·Review·

Review of studies on target acquisition in virtual reality based on the crossing paradigm

Susu HUANG[1], Daqing QI[2], Jiabin YUAN[1], Huawei TU[1*]

1. *College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing* 211106, *China*

2. *Jiangsu Hongao Mdt InfoTech Co., Ltd., Lianyungang* 222000, *Jiangsu, China*

* **Corresponding author,** tuhuawei09@gmail.com
Received: 14 January 2019   **Accepted:** 28 April 2019

Abstract   Crossing is a fundamental paradigm for target selection in human-computer interaction systems. This paradigm was first introduced to virtual reality (VR) interactions by Tu et al., who investigated its performance in comparison to pointing, and concluded that crossing is generally no less effective than pointing and has unique advantages. However, owing to the characteristics of VR interactions, there are still many factors to consider when applying crossing to a VR environment. Thus, this review summarizes the main techniques for object selection in VR and crossing-related studies. Then, factors that may affect crossing interactions are analyzed from the perspectives of the input space and visual space. The aim of this study is to provide a reference for future studies on target selection based on the crossing paradigm in virtual reality.

Keywords   Virtual reality; Target selection; Crossing paradigm; Pointing paradigm

1 Introduction

Virtual reality (VR) consists of a computer simulation system that can generate three-dimensional (3D) virtual environments. VR provides users with a visual representation of the simulation, which allows them to interact with objects in the 3D space. Human-computer interaction provides multiple interaction modes for VR according to different functions and purposes. This paradigm was first introduced to virtual reality (VR) interactions by Tu et al.,[1]. For example, professional devices, such as head-mounted displays, data gloves, or joystick controllers, allow users to perceive and manipulate various virtual objects in real time with a realistic feeling.

Target acquisition is a basic task in the VR environment. An effective target selection method enables users to perceive and manipulate virtual objects in a natural manner. As illustrated in Figure 1, there are two mainstream VR target selection categories: ray-casting techniques[2-5] and virtual hand techniques[6-8].

Virtual hand techniques simulate the movement of the human hand through the motion of VR sensors. The advantage of virtual hand techniques is that they simulate human hands for target selection, in line with users habits of operating objects in the real world. However, the virtual hand technique has some

limitations. For example, it is difficult to choose distant targets[7].

Ray-casting techniques can easily allow the selection of distant objects, and require relatively small hand movements, and so they have come to represent the main selection method in current commercial VR devices (such as Oculus Rift and HTC Vive). Many studies have verified that ray-

Figure 1   Target selection techniques in VR. (a) Ray-casting techniques; (b) Virtual hand techniques.

casting techniques are superior to virtual hand techniques[9,10]. However, when ray-casting techniques select a distant or small target, a Heisenberg effect occurs (the hand needs to maintain a stable posture when selecting the target, but in practice the hand may shake), and so the target selection accuracy is decreased. To address this limitation, some studies[11-13] have increased the size of the ray cursor to improve the performances of ray-casting target selection techniques. However, large ray cursors may select multiple targets simultaneously. While this can be mitigated by employing ambiguous elimination methods to determine the intended target, this increases the selection time[13].

Tu et al. [1] first introduced the crossing paradigm to VR for target acquisition in 2018. The crossing-based target selection method implies that the target can be selected by crossing the boundary of the target. They studied the crossing-based target selection performance in the VR environment for the first time, in comparison to pointing-based target selection interactions (Figure 2). Two experiments were performed to cross discs in a 3D space and straight lines on a two-dimensional (2D) plane. Five factors were considered in their experiments: the task difficulty, direction of movement constraint (collinear *vs* orthogonal), nature of the task (discrete *vs* continuous), field of view of the VR device, and target depth.

Figure 2 Two target selection paradigms[14]. (a) Pointing; (b) Crossing.

The experimental results showed that the crossing-based selection method could compensate effectively for or replace the pointing selection method. Furthermore, the selection task based on the crossing paradigm can be modelled by Fitts' law. It was also observed that the target depth has a significant impact on the performance of a crossing interaction. For the two tasks of selecting a disc in 3D space and selecting a straight line on a 2D plane, clear differences were observed in terms of the operation time, error rate, and Fitts' model parameters. On the basis of these findings, the authors proposed several suggestions for crossing-based target selection design in VR. For example, when selecting small or distant targets, priority should be given to the crossing paradigm. For continuous crossing-selection operations, the target can be orthogonal or collinear with the direction of motion, and for discrete crossing-selection operations the target should be collinear with the direction of motion.

