Portal Selection

Michael Hanlon[*]
Cornell Tech

Gilberto E. Ruiz[†]
Cornell Tech

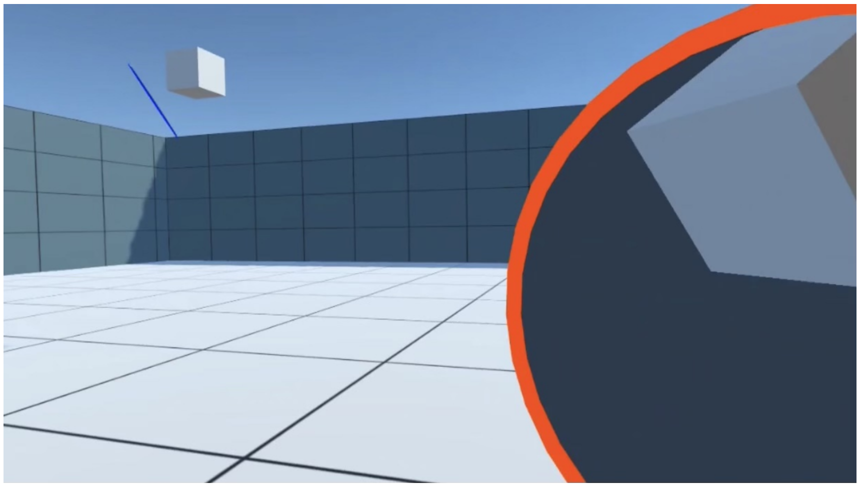

Figure 1: Screenshot of the Portal Selection technique in use.

Abstract

The Portal Selection Technique presents a novel VR methodology for manipulating portals in a virtual environment. Utilizing hand gestures, users can activate, position, and orient portals seamlessly within their virtual surroundings. This technique combines the precision manipulation of grasping techniques with the long distance selection of ray casting techniques.

Index Terms: Virtual Reality, Gesture Control, Spatial Interaction, Portal Mechanics, Hand Gesture Recognition, Immersive Experience.

1 Introduction

Portal Selection is a VR interaction technique inspired by games like *Portal* and *Portal 2,* enabling users to interact with objects through two portals activated by hand gestures. Users can shoot portals with their left hand and manipulate objects through them with their right hand. By making use of ray casting with their left hand, users can place a blue portal, which will automatically connect to the orange portal near their side. Objects can be brought through the portals by simply reaching through the portal and grabbing with their right hand.

To enhance the viability of Portal Selection as a selection technique, the ability to fix the length of the ray cast and place portals midair has also been implemented. This enables users to select and manipulate objects that would otherwise be too far from walls or the floor to be reached with standard portal placement.

2 Related Work

The study conducted by Ablett et al. [1] delves into efficient portal rendering algorithms and user interactions within VR environments. Their research, aimed at optimizing portal performance on mobile VR devices and exploring innovative portal manipulation techniques, closely aligns with the objectives of our Portal Selection project. By leveraging their insights on portal creation interactions and manipulation, we aim to enhance the intuitiveness and efficiency of our system, facilitating seamless user experiences in interacting with distant objects through portals.

In another relevant paper, Bowman and Hodges [2] discuss various selection and manipulation techniques, including the Go-Go technique, arm-extension techniques, and ray casting. After examining their strengths and limitations, the authors introduce the HOMER technique, which involves a hybrid approach of ray casting and grasping for object selection and manipulation. This paper provided valuable insights into the advantages and disadvantages of our technique compared to approaches like HOMER. While our technique allows for quick retrieval of distant objects by pulling them through portals and offers precise grasping, it may be limited by the proximity of surfaces to the objects being manipulated. Because of this, we are investigating ways of placing portals midair.

Finally, many of our design choices, such as the color of the primary and secondary portals, were inspired by the popular video games *Portal* and *Portal 2* developed by Valve.

3 Technique

3.1 Design

The ”Portal Selection” virtual environment is set within a predefined play area from the XRC Core package, featuring four interactables at various heights and distances to explore the system’s capabilities. Users navigate and interact within this space using a ray casting system that originates from their hands. This system adjusts the ray’s length and direction based on hand orientation and position, offering visual feedback on potential portal locations through color-coded line renderers—blue for valid surfaces and red for invalid ones.

Users control the system with intuitive gestures: aiming with the left hand and pinching to place a portal at the targeted location. Subsequent adjustments are made by pinching with the right hand

[*] e-mail: mph99@cornell.edu
[†] e-mail: ger83@cornell.edu

to rotate and finalize the portal's orientation. Visual cues further enhance usability; the line renderer's color shift and a portal preview at the ray's endpoint assist in precise placement, while yellow highlights on interactable objects indicate their availability for manipulation, ensuring a seamