significant difference is observed either between ray-casting and SQUAD ( $p = 0.781$ ), or between ray-casting and EyeSQUAD ( $p = 1.000$ ) as shown in Figure 12.



**Figure 12: Average selection time with different techniques.**

For the target size (Figure 13), the small target size has significantly longer selection time than the large target size ( $p < 0.0167$ ) and the medium target size ( $p < 0.001$ ) by checking the pairwise comparisons with the Bonferroni test. Besides, the

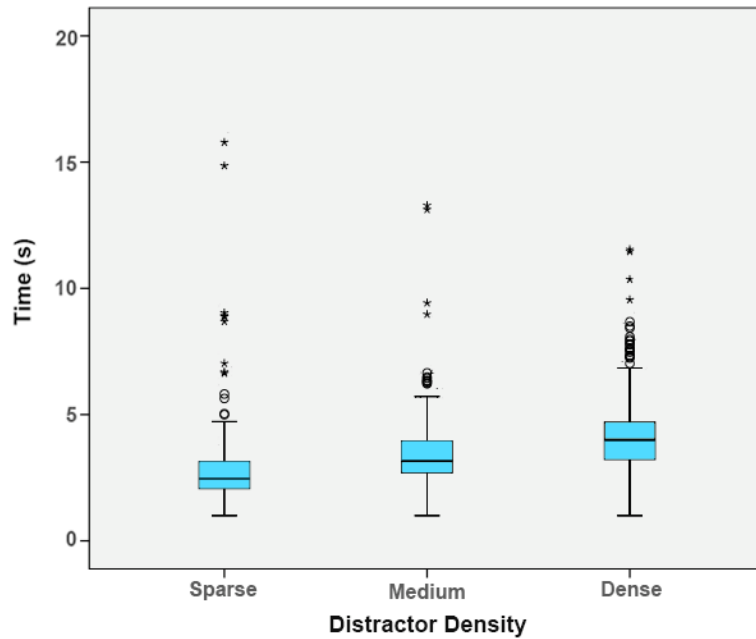
medium target size has significant longer selection time than the large target size ( $p < 0.0167$ ).



**Figure 13: Average selection time with different target sizes.**

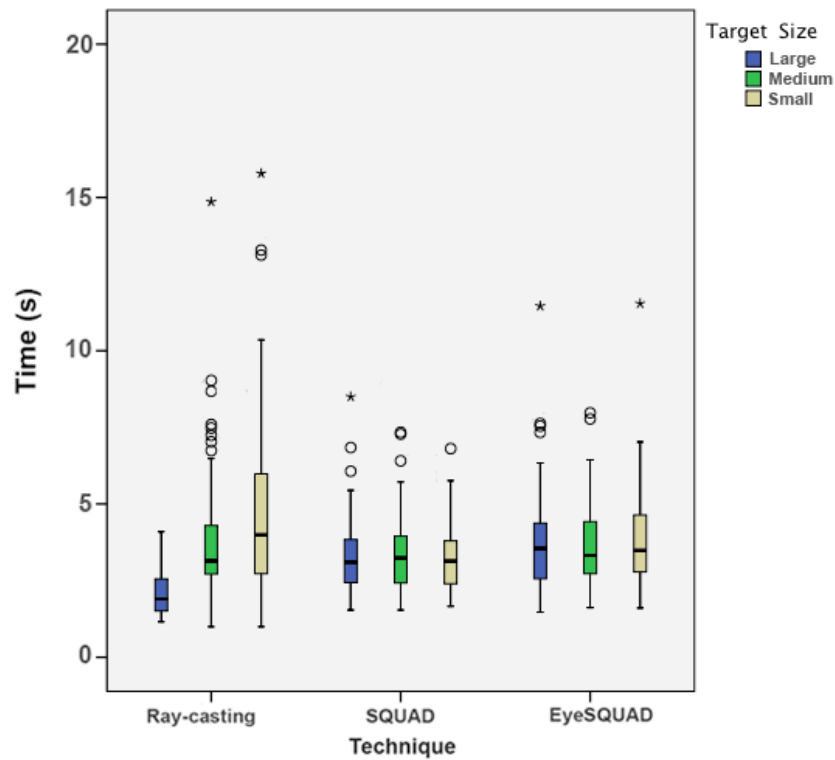
On overall, the distractor density also has significant effect on time (Figure 14).

With increasing the density of distractor in the virtual environment, the selection time is significantly increased. Since sparse density has the lowest selection time compared with medium ( $p < 0.001$ ) and dense ( $p < 0.001$ ) densities with dense density being slower than medium density ( $p < 0.001$ ).