Tu et al.'s research laid the theoretical foundation for applying the crossing paradigm to VR interactions. However, their study only considered five factors. If the crossing paradigm is to be applied to more general VR environments, then the characteristics of VR interactions should be more comprehensively taken into account. In view of this goal, this review systematically summarizes a number of factors affecting crossing interactions in VR environments from the perspe-ctive of the input space and visual space and proposes a corresponding category framework. The purpose of this work is to provide a reference for better utilization of the crossing paradigm in VR.
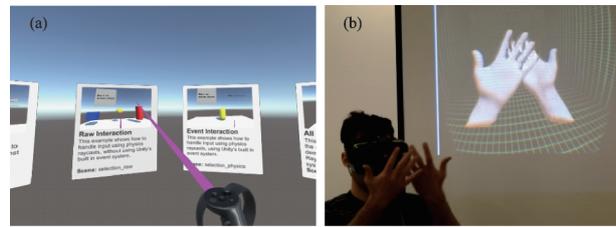
2 Crossing-related studies

Fitts' law[15] can be used to model the task time in the crossing paradigm. Therefore, Fitts' law will first be briefly introduced here before introducing related research on crossing. The law has been extensively employed to predict target selection times in human-computer interactions. It can be expressed by the formula

$$T = a + b log2 (D/W + 1) \tag{1}$$

and the difficulty index (ID) is defined as

$$ID = log2 (D/W + 1) \tag{2}$$

Here, a and b are empirical parameters, which depend on the specific physical characteristics of pointing devices and other factors such as operators and environments; D represents the target distance; and W represents the target size. The time at which the pointing device reaches the target is proportional to D and inversely proportional to W. This law has provided an important theoretical basis for studying the crossing paradigm.

The crossing paradigm has been applied to various interaction scenarios since its introduction. The following provides a summary of crossing research from four aspects: crossing with a pen input, mouse input, direct touch, and in a VR environment.

2.1 Crossing with a pen input

The crossing paradigm was first introduced for pen input, which was first proposed by Accot et al. in an experiment of deriving the Steering law[16]. Subsequently, in 2002 Accot and Zhai[14] first examined the crossing paradigm as a complement to the pointing interaction. For an indirect pen input, they found that the completion time of each task depended on the ID of Fitts' law for four crossing tasks and two pointing tasks. In cases with the same ID, the time for the crossing task was no longer than that for the pointing task. Inspired by the work of Accot et al., which was based on an indirect pen input (the pen input and display were separated), Forlines and Balakrishnan designed six tasks (four crossing tasks and two pointing tasks) similar to those of the Accot[14] experiment based on indirect and direct pen inputs[17]. The results showed that the performance of a crossing-based direct input were superior to those for an indirect input. The pointing-based direct input was observed to be essentially equivalent to the indirect input. It was found that the times for the pointing- and crossing-based target selections were reduced after tactile feedback was added. Such results indicate that reasonable feedback could shorten the target selection time.

In terms of crossing-based applications, Apitz and Guimbretièreproposed a pen-based drawing application called Cross Y, which employed crossing rather than pointing as its target acquisition method[18]. Their work demonstrated the feasibility of a crossing-based interactive interface, and that the crossing interaction could integrate multiple commands into one stroke, unlike the pointing interaction. In 2008, Dixon et al. explored the minimum space requirements for pointing- and crossing-based dialog boxes from the perspective of the interface design[19]. Their results showed that the space requirements with crossing-based dialog boxes were smaller than for pointing-based cases. On this basis, they proposed a revised crossing-based dialog box design.

In general, the crossing paradigm introduced for pen interactions was proved to supplement the pointing paradigm. It has advantages in the selection of continuous targets, and the convenience for selecting multiple targets through one stroke is greater than for the pointing paradigm.

2.2 Crossing with a mouse input

Some studies have also focused on crossing with a mouse input. Previous studies have shown that patients with motor disabilities face considerable challenges when using traditional mice and trackballs. One challenge is to locate the mouse cursor in a limited target area. Another is to perform accurate pointing. These problems may make it very difficult for people with motor disabilities to point on a graphical interface. To address this problem, Wobbrock et al. introduced crossing into mouse interaction. Because pointing-based target selection requires the mouse cursor to cross the boundary of the target, crossing-based target selection can circumvent the abovementioned problems[20]. Their research results showed that the crossing paradigm was superior to the pointing paradigm for users with motor disabilities, and they provided design guidance for crossing-based target selection interfaces for mouse interactions.

For mouse interaction applications, Dragicevicproposed combining crossing and paper interactions to drag and drop files between overlapping windows[21]. In 2008, Sulaiman et al. designed an attribute gate technique based on the crossing paradigm[22]. This can be utilized to solve the problem of setting object attributes and moving objects simultaneously on a digital desktop. It also allows users to operate multiple subtasks in a sliding motion. Yoshikawa et al. proposed four design principles for crossing small goals on the desktop, namely ease of use, security, efficiency, and scalability for users with motor disabilities[23]. Perin et al. designed a multi-slider simultaneous control technique based on the crossing paradigm[24]. This made use of the characteristics of the crossing paradigm to easily select continuous targets.

2.3 Crossing with direct touch

As touch input has been increasingly adopted in mobile phones, tablets, and other devices, some studies have introduced the crossing paradigm into this interactive mode. In 2014, Luo and Vogel investigated the crossing paradigm with direct touch interactions for the first time[25], and designed six tasks similar to those of Accot et al.[14]. The research results showed that crossing took no longer than pointing with direct touch. Unlike indirect pen and mouse interactions, not all types of direct touch tasks could be accurately modeled by Fitts' law. The results provided necessary support for crossing-based interactions in touch interfaces.

In addition, many crossing-based application techniques have been designed. In 2015, Luo and Vogel designed the pin-and-cross technique based on their previous research on direct touch[26]. This can present pop-up menus after the first touch or drag. These menus have a line segment as a pin, and each pin corresponds to a command. When the user crosses the corresponding pin with their finger, the corresponding command is executed. Users can transmit various commands (such as paste and copy) to the target while dragging it. From the pin-and-cross and fold-and-drop techniques, it can be inferred that the crossing paradigm can be combined with other interactive paradigms, such as dragging.

The application of the crossing paradigm is not limited to devices such as tablets or mobile phones. Intelligent touch control technology has gradually become a popular topic. Researchers have begun to consider how to incorporate the target selection function in gesture designs, to enable users to control intelligent appliances. Owing to the conflict between the pointing paradigm and gestures concerning continuity, Ren et al. proposed a crossing-based gesture in the design of interactive TV gestures to solve the problem of gesture separators[27]. They combined the crossing paradigm with gestures, and proposed a new approach for target selection in virtual environments.

2.4 Crossing in a VR environment

With the development of VR technology in recent years, target selection in VR environments has become a

popular topic in the VR and human-computer interaction fields. It is difficult to select distant and small targets based on ray-casting pointing techniques. Given the advantages of crossing over pointing, Tu et al.[1] introduced the crossing paradigm into VR for the first time. The results showed that the performance of the crossing paradigm was no worse than that of the pointing paradigm, and it was verified that the target acquisition time in VR can be modeled by Fitts' law. They also provided VR developers with suggestions and guidance for designing VR interfaces. However, there are many other factors that Tu et al. did not consider, such as the selection of occluded targets, so that their conclusions may not be more widely employed in VR interactions. This study will analyze a number of factors concerning crossing in VR environments, to extend the usability of the crossing paradigm in VR.

3 Results

Because VR involves a 3D environment, interactions in VR are different from those with a pen, mouse, or touch. In view of such differences, this section proposes a classification framework, and analyzes the factors in virtual environments from the perspectives of the input space and visual space (Table 1).

Table 1   Classification framework of factors affecting the crossing interaction in a VR environment

Category	Factors	Factor Details	Influence
Input space	Input Mode	Virtual hand	Difficulties in selecting distant targets
		Ray-casting	Difficulties in selecting small targets
	Control-Display Ratio	Manual switch	Manually adjust C/D ratio
		Target oriented switch	Dynamically adjust the C/D ratio according to the target distance
		Speed oriented switch	Dynamically adjust the C/D ratio according to input device speed
	Feedback Type	Visual feedback	Need good eyesight
		Tactile Feedback	Reduce the time of target selection; affect the accuracy of selection to some extent
		Audio feedback	Easy to expose privacy, need to wear headphones
Visual space	Target Dimension	2D	Crossing performs better than 3D
		3D	Crossing performed worse than 2D
	Target Shape	Distant and small targets	The crossing paradigm is better than the pointing paradigm
		Flat targets	The crossing paradigm is better than the pointing paradigm
	Target Depth	Environment with background	Better perception of depth information
		Environment without background	Poor perception of depth information
	Target Distance and Size	Reduce the distance between targets	Shorten the time of target selection
		Increase the size of targets	Shorten the time of target selection
	Field of View	Targets in the field of view	No need for the head to locate targets; save time for target selection
		Targets outside the field of view	Need for the head to locate targets
	Occlusion	Proximity	Switch viewpoints to solve occlusion problems
		Intersection	Combine traditional methods with the crossing paradigm to better solve occlusion problems
		Enclosement	Combine traditional methods with the crossing paradigm to better solve occlusion problems
		Containment	Combine traditional methods with the crossing paradigm to better solve occlusion problems

3.1 Input space

This section analyzes crossing in VR environments in terms of three factors for the input space. First, unlike pen, mouse, and touch inputs, the main input methods of VR are ray-casting and virtual hand techniques. Second, the control-display ratio is an important factor affecting the accuracies of input devices

in the input space. Finally, the feedback types of the input space also impact the performance of crossing.

3.1.1 Input mode

The input mode is undoubtedly a major factor affecting crossing in VR. The traditional input methods are the pen, mouse, and direct touch. For example, the six tasks proposed by Accot et al.[14] in 2002 to compare crossing and pointing were based on a pen input. Wobbrock et al. [20] introduced crossing for mouse interactions in 2008, and demonstrated the feasibility. Luo et al.[25] designed a set of the same six tasks as Accot et al. [14]. The only difference was that the input mode was changed from pen to direct touch. They found that the crossing performance with direct touch differed from those for the pen and mouse. The time for the clicking task could not be modeled by Fitts' law. From the above research, it can be concluded that the performance of crossing varies across different input modes.

The main input mode in VR is the ray-casting technique. Many studies have verified the effectiveness of the ray-casting technique. The major VR devices, such as Oculus Rift and HTC Vive, use the ray-casting technique to select targets. Tu et al.[1] combined the crossing and ray-casting techniques, and found that ray-casting crossing outperformed that ray-casting pointing for many tasks. As another complementary technology for target selection, the virtual hand technique has its own advantages in some cases. Its unique virtual hand is more in line with human cognition, and can increase user immersion. Being restricted by the inaccuracy in selecting small or distant targets in the VR environment, it is not widely employed. However, if the virtual hand technique and crossing paradigm can be combined to solve the problem of selection, then the virtual hand technique may be more widely adopted.