**Figure 14: Average selection time with different distractor densities.**

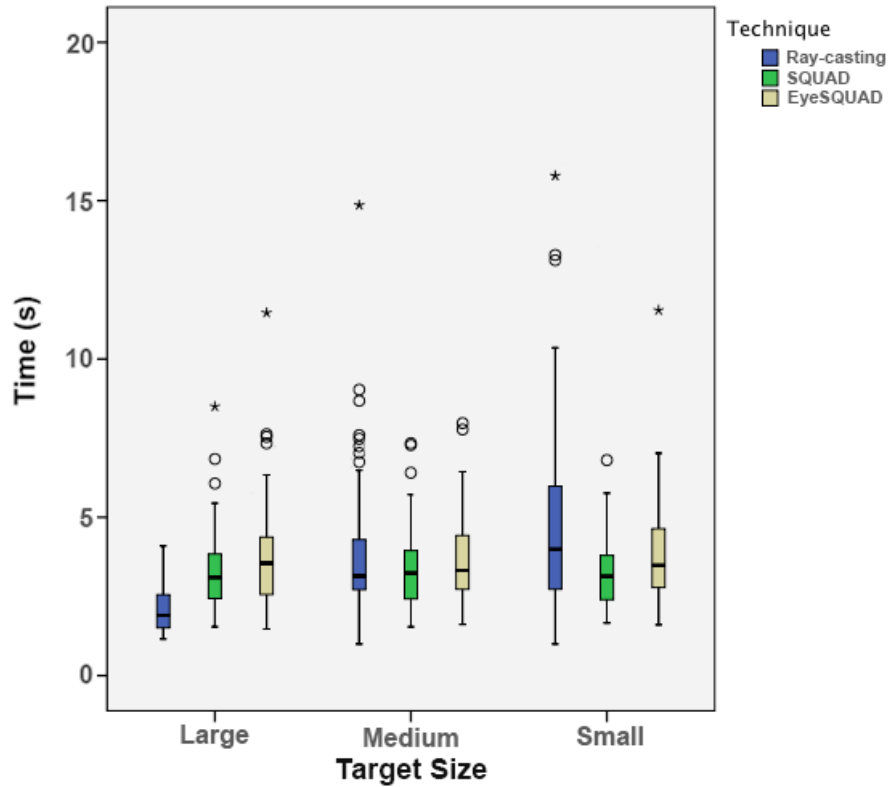
Examining the interaction of technique and target size ( $F(0.95; 4,72) = 34.545$ ,  $p < 0.001$ , power  $> 0.999$ ) yields significant effect. First, the impact of target size on the interaction can be revealed when fixing technique (Figure 15). With the ray-casting technique, the large target size is significantly faster than medium target size ( $p < 0.001$ ) and small target size ( $p < 0.001$ ). The ray-casting technique has faster speed when target size is medium than when target is small ( $p < 0.0167$ ). However, no significance is observed by changing the target size in either SQUAD or EyeSQUAD technique. These confirm our hypothesis (1).



**Figure 15: Interaction between technique and target size when fixing technique.**

Furthermore, we evaluate the effects of technique on the interaction with target size fixed (Figure 16). Under large target size scenario, ray-casting is significantly faster than SQUAD ( $p < 0.001$ ) and EyeSQUAD ( $p < 0.001$ ), and SQUAD is faster than EyeSQUAD ( $p < 0.0167$ ). When target is small, SQUAD is significantly faster than ray-casting ( $p < 0.0167$ ) while no significant difference is found either between ray-casting and EyeSQUAD ( $p = 0.100$ ) or between SQUAD and EyeSQUAD ( $p = 0.021$ ). No significant difference is found when target is medium. This indicates that ray-casting outperforms two other techniques with respect to selection speed only when the target is

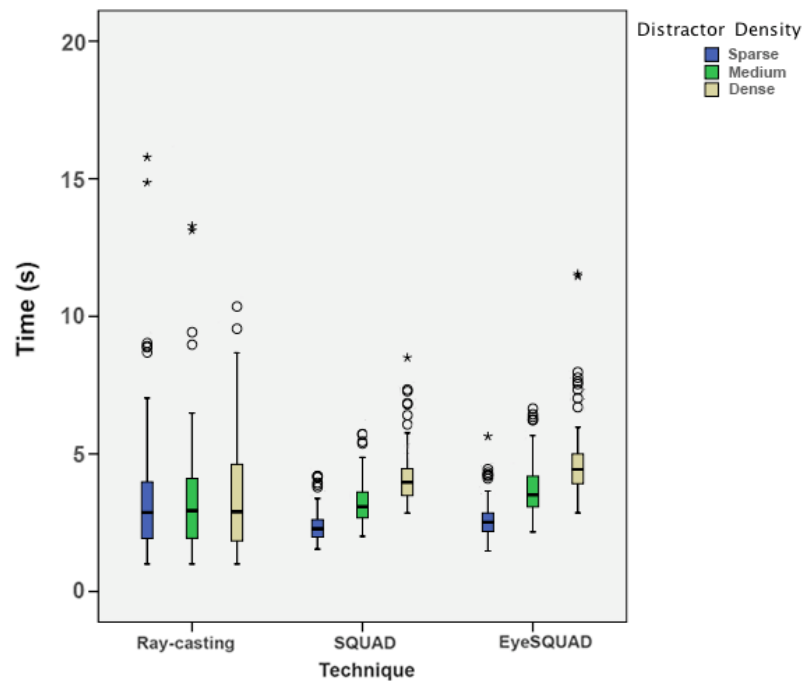
large, and SQUAD outperforms ray-casting when the target is small. Since SQUAD is faster than EyeSQUAD when target is large, the hypothesis (4) is weakened.



**Figure 16: Interaction between technique and target size when fixing target size.**

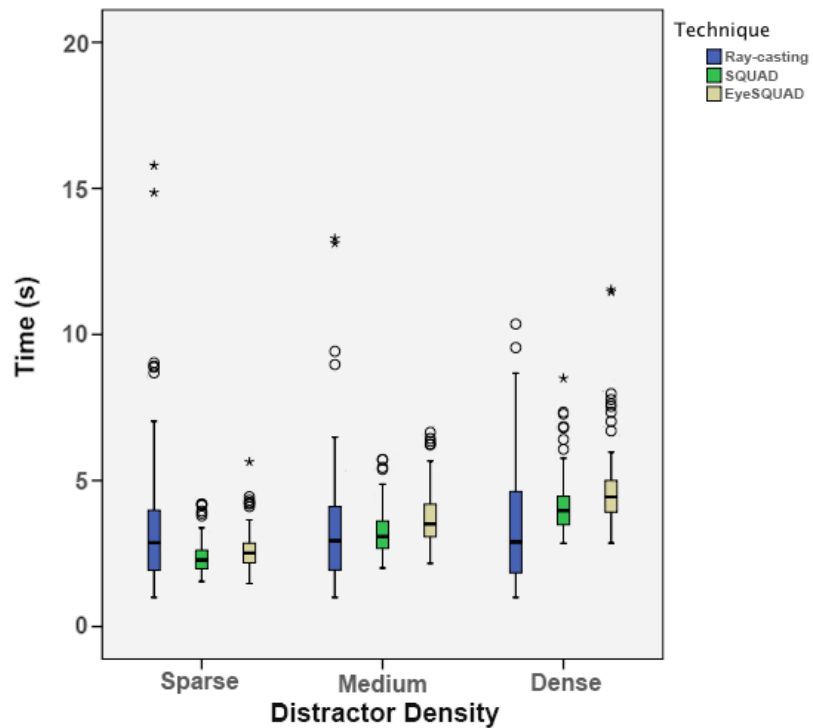
At first glance, the interaction of technique and density ( $F(0.95; 4,72) = 15.487, p < 0.001, \text{power} > 0.999$ ) also shows a significant effect on time in overall. When we delve into details of the interaction, significant differences can be found for the SQUAD and EyeSQUAD techniques under different distractor density scenarios when fixing technique (Figure 17). For SQUAD, sparse distractor density yields significantly less average selection time than medium distractor density ( $p < 0.001$ ). Also, medium