Hence, input methods impact crossing interactions in VR environments. For example, crossing-based ray casting can solve the problem of small target selection, while the virtual hand is more in line with a user's habits when selecting close targets.

3.1.2 Control-display ratio

The control-display ratio is abbreviated as the "C/D ratio". This represents the ratio of the moving distance between a controller and cursor. When the ratio is large, the sensitivity of the controller is low. In this case, the controller moves for a long distance, and the cursor moves for a short distance. For ray-casting techniques in VR environments, when the target is close to the user the motion of the cursor is relatively small. Then, the control sensitivity is low, and a larger motion leads to less cursor motion, so that the user may select the target more accurately. When the target is far away from the user, the motion of the cursor is large. At this time, the control sensitivity is high. A smaller motion leads to greater cursor motion, and the user cannot select a distant target very accurately. Therefore, the C/D ratio is another important factor affecting the crossing performance in a VR environment. The movement of the pointer is controlled according to human motion in the direct input environment (pen or finger). The VR environment contains a large space, and the display interface represents the entire 3D space. However, the scope of control is only in the range of people's activities, and so choosing the correct C/D ratio has always been a key issue in the research on target acquisition in VR. For example, Poupyrev et al.[7] proposed a dynamic C/D ratio method to solve the problem that virtual hands cannot select distant targets. They adjusted the C/D ratio based on the distance between the user's hand and torso, increasing the target selection range of the virtual hand technique. Techniques based on the C/D ratio in virtual environments can now be divided into three categories: manual switching, target-oriented, and speed-oriented techniques.

The **manual switching** technique allows users to manually control the C/D ratio. For example, Vogel et al. proposed the utilization of gestures to manually adjust the C/D ratio for the ray-casting technique[28].

The **target-oriented** technique increases the C/D ratio as the selection tool enters or approaches the target. For example, Andujar et al. proposed a technique for automatically adjusting the C/D ratio based on the width and height of the graphical user interface window, to achieve a lower target selection error rate[29].

The **speed-oriented** technique dynamically adjusts the C/D ratio based on the speed of the input device. For example, the PRISM technique proposed by Frees and Kessler[30] applies a speed-oriented C/D ratio technique to target selection in a 3D space.

These C/D ratio techniques have been applied to the virtual hand technique. If the control/display ratio can be dynamically adjusted, then the ray-casting technique based on crossing can enhance the selection efficiency. Specifically, when the ray is close to the target the C/D ratio is increased to improve the target selection accuracy, and when the ray is far away from the target the C/D ratio is decreased to improve the motion efficiency of the controller. In general, the former is intended to improve the accuracy of the selection, while the latter is intended to increase the speed of selection.

The manual switching technique requires manual control by the user. This is inconvenient to use, and is not widely employed. Nowadays, target-oriented and speed-oriented automatic switching techniques are widely employed in the VR environment. It is of considerable research value to select different C/D ratios according to the distance or speed of the ray relative to the target to optimize target selection based on crossing.

3.1.3 Feedback type

For pen, mouse, and touch inputs, the main feedback type for crossing is visual feedback. In 2008, Forlines et al. studied the impact of tactile feedback on pointing and crossing tasks for a pen input[17]. Their results showed that tactile feedback could reduce the target selection time to a certain extent, most obviously in the crossing task. In addition, they hypothesized that tactile confirmation of a successful selection allows users to move onto the next target selection task more quickly. In other words, the benefit of tactile feedback is not that it improves the selection time, but rather that it provides confirmation of the selection without requiring visual attention. In 2014, Stuerzlinger[31] found that in the 3D pointing environment the error rate with highlighting of the target was half that with tactile feedback, but the target selection time was increased. In the study of Tu et al.[1], visual and audio feedback were combined. Visual feedback was utilized to change the color of the target, and audio feedback was provided to remind users when the target was not selected. Visual and tactile feedback also have limitations. If the user's vision is not very good when selecting small or distant targets, then the effect of visual is considerably decreased. Tactile feedback is transmitted to the user through vibration of the controller, which may affect the target selection accuracy.

However, not all tactile feedback in the virtual environment is rendered through the controller: it can be rendered through other hardware. Under the premise that tactile transmission does not affect the user, tactile feedback can be given priority. After all, visual feedback is affected by occlusion. Audio feedback does not suffer from the problem of affecting the user's control as tactile feedback does, nor from occlusion problems. However, audio feedback may breach privacy, although this can be solved by wearing headphones in a single-user environment. Therefore, in VR environments different feedback modes should be selected for different situations. Different feedback modes have different effects on crossing-based target selection.

In summary, the setting of the feedback mode in the VR environment is crucial for crossing-based target selection, and researchers and designers should select the appropriate feedback mode according to the specific situation.

3.2 Visual space

This section analyzes the crossing performance in a VR environment from the perspective of the visual space. Specifically, this section is focused on six factors: the target dimension, target shape, target depth, target distance and size, field of view, and occlusion. First, because there are both 3D and 2D objects in a 3D virtual space, the dimension of an objects affect the performance of crossing-based target selection. Second, when the user selects a target in the VR environment, the shape of the target is an important contributing factor. For example, crossing is convenient when selecting a flat target, but may be slower than pointing for large cubes. Third, as a major research focus in VR the field of view is also an important factor in determining the crossing performance. Fourth, it can be observed from Fitts' law that the distance to and size of the target are important factors affecting crossing in VR. Fifth, one of the reasons why crossing in 3D space is different from that in 2D space is that depth information is included in 3D space, and so the target depth influences the crossing performance. Finally, occlusion is also an important factor affecting the performance of crossing in VR.

3.2.1 Target dimension

The dimension of the target is a major factor in target selection in VR. The pen, mouse, hand, and other traditional input environments have mainly 2D targets, while VR environments have 3D targets. For example, the studies[3,8,28] consider the selection of 3D objects. There are also 2D targets (mainly for menus or toolbox interfaces)[32-36], which consider 2D objects in 3D space. Pierce et al.[12] proposed image-plane interaction technology to transform a 3D target in space into a 2D target in a plane, and optimized the virtual hand technology. It can be concluded that virtual hand technology is more suitable for target selection in a 2D plane. In an experiment on crossing-based target selection, Tu et al.[1] found that selecting a line segment on a 2D plane was faster and more accurate than selecting a disc in a 3D space, but the reverse was true for the pointing task. Therefore, researchers should fully consider the target dimension to improve the target selection efficiency when designing crossing-based interaction interfaces.

3.2.2 Target shape

Objects have many shapes in VR environments. It is not very convenient in ray-casting pointing to select small or far away targets, especially targets such as discs and line segments. Therefore, researchers have begun to enhance the ray-casting technique, and have proposed the spotlight technique[37], silk cursor technique[8], and others to solve the problem of selecting small targets. Tu et al. introduced the crossing paradigm into VR to solve the small target selection problem[1]. The targets in their experimental tasks consist of discs and line segments, and the experimental results demonstrate the effectiveness of crossing in selecting such small targets. Of course, the crossing paradigm only supplements the pointing paradigm. Not all target shapes are suited to selection by crossing. For example, the speed of pointing is clearly faster than that of crossing when selecting large spheres. This is because the rays need to be moved a long distance when crossing the sphere, in which case the pointing performance is better than that of crossing. Therefore, crossing does not provide a complete replacement for pointing, but rather has a complementary relationship with pointing in VR. When designing crossing interactions, the shape of the target should be small or flat, so that the advantages of crossing can be fully exploited.

In summary, the impact of the target shape on the crossing performance should be considered in VR. Relevant researchers and developers should understand what target shapes are suitable for the pointing paradigm, and which are suitable for the crossing paradigm.

3.2.3 Target depth

A significant difference between VR and non-VR target selection is that there is a depth dimension in the VR environment. The target depth refers to the distance between the target and user. A virtual target can be located in various positions in the virtual environment, which leads to different target depths. The most intuitive effect of the depth is that for targets of an identical size, the larger the depth the smaller the image in the user' view, and vice versa. In general, the depth of the target determines the size of the target when viewed by the user. As an important feature in VR, depth has attracted considerable research focus. For example, in the studies[38,39] on pointing-based 3D target selection in a VR environment, the depth is considered as an important factor, and many studies have considered depth perception in a VR environment[40,41]. Tu et al.[1] utilized depth information as a control variable, to study the effects of different depths on crossing and pointing tasks. The experimental results showed that depth information has a significant impact on crossing tasks. At the same time, to increase the accuracy of the user's depth perception, they added a background environment (a textured wall) to help users better perceive the spatial depth. In addition, Tu's research[38, 42, 43] also added similar background environments to enhance the user's depth perception. The official Oculus documents also explicitly limit the range of the target depth, preferably between 0.75m and 3.5m. Therefore, the depth of the target is an important factor in crossing selection in a VR environment. An appropriate background environment can enhance the user's depth perception.

3.2.4 Target distance and size

The target distance and size directly impact the selection of a crossing-based target in a VR environment. Common approaches to shortening the selection time include increasing the target size and shortening the target distance. For example, Dachselt et al.[44] reduced the value of the ID and the difficulty of a task by minimizing the distances between targets, which led to a shorter target selection time. However, certain problems occur when this method is employed in the crossing paradigm. If the boundary of the target is too close, then it may become unclear, resulting in multiple choices or the wrong choice.

Another approach is to increase the size of the target. For example, Vanacken et al.utilized bubble cursor technology for VR interactions[6]. By increasing the activation area of the target, the size of the target was actually increased, making the selection of the target easier. Argelaguet et al.proposed increasing the target size in 3D space to reduce the selection difficulty, but this technology cannot be utilized in a spatial environment in which the relationships between objects are particularly complex[45]. Tu et al. proved through their experiments that crossing and pointing in a VR environment obeyed Fitts' law, so that the location and size of the target directly affect the crossing performance[1]. Therefore, researchers and designers should consider the size and distance of the target when designing an interface, to make the target acquisition easier and faster. When designing crossing-based interactions, the target size can be appropriately increased or the target distance can be reduced to improve the target selection efficiency.