distractor density has significantly less average selection time than dense distractor density ( $p < 0.001$ ). Similarly, for EyeSQUAD, significant differences can be found among different distractor density scenarios (sparse and medium ( $p < 0.001$ ), medium and dense ( $p < 0.0167$ ), dense and sparse ( $p < 0.001$ )). This coincides with our hypothesis (2) that the selection time of techniques based on progressive refinement (i.e. SQUAD and EyeSQUAD) depends on the density of distractor in the environment which directly related to the number of selection steps. Besides, as expected, the distractor density has no significant effect on selection time when using the ray-casting technique which also supports hypothesis (2).



**Figure 17: Interaction between technique and distractor density when fixing technique.**

Moreover, if the distractor density is fixed, we can evaluate the importance of technique on the interaction of technique and distractor density (Figure 18). In any distractor density, no significant difference of selection speed can be observed between ray-casting and EyeSQUAD (sparse:  $p = 0.087$ , medium:  $p = 0.751$ , dense:  $p = 0.139$ ). However, significance can be found when looking into other pairwise comparisons. When the distractor density is sparse, ray-casting is significantly slower than SQUAD ( $p < 0.0167$ ) although no significance is observed between SQUAD and EyeSQUAD ( $p = 0.032$ ). Furthermore, SQUAD is significantly faster than EyeSQUAD when distractor density is medium ( $p < 0.0167$ ) and dense ( $p < 0.0167$ ). This indicates that significantly more time is spent on the quad-menu refinement process when using the EyeSQUAD compared with SQUAD. Previous results of error demonstrate that the SQUAD is more precise than EyeSQUAD in any distractor density. The nature of varying the distractor density is actually changing the refinement steps. Hence, the impact of the quad-menu selection process need to be examined for explaining the differences of performance between EyeSQUAD and SQUAD. Since the essential difference between EyeSQUAD and SQUAD in the quad-menu process is whether using a hand-control metaphor or an eye tracking metaphor, eye tracking technology from both system and human aspects could contribute to the differences in performance.

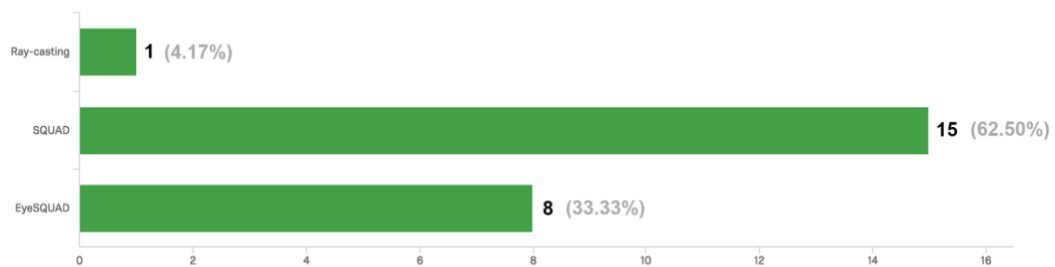


**Figure 18: Interaction between technique and distractor density when fixing distractor density.**

No significance is found in the interaction of technique, target size and distractor density ( $F(0.95; 8,144) = 0.764$ ,  $p = 0.636$ , power = 0.335). However, significance can be observed that ray-casting is faster than SQUAD ( $p < 0.001$ ) and EyeSQUAD ( $p < 0.001$ ) when target size is large and distractor density is dense. On the contrary, when target size is small and distractor density is sparse, SQUAD is faster than ray-casting ( $p < 0.0167$ ) though no significant difference is found between ray-casting and EyeSQUAD ( $p = 0.026$ ).

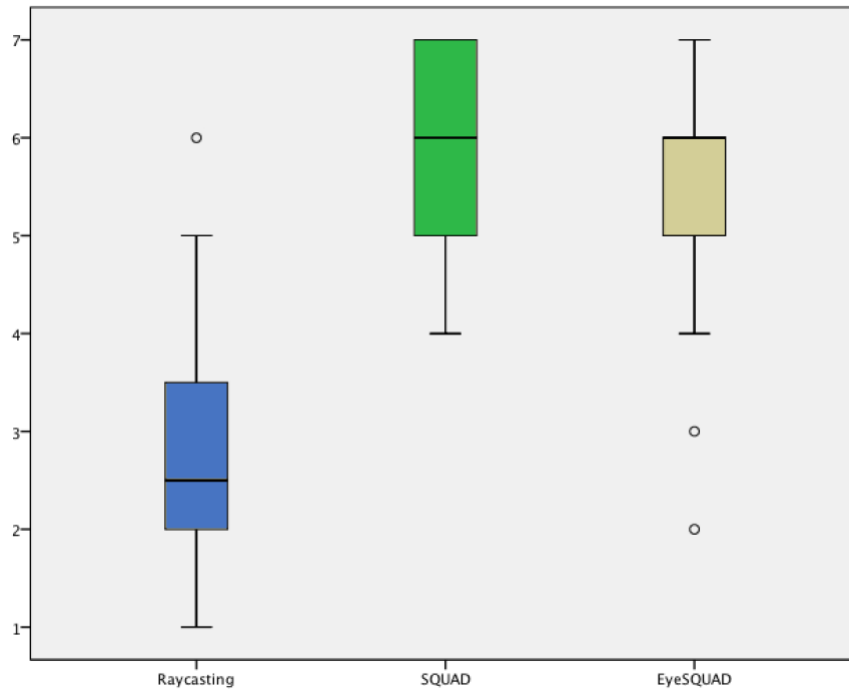
Under our experimental settings, no other significance of interactions is found. The order is found also not significant in the study on the time ( $F(0.95; 5,18) = 2.426$ ,  $p = 0.097$ , power = 0.558).