3.2.5 Field of view

The field of view is an important factor affecting a user's immersion in VR. In the real world, the effective horizontal field of view of the human eye is approximately 200° , and the vertical field of view is approximately 130°. However, the field of view for mainstream VR devices is significantly smaller than for human eyes. For example, the Oculus Rift CV1 has a horizontal and vertical fields of view of approximately 80° and 90° , respectively. A narrow field of view has been shown to affect human performance in navigation, manipulation, spatial perception, and visual search tasks, and even to disrupt

the coordination of our eye and head movements[46,47]. Lin et al.[48] studied the effect of the field of view on user performance in a virtual environment in 2002. From data on 10 participants for four fields of view (60°, 100°, 140°, and 180°), they revealed that participants' performances were better for a larger field of view. Xiao et al.[49] increased the horizontal field of view to 190° through a sparse external display, and their research suggested that this approach helped to reduce nausea in patients with motion sickness. Tu et al.[1] considered the horizontal field of view of the Oculus Rift CV1 to be only approximately 80°, and designed angles between two targets and the human eye of 20.4° and 90.2°, respectively. Because only one target appeared in the field of view at 90.2°, the user needed to turn their head when selecting the target, and in the case of 20.4° both targets appeared in the field of view, and the user did not need to turn their head. These studies indicate that the field of view of a VR device plays an important role in the crossing performance. Therefore, when designing a crossing interaction scheme the inherent field of view of the device and the angles between targets should be fully considered.

3.2.6 Occlusion

Occlusion between objects is undoubtedly an important factor in crossing-based selection for VR. When a target is occluded by other objects, this increases the difficulty for users to select the target. In the worst cases, users cannot select the target at all. With the development of VR technology, the relationships between objects in virtual environments will become increasingly complex. Therefore, the selection of occluded targets has become one of the most important aspects in VR. Although the removal of occluded objects by dragging and rotating operations in VR is a common solution[50,51], with the increasing complexity of virtual environments and the increasing numbers of objects in space, this approach cannot effectively solve the problem of target occlusion in a complex environment. Therefore, Elmqvist et al.[52] classified the occlusion positions of objects in complex space environments into four categories:

Proximity: Two objects are close, but not in contact. At this time, occlusion occurs from some viewpoints, but there is no occlusion from other viewpoints.

Intersection: Two objects are partially intersecting, and are occluded from any point of view.

Enclosement: An object is in the interior of another object, such as a room and an object in the room. At this time, the object can be seen from the interior view of the room, but cannot be seen from the outside view of the room.

Containment: An object is completely contained by another object, and there is occlusion from any point of view.

The above situations still occur when users cross targets. How to employ crossing to better solve the occlusion problem constitutes an important question. For proximity and intersection situations, Stoakley et al.[53] proposed a miniature world technique. This provides users with a small window to display the virtual environment. Users can operate on the "mini-world" to complete the selection of occluded objects. If users select small objects in the mini-world using crossing, the selection accuracy is improved. For enclosement and containment situations, Vanacken et al.[6] proposed translucent technology to alter the transparencies of objects to make occluded objects visible. However, the spatial relationships between semi-transparent objects may become unclear to users. If users select unclear objects using crossing instead of pointing, then the selection difficulty is significantly decreased. Therefore, researchers and designers should consider using the crossing paradigm instead of pointing when designing selection techniques for occlusion targets. In summary, it is worth exploring how to select occluded targets using the crossing paradigm in a VR environment.

4 Summary

This study introduces target selection based on the crossing paradigm in VR, summarizes and analyzes the factors affecting crossing in the input and visual spaces, and proposes a detailed classification framework. The aim of our study is to provide a reference for target selection interaction design based on crossing in VR environments.

References

1   Tu H W, Huang S S, Yuan J B, Ren X S, Tian F. Crossing-based selection with virtual reality head-mounted displays. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, New York, NY, USA, ACM, 2019
DOI:10.1145/3290605.3300848

2   Argelaguet F, Andujar C. A survey of 3D object selection techniques for virtual environments. Computers & Graphics, 2013, 37(3): 121−136
DOI:10.1016/j.cag.2012.12.003

3   Grossman T, Balakrishnan R. The design and evaluation of selection techniques for 3D volumetric displays. In: Proceedings of the 19th annual ACM symposium on User interface software and technology. Montreux, Switzerland, ACM, 2006
DOI:10.1145/1166253.1166257

4   Steinicke F, Ropinski T, Hinrichs K. Object selection in virtual environments using an improved virtual pointer metaphor//Computational Imaging and Vision. Dordrecht, Kluwer Academic Publishers, 2006, 320−326
DOI:10.1007/1-4020-4179-9_46

5   Cournia N, Smith J D, Duchowski A T. Gaze- vs. hand-based pointing in virtual environments. In: Human factors in computing systems. Lauderdale, Florida, USA, ACM, 2003
DOI:10.1145/765891.765982

6   Vanacken L, Grossman T, Coninx K. Exploring the effects of environment density and target visibility on object selection in 3D virtual environments. In: 2007 IEEE Symposium on 3D User Interfaces. Charlotte, NC, USA, IEEE, 2007
DOI:10.1109/3dui.2007.340783

7   Poupyrev I, Billinghurst M, Weghorst S, Ichikawa T. The go-go interaction technique. In: Proceedings of the 9th annual ACM symposium on User interface software and technology. Seattle, Washington, USA, ACM,1996
DOI:10.1145/237091.237102

8   Zhai S, Buxton W, Milgram P. The "Silk Cursor": Investigating Transparency for 3D Target Acquisition, 1994
DOI:10.1145/191666.191822

9   Grossman T, Balakrishnan R, Kurtenbach G, Fitzmaurice G, Khan A. Creating principal 3D curves with digital tape drawing. Minneapolis, Minnesota, USA, ACM, 2002
DOI:10.1145/503376.503398

10 Bowman D A, Johnson D B, Hodges L F. Testbed evaluation of virtual environment interaction techniques. Presence: Teleoperators and Virtual Environments, 2001, 10(1): 75−95
DOI: 10.1162/105474601750182333

11 Haan G D, Koutek M, Post F H. IntenSelect: Using Dynamic Object Rating for Assisting 3D Object Selection. In: Workshop on Immersive Projection Technology, 2005
DOI: 10.2312/EGVE/IPT_EGVE2005/201-209

12 Pierce J S, Forsberg A S, Conway M J, Hong S, Zeleznik R C, Mine M R. Image plane interaction techniques in 3D immersive environments. In: Proceedings of the 1997 symposium on Interactive 3D graphics. Providence, Rhode Island, USA, ACM, 1997
DOI:10.1145/253284.253303

13 Forsberg A, Herndon K, Zeleznik R. Aperture based selection for immersive virtual environments. In: Proceedings of the 9th annual ACM symposium on User interface software and technology. Seattle, Washington, USA, ACM, 1996

DOI:10.1145/237091.237105

14 Accot J, Zhai S M. More than dotting the i's: - foundations for crossing-based interfaces. In: Proceedings of the SIGCHI conference on Human factors in computing systems Changing our world, changing ourselves. Minneapolis, Minnesota, USA, ACM, 2002
DOI:10.1145/503376.503390

15 Fitts P M. The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology: General, 1992, 121(3): 262—269
DOI:10.1037//0096-3445.121.3.262

16 Accot J, Zhai S M. Beyond fitts' law. In: Proceedings of the SIGCHI conference on Human factors in computing systems. Atlanta, Georgia, USA, ACM, 1997
DOI:10.1145/258549.258760

17 Forlines C, Balakrishnan R. Evaluating tactile feedback and direct vs. indirect stylus input in pointing and crossing selection tasks. In: Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems. Florence, Italy, ACM, 2008
DOI:10.1145/1357054.1357299

18 Apitz G, Guimbretière F. CrossY. In: Proceedings of the 17th annual ACM symposium on User interface software and technology. Santa Fe, NM, USA, ACM, 2004
DOI:10.1145/1029632.1029635

19 Dixon M, Guimbretière F, Chen N. Optimal parameters for efficient crossing-based dialog boxes. In: Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems. Florence, Italy, ACM, 2008
DOI:10.1145/1357054.1357307

20 Wobbrock J O, Gajos K Z. Goal crossing with mice and trackballs for people with motor impairments. ACM Transactions on Accessible Computing, 2008, 1(1): 1—37
DOI:10.1145/1361203.1361207

21 Dragicevic P. Combining crossing-based and paper-based interaction paradigms for dragging and dropping between overlapping windows. In: Proceedings of the 17th annual ACM symposium on User interface software and technology. Santa Fe, NM, USA, ACM, 2004
DOI:10.1145/1029632.1029667

22 Sulaiman A N, Olivier P. Attribute Gates. In: Proceedings of the 21st annual ACM symposium on User interface software and technology. Monterey, CA, USA, ACM, 2008
DOI:10.1145/1449715.1449726

23 Yoshikawa T, Shizuki B, Tanaka J. HandyWidgets. In: Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces. Cambridge, Massachusetts, USA, ACM, 2012
DOI:10.1145/2396636.2396667

24 Perin C, Dragicevic P. Manipulating multiple sliders by crossing. In: Proceedings of the 26th Conference on interaction Homme-Machine. 2014
DOI:10.1145/2670444.2670449

25 Luo Y X, Vogel D. Crossing-based selection with direct touch input. In: Proceedings of the 32nd annual ACM conference on Human factors in computing systems. Toronto, Ontario, Canada, ACM, 2014
DOI:10.1145/2556288.2557397

26 Luo Y X, Vogel D. Pin-and-cross. In: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology.
DOI:10.1145/2807442.2807444

27 Lv Z. Towards the design of effective freehand gestural interaction for interactive TV. Journal of Intelligent & Fuzzy Systems, 2016, 31(5): 2659—2674
DOI:10.3233/jifs-169106

28 Vogel D, Balakrishnan R. Distant freehand pointing and clicking on very large, high resolution displays. In: Proceedings of the 18th annual ACM symposium on User interface software and technology. Seattle, WA, USA, ACM, 2005

DOI:10.1145/1095034.1095041

29 Andujar C, Argelaguet F. Anisomorphic ray-casting manipulation for interacting with 2D GUIs. Computers & Graphics, 2007, 31(1): 15–25
    DOI:10.1016/j.cag.2006.09.003