### 5.3 User Preference



**Figure 19: Overall user preference for the selection techniques.**

In general, all 24 participants were asked to complete post-study survey which is an overall performance questionnaire. In Figure 16, among all participants, 15 participants preferred the SQUAD technique and 8 participants favored the EyeSQUAD technique while only 1 participant would choose the ray-casting technique if needed to perform additional selection tasks. The only participant showed excellent performance in ray-casting without suffering from hand jitter since the participant has been playing piano since childhood and hence has steady hand movements. This participant is the one preferred ray-casting against other two techniques in the study.



**Figure 20: Overall average ratings of techniques.**

All participants rated each technique based on levels of 1 to 7 (1 to be very bad, 7 to be very good) right after they had performed certain technique. We performed a one-way repeated measure ANOVA on the ratings. The effect of technique is found significant ( $F(0.95; 2,36) = 47.502, p < 0.001, \text{power} > 0.999$ ) though the effect of order is found insignificant ( $F(0.95; 5,18) = 1.039, p = 0.425, \text{power} = 0.285$ ). In Figure 17, the mean ratings for ray-casting is 2.75 which is lower compared with 5.92 of SQUAD ( $p < 0.001$ ) and 5.33 of EyeSQUAD ( $p < 0.001$ ) with Bonferroni test. However, no significant difference is found between SQUAD and EyeSQUAD ( $p = 0.408$ ) which suggests that participants have similar favors for SQUAD and EyeSQUAD. These ratings coincide

with the above overall preference of techniques which SQUAD leads the favors and ray-casting is least preferable while from rating perspective, EyeSQUAD and SQUAD have similar preferences. Combining oral feedbacks from the experiments, we found that many participants prefer the SQUAD technique instead of the EyeSQUAD technique since they are more familiar with distal pointing tasks. For example, most individuals have been using remote controllers such as TV controller since childhood. However, several participants found that with EyeSQUAD, they can hold a controller in any comfortable position rather than always positioning it with certain gestures which causing hand and arm fatigue compared with SQUAD and hence they preferred EyeSQUAD. Although controlling with eye movements is novel, many participants said that they were eager to explore eye tracking techniques for not only selection but also more interaction techniques.

## 6. Discussion

The empirical results provide both expected and unexpected evidences that can be found reasonable after further analyzing the causes under the hood.

The error rate of ray-casting greatly increases with decreasing the target size and the SQUAD technique almost yields no errors (0.9% error rate), which coincide with the hypothesis (4) except the error rate of EyeSQUAD is not negligible. This unexpected high error rate of EyeSQUAD is mainly caused by several aspects. First, unlike holding a controller with hand like using the SQUAD technique, EyeSQUAD requires users to fixate at certain part of the screen when pressing the button on the controller during the quad-menu refinement process, otherwise a wrong part can be easily selected if the user blinks or looks away at the time of pressing the button. Results indicate that the performance of EyeSQUAD is weakened with increasing the refinement steps on the quad-menu. Many involuntary eye movements usually can be involved especially the user loses attention. This problem could be avoided if taking certain number of frames (e.g. 100 frames) ahead of getting selection commands and choosing the quadrant part with highest score rather than the part where the point-of-regard locates immediately when the user is pressing the button. By this way, the sensitivity of quad-menu selection could be greatly lowered down, providing a more user-friendly experience.

Secondly, apart from the design aspect, human aspects such as “lazy eyes” and dominant eyes (Porac and Coren, 1976) also need to be noticed when implementing eye

tracking techniques. EyeSQUAD currently takes equal weights for data of left eye and right eye to calculate the point-of-regard that mapped in three-dimensional space.

Redistributing the weights for left eye and right eye according to user's dominance of eyes may be a good choice for enhancement of stability and accuracy while can also suffer from implementation since the dominance of eyes is such a user specific variable which can be hard to measure. This redistributing the weights for left eye and right eye needs to be further studied before drawing deterministic conclusion.

Thirdly, the eye tracking stability and accuracy from the hardware could be a potential factor that contaminates the performance of EyeSQUAD. Since we noticed several participants encountered different extents of inaccuracy of eye tracking especially running the device for a period of time even without any movements of the headset. This could be caused by the heavy weights of the FOVE headset mounted with a Vive tracker with total weight of 820 g which should also be further checked before forming statements. As another system aspect factor, relatively low framerate (e.g. around 60 ~ 80 Hz and 44 Hz in the worst case) in EyeSQUAD compared with the ones of SQUAD and ray-casting (around 80 ~ 100 Hz) perhaps can be detrimental for the performance of the technique. This relatively low framerate seems to be caused by the eye tracking algorithms such as the approximation method whose cost depends on the number of potential targets in the environment. A certain amount of error might be removed if the framerate is fixed (e.g. 90 Hz).

Looking back to the results, the hypothesis (3) is proved because the EyeSQUAD outperforms ray-casting in all conditions with respect to error. However, the results disprove the hypothesis (4) since the SQUAD technique has higher accuracy and faster selection speed than EyeSQUAD on overall.

We have discussed the unexpected high error rate of EyeSQUAD above. Similarly, due to these system, human and design factors, participants need to be more careful when using the EyeSQUAD in the experiment in case of making more mistakes which results in longer selection time than expected in hypothesis. We believe this selection time of EyeSQUAD can also be improved if more attentions are paid to these aspects especially the error-tolerant rate of the technique that encapsulated in design which narrows down the barriers of using the technique as well as improves the performance.