30 Frees S, Kessler G D. Precise and rapid interaction through scaled manipulation in immersive virtual environments. In: IEEE Proceedings. Virtual Reality, 2005. Bonn, Germany, IEEE, 2005
    DOI:10.1109/vr.2005.1492759

31 Stuerzlinger, W. Considerations for Targets in 3D Pointing Experiments. In: The Workshop on Interactive Surfaces for Interaction with Stereoscopic3d. 2014

32 Das K, Borst C W. An evaluation of menu properties and pointing techniques in a projection-based VR environment. In: 2010 IEEE Symposium on 3D User Interfaces (3DUI). Waltham, MA, USA, IEEE, 2010
    DOI:10.1109/3dui.2010.5444721

33 Dachselt R, Hübner A. Three-dimensional menus: A survey and taxonomy. Computers & Graphics, 2007, 31(1): 53–65
    DOI:10.1016/j.cag.2006.09.006

34 Bowman D A, Wingrave C A. Design and evaluation of menu systems for immersive virtual environments. In: Proceedings IEEE Virtual Reality, 2001
    DOI:10.1109/vr.2001.913781

35 Mine M R. Virtual environment interaction techniques, 1995

36 Jacoby R H, Ellis S R. Using virtual menus in a virtual environment. In: Visual Data Interpretation. International Society for Optics and Photonics. 1992

37 Liang J, Green M. JDCAD: A highly interactive 3D modeling system. In: International Conference on Cad and Computer Graphics. Beijing, China, 1994

38 Ramcharitar A A, Teather R J R. EZCursor VR: 2D selection with virtual reality head-mounted displays, 2018

39 Teather R J, Stuerzlinger W. Pointing at 3d target projections with one-eyed and stereo cursors. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Paris, France, 2013
    DOI:10.1145/2470654.2470677

40 Kelly J W, Cherep L A, Klesel B, Siegel Z D, George S. Comparison of two methods for improving distance perception in virtual reality. ACM Transactions on Applied Perception, 2018, 15(2): 1–11
    DOI:10.1145/3165285

41 Armbrüster C, Wolter M, Kuhlen T, Spijkers W, Fimm B. Depth perception in virtual reality: distance estimations in peri- and extrapersonal space. CyberPsychology & Behavior, 2008, 11(1): 9–15
    DOI:10.1089/cpb.2007.9935

42 Renner R S, Velichkovsky B M, Helmert J R. The perception of egocentric distances in virtual environments—A review. ACM Computing Surveys, 2013, 46(2): 1–40
    DOI:10.1145/2543581.2543590

43 Naceri A, Chellali R. Depth perception within peripersonal space using head-mounted display. Presence: Teleoperators and Virtual Environments, 2011, 20(3): 254–272
    DOI:10.1162/pres_a_00048

44 Dachselt R, Hübner A. A survey and taxonomy of 3D menu techniques. In: Proceedings of the 12th Eurographics conference on Virtual Environments. Lisbon, Portugal: Eurographics Association, 2006, 89–99

45 Argelaguet F, Andujar C. Improving 3D selection in VEs through expanding targets and forced disocclusion. Smart Graphics. Berlin, Heidelberg: Springer Berlin Heidelberg, 45–57
    DOI:10.1007/978-3-540-85412-8_5

46 Alfano P L, Michel G F. Restricting the field of view: perceptual and performance effects. Perceptual and Motor Skills, 1990, 70(1): 35–45
    DOI:10.2466/pms.1990.70.1.35

47 Jeannerod M. Living in a world transformed. Perceptual and performatory adaptation to visual distortion. Neuropsychologia, 1983, 21(2): 184–185

DOI:10.1016/0028-3932(83)90090-8

48　Lin J J W, Duh H B L, Parker D E, Abi-Rached H, Furness T A. Effects of field of view on presence, enjoyment, memory, and simulator sickness in a virtual environment. In: Proceedings IEEE Virtual Reality 2002. Orlando, FL, USA, IEEE, 2002
DOI:10.1109/vr.2002.996519

49　Xiao R, Benko H. Augmenting the field-of-view of head-mounted displays with sparse peripheral displays. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, Lisbon, Portugal, ACM, 2016
DOI:10.1145/2858036.2858212

50　Gabbard J L. A taxonomy of usability characteristics in virtual environments. Virginia Tech, 1997

51　Hinckley K, Pausch R, Goble J C, Kassell N F. A survey of design issues in spatial input. In: Proceedings of the 7th annual ACM symposium on User interface software and technology. Marina del Rey, California, USA, ACM, 1994
DOI:10.1145/192426.192501

52　Elmqvist N, Tsigas P. A taxonomy of 3D occlusion management for visualization. IEEE Transactions on Visualization and Computer Graphics, 2008, 14(5): 1095−1109
DOI:10.1109/tvcg.2008.59

53　Stoakley R, Conway M J, Pausch R. Virtual reality on a WIM. In: Proceedings of the SIGCHI conference on Human factors in computing systems. Denver, Colorado, USA, ACM, 1995
DOI:10.1145/223904.223938