We found the selection time of either SQUAD or EyeSQUAD is not affected by the target size while ray-casting will be slow with small targets and fast with large targets, which is consistent with our hypothesis (1). Besides, the selection time of ray-casting is not influenced by distractor density while the time of selecting a target with SQUAD or EyeSQUAD is proportional to the number of distractors, which is consistent with our hypothesis (2).

All in all, EyeSQUAD might be a taste of implementing selection techniques with eye tracking in virtual reality. Although EyeSQUAD cannot outperform SQUAD on both

accuracy and speed, we are not upset by the results. On the contrary, we are glad to explore the potentials of eye tracking techniques in virtual reality and find the reasons hidden in the unexpected empirical results. By carefully paying attention to several aspects such as system, design, human and eye tracking itself, we believe the performance of EyeSQUAD could be further improved greatly (e.g. error rate decreased from 17.4% in pilot to 6.2% in experiment mainly by carefully redesigning the quad-menu). Even if with current EyeSQUAD, hands-free interaction can be achieved once replaced the button input to a sound input. (We used the button input for EyeSQUAD is due to the consistency of using a controller as the input of selection commands as same as the other two techniques) A tongue click sound or head nod may be good choices to perform selections with EyeSQUAD, which will be one of the main goal of improvements after this study. Because the movements of eyes and head is decoupled, using nod as control commands is possible for not influencing fixation point. A complete user survey is necessary to determine which one is better since this commands by either tongue click or nod brings extra workload which may be biased by different individuals. Once EyeSQUAD becomes a completely hands-free selection technique, it can be utilized in some tasks where two hands are all occupied (e.g. car and plane driving, high precision required modeling) and even allow disabled individuals to be able to interact with virtual reality, which the previous selection techniques are difficult to achieve.

## 7. Conclusion

We designed a novel eye tracking selection technique with progressive refinement - eye-controlled sphere-casting refined by quad-menu (EyeSQUAD) selection technique. An approximation method is included in the EyeSQUAD to stabilize the calculated point-of-regard with expense of computational power. A user study was performed to examine the performance of the EyeSQUAD selection technique with comparison of two previous selection techniques under different target size and distractor density scenarios.

EyeSQUAD can achieve similar selection speed as ray-casting and SQUAD techniques although most participants are still not familiar with interaction techniques by eye movements. However, though has higher precision than ray-casting, EyeSQUAD is less accurate than SQUAD which may attribute to system, human and design aspects as well as eye tracking itself. Comparison between results in pilot study (17.4% error rate of EyeSQUAD) and actual experiment (6.2% error rate of EyeSQUAD) suggests that optimizing EyeSQUAD technique is possible from the design aspect.

Several design insights of designing eye tracking interaction techniques can be drawn from the user study and feedbacks of participants. First, the performance of the technique can be poisoned if there are visual stimulus in the environment due to the essence of using eyes as both input and output. Secondly, separating commands from

the eye tracking technique by using other inputs is necessary to prevent errors caused by involuntary eye movements.

In summary, we provided a new selection technique (EyeSQUAD) with the eye tracking metaphor which is rarely implemented for manipulation tasks in virtual reality. With selection by progressive refinement, this new technique can obtain better accuracy and precision than standard ray-casting technique when target is small. Additionally, transferring some workload from hands to eyes, we surprisingly found that similar performance can be achieved via controlling by eye movements as controlling with hands even with relatively low-cost eye tracking devices. This infers that implementing eye tracking into human-computer interaction techniques is possible to achieve similar performance as usual hand controls which could potentially extend the applicability and the population of the human-computer interactions.

In near future, the performance of EyeSQUAD can be further improved by careful design of the quad-menu. For selection in three-dimensional space in view of depth, the accuracy of the calculated point-of-regard can be optimized by separating the depth control of the convergence point from original calculation. To be clearer, instead of directly measuring the convergence point from eye ray data, the calculated point-of-regard can be casted onto an invisible plane which is controlled by the pupil size or extent of squinting of the user. Additionally, the input by pressing a button on controller

of EyeSQUAD can be replaced by a sound input such as “tongue click” sound or head nod to support completely hands-free experience.

## Appendix A Informed Consent Form

*Directions: Read this consent form carefully and ask any questions you may have. If you decide to continue with participation in this research study, please sign on the designated line in the final section and return this copy to the experimenter.*

**Title:** Evaluation of an Eye Tracking Selection Technique with Progressive Refinement

**Investigators:** Yunhan Wang, Mark Steelman, Regis Kopper

### 1) Exclusion Criteria

All participants must be:

- 18 years of age or older
- non-epileptic with no family history of epilepsy
- have normal or corrected-to-normal vision (have no color blindness)
- fluent in English

Because it is important to our research that all participants complete the VR tasks, if you have physical limitations or impairments that would make you unable to finish the tasks, please do not enroll in this study. We will screen you for color blindness prior to the experiment to determine eligibility.

### 2) Purpose of this Research

The purpose of this research is to examine if a novel new virtual reality selection technique (eye-controlled sphere refined by QUAD-menu selection technique) is better than ray casting selection technique in a cluttered environment, in terms of user performance. While we do not anticipate any direct benefits to you, your participation in this research could help us potentially advance the field of virtual reality.

### 3) Study Procedures

This research will take place in the Visualization Lab 1411, CIEMAS on Duke University's West Campus. The study will take 1 session, 40 minutes. If you agree to participate, you will wear a head-mounted display while using a 3D input device (controller) to experience and complete tasks in the virtual reality system. Before the study, you will first need to sign the informed consent form, and then be asked to complete demographics questions (gender, age, major) and a colorblind test. You can skip any questions, but you cannot continue the study if you skip the colorblind test.

The investigator will explain how you can use the input (either the controller or your own eyes) to perform selection. The target is highlighted by red color. The task is to select the target within a bunch of similar objects. You will utilize the input to control the movement of a selection ray/bubble. If you want to perform selection, you need to pronounce a "click" sound, which the

investigator will tell you how to do so in advance. You will select the object pointed by the ray casting in condition 1 or transform to a selection scene with a quad menu in it in other three conditions. In the latter case, you will “click” several times to refine the objects on the quad menu to perform further selection until obtain a single object. You may do this task several times under three different difficulty levels in one condition.

After completing the task, you will take a presence questionnaire (how real the experience felt), a usability questionnaire, and simulator sickness questionnaire. We will then switch condition, and you will utilize the other selection technique, and after completing the 3 questionnaires again (presence, simulator sickness, and input device usability).

You will also be asked to fill out additional questionnaires: a background survey to gather information about your experiences with virtual reality (VR) systems, and an overall preference questionnaire.

#### **4) Risks**

The only foreseeable physical risks are slight eyestrain, dizziness, and mild nausea during the VR Tasks. If you think you would not be able to finish the VR tasks, please do not agree to be in this study. During the study, you will be allowed to stop or skip the VR tasks at any time for any reason.

#### **5) Confidentiality**

We will attach a code to your data, which will not be connected to your name. Should the results of the study be published in a peer-reviewed psychology journal or be presented at conferences, only group data will be presented and no names or study IDs be included.

#### **6) Voluntariness**

Your participation in this study is completely voluntary. You are also free to withdraw from the study at any time without giving a reason. You can skip any questions but you cannot continue the study if you choose to skip the colorblind test.

#### **7) Compensation**

There is no compensation in this study.

#### **8) Further Information**

Should you have any questions about this research now, please ask the experimenter. Should you have any questions about this research later, please contact one of the investigators:

Yunhan Wang  
[yunhan.wang@duke.edu](mailto:yunhan.wang@duke.edu)  
919-536-8903

Regis Kopper  
[regis.kopper@duke.edu](mailto:regis.kopper@duke.edu)  
919-660 5570

Should you have any questions about your rights as a research subject, please contact:

Duke University Campus Institutional Review Board  
campusirb@duke.edu  
(919) 684-3030

**I have read this consent form and agree to be in this study.**

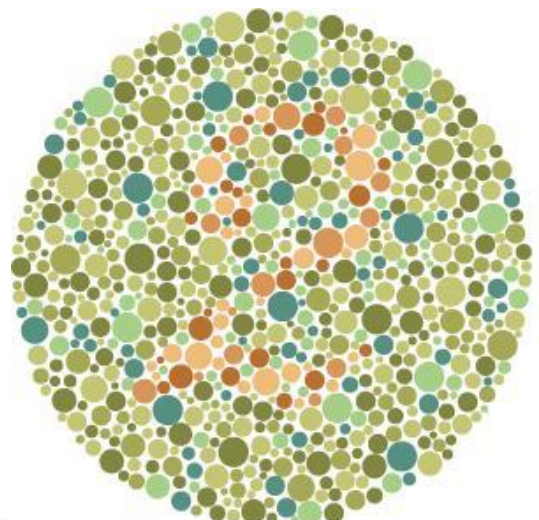
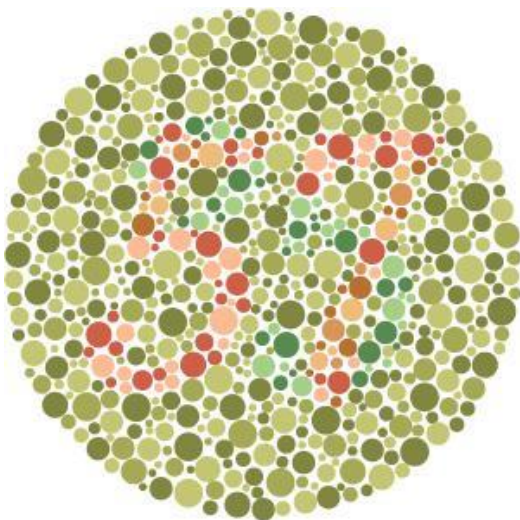
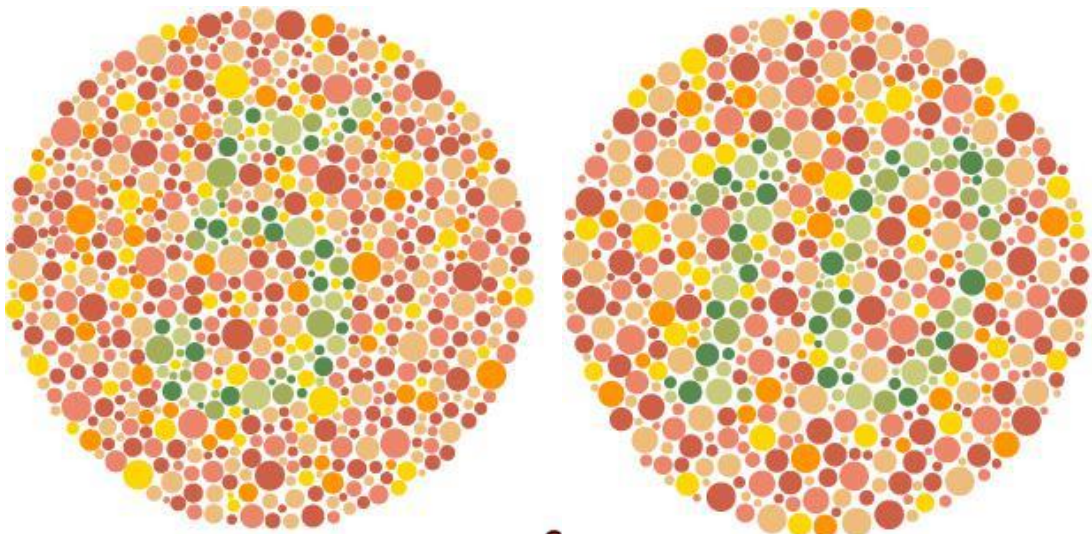
---

Participant Signature

---

Date

## Appendix B Color Blindness Assessment



## Appendix C Background Survey

### General

What is your gender? \_\_\_\_\_

What is your age? \_\_\_\_\_

What is your height? \_\_\_\_\_

What is your major? \_\_\_\_\_

Did you wear glasses or contacts during the study? Choose one.

No       Glasses       Contacts

Are you right- or left-handed or do you use both hands?

right       left       both

Are you a native speaker of English?

Native       Non-Native

### Experience with Virtual Reality

|   | <b>Ever experienced before?</b> | <b>EXPERTISE</b><br>(1=never, 7 = expert) | <b>HRS/WK</b> (last 6mths) | <b>5+ hrs/wk EVER?</b> |
|---|---------------------------------|---|----------------------------|------------------------|
|   |                                 |   |                            |                        |
| <b>CAVE-type system</b><br>(Eg. Multi-Wall Projector based VR system like the DiVE, which the experiment is being conducted in) | <b>Y / N</b>                    | <b>1 2 3 4 5 6 7</b>                      | <b>0 1-2 3-6 7-12 13+</b>  | <b>Y / N</b>           |
|   |                                 |   |                            |                        |

| <b>HMD – head mounted display</b><br>(Eg. Oculus Rift, HTV/Vive, Samsung Gear VR, Google Cardboard) | Y / N | 1 2 3 4 5 6<br>7 | 0 1-2 3-6 7-12<br>13+ | Y / N |
|---|-------|------------------|-----------------------|-------|
|   |       |                  |                       |       |



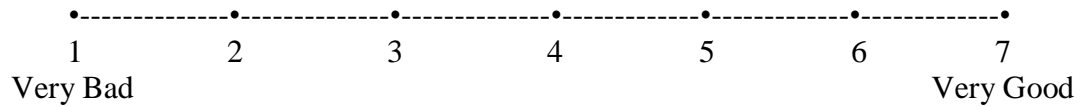




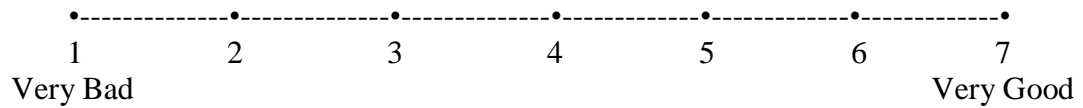
## Appendix G Overall Performance Questionnaire

What is your study ID? \_\_\_\_\_

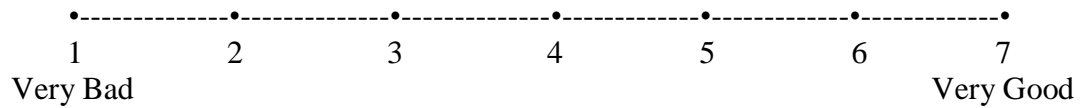
1. Please rate your preference of the ray casting technique



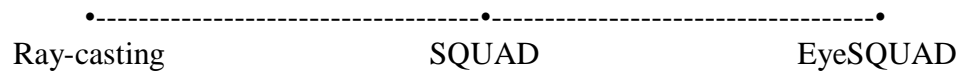
2. Please rate your preference for the SQUAD technique



3. Please rate your preference for the EyeSQUAD technique



4. If you had to choose a device to perform additional selection tasks, would you choose?



5. Please give any additional comments you may have.

## Appendix H Experiment Oral instructions

### [Initial Instructions]

Welcome and thank you for your participation in our experiment. You will help us evaluate different selection techniques by accomplishing several selection tasks with different techniques in virtual reality. The objects' size and density will vary.

The goal of this study is to compare different selection techniques. You need to perform the trials as quickly as possible while making as few mistakes as you can. Making fewer mistakes is more important than being quick.

[give the informed consent]

Before start of the experiment, please first read and sign an informed consent.

[show my laptop]

Then, let's take a simple color blindness assessment and background survey online.

[After finishing the color blindness assessment and background survey]

Now, I will explain how to complete the tasks.

[show the HTC Vive controller] This is the input device you will use through all the trials. Hold it with your dominant hand. The only button you will use is the touchpad. Hold it in a comfortable position and use your thumb to press its middle/upper part. During the experiment, you will not be allowed to use your other hand to steady the device.

Now move to the experiment area.

[give FOVE headset] This is the FOVE headset which you will use to see the virtual contents. Please put on it and fasten the belts to a comfortable position.

[open the training app] [press O and R]

Move to the starting point which is a red dot on the ground in the virtual environment, it will turn to green once you reached it. You should not move away from that point during the trials. (Do not move your knees during selection)

You can see a grey controller which is consistent with the actual controller in the real world. Your goal is to get the target which is a red sphere in front you as soon as possible while making as few mistakes as possible.

### [ray-casting training session]

In this technique, you will use the controller to control the selection.

[open the training app, choose ray-casting] [press O]

Please don't click unless I tell you do so.

You can see the controller shots a purple ray into the environment. Your goal here is to use the ray to point at a target which is a red sphere in front you among blue distractors. To start the trial, select the large "Reset" circle in the center of the screen. Once you are pointing at the target, there will be a green bounding box around it. You

can press the touchpad on the controller to perform the selection. If you successfully got that target, a check mark will appear. On the contrary, a cross will be displayed if failed. Go ahead and make a mistake to see the cross. Now try to select the correct target to see the checkmark. The next trial will be automatically loaded every time you finished a selection.

Go ahead and try several times of the ray-casting condition. Let me know if you have any questions.

[after few minutes training]

### **[ray-casting experiment]**

Let's start performing the ray-casting condition.

[open the experiment app, enter participant ID and choose the ray-casting condition]

Move forward to the starting point and tell me once you are ready.

[After got "ready" from the participant, press 0]

[participant performing the ray-casting condition]

[Once the participant finished the condition] Now you can remove the headset. There are few questions about the technique you just used.

[show my laptop]

[few minutes break]

### **[SQUAD training session]**

In this technique, you will use the controller to control the selection.

[open the training app, choose SQUAD] [press 0]

Please don't click unless I tell you do so.

To start the trial, select the large "Reset" circle in the center of the screen. (move the green dot into the circle and press the touchpad) You can see the controller casts a transparent bubble into the environment. Your goal here is to first use the bubble to catch the target (press the touchpad to select). (You don't need get the target exactly in the center of the bubble, just inside it.) Then you will see a QUAD-menu consists of 4 parts. The part hit by your ray will be highlighted as green. Choose the quadrant part that contains the target and press touchpad. Several steps you may get the target. If you missed the target during the selection process, a cross will be displayed indicating you failed. Otherwise, a check mark will appear if you successfully got the target. The next trial will be automatically loaded every time you finished a selection.

Go ahead and try several times of the SQUAD condition. Let me know if you have any questions.

[after few minutes training]

### **[SQUAD experiment]**

Let's start performing the SQUAD condition.

[open the experiment app, enter participant ID and choose the SQUAD condition]

Move forward to the starting point and tell me once you are ready.

[After got "ready" from the participant, press 0]

[participant performing the SQUAD condition]

[Once the participant finished the condition] Now you can remove the headset.

There are few questions about the technique you just used.

[show my laptop]

[few minutes break]

### **[EyeSQUAD training session]**

In this technique, you will use your eyes to control the selection. In this technique, you need to pay attention to ...

[open the training app, choose EyeSQUAD] [press 0]

For the EyeSQUAD condition, we need to first do the calibration of eyes.

[perform the calibration] Please use your eyes to follow a green dot.

[finished calibration] (Do not move the headset after calibration.)

Please don't click unless I tell you do so.

To start the trial, select the large "Reset" circle in the center of the screen. (move the green dot into the circle and press the touchpad) You can see the transparent bubble now moves with your eyes. Your goal here is to first use the bubble to catch the target (press the touchpad to select). (You don't need get the target exactly in the center of the bubble, just inside it.) Then you will see a QUAD-menu consists of 4 parts. The part where you look at will be highlighted as green. Choose the part that contains the target and press the touchpad. Make sure do not move your eyes, or blink or look away when pressing the button. After several steps you may get the target. If you missed the target during the selection process, a cross will be displayed indicating you failed. Otherwise, a check mark will appear if you successfully got the target. The next trial will be automatically loaded every time you finished a selection.

Go ahead and try several times of the EyeSQUAD condition. Let me know if you have any questions.

[after few minutes training]

### **[EyeSQUAD experiment]**

Let's start performing the EyeSQUAD condition.

[open the experiment app, enter participant ID and choose the EyeSQUAD condition]

Move forward to the starting point and tell me once you are ready.

[After got "ready" from the participant, press 0]

[participant performing the EyeSQUAD condition]

[Once the participant finished the condition] Now you can remove the headset.  
There are few questions about the technique you just used.  
[show my laptop]

[After finishing all the conditions]  
There is an overall performance questionnaire about all the techniques you just used.  
[show my laptop]

## References

- Bowman, D., Kruijff, E., LaViola Jr, J. J., & Poupyrev, I. P. (2004). 3D User interfaces: theory and practice, CourseSmart eTextbook. Addison-Wesley.
- Deubel, H., & Schneider, W. X. (1996). Saccade target selection and object recognition: Evidence for a common attentional mechanism. *Vision research*, 36(12), 1827-1837.
- De Haan, G., Koutek, M., & Post, F. H. (2005, October). IntenSelect: Using Dynamic Object Rating for Assisting 3D Object Selection. In IPT/EGVE (pp. 201-209).
- Duchowski, A. T. (2007). Eye tracking methodology. *Theory and practice*, 328.
- Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6), 381.
- Frees, S., Kessler, G. D., & Kay, E. (2007). PRISM interaction for enhancing control in immersive virtual environments. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 14(1), 2.
- Field, A. (2013). *Discovering statistics using IBM SPSS statistics*. sage.
- Goldberg, J. H., & Wichansky, A. M. (2003). Eye tracking in usability evaluation: A practitioner's guide. In *the Mind's Eye* (pp. 493-516).
- Jacob, R. J., & Karn, K. S. (2003). Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. In *The mind's eye* (pp. 573-605).
- Kaur, M., Tremaine, M., Huang, N., Wilder, J., Gacovski, Z., Flippo, F., & Mantravadi, C. S. (2003, November). Where is it? event synchronization in gaze-speech input systems. In *Proceedings of the 5th international conference on Multimodal interfaces* (pp. 151-158). ACM.
- Kopper, R., Bacim, F., & Bowman, D. A. (2011, March). Rapid and accurate 3D selection by progressive refinement. In *3D User Interfaces (3DUI), 2011 IEEE Symposium on* (pp. 67-74). IEEE.
- Porac, C., & Coren, S. (1976). The dominant eye. *Psychological bulletin*, 83(5), 880.

- Poupyrev, I., Ichikawa, T., Weghorst, S., & Billinghurst, M. (1998, August). Egocentric object manipulation in virtual environments: empirical evaluation of interaction techniques. In *Computer graphics forum* (Vol. 17, No. 3, pp. 41-52). Blackwell Publishers Ltd.
- Pfeiffer, T. (2008). Towards gaze interaction in immersive virtual reality: Evaluation of a monocular eye tracking set-up. In *Virtuelle und Erweiterte Realität-Fünfter Workshop der GI-Fachgruppe VR/AR*.
- Tanriverdi, V., & Jacob, R. J. (2000, April). Interacting with eye movements in virtual environments. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (pp. 265-272). ACM.
- Vanacken, L., Grossman, T., & Coninx, K. (2007, March). Exploring the effects of environment density and target visibility on object selection in 3D virtual environments. In *3D User Interfaces, 2007. 3DUI'07. IEEE Symposium on*. IEEE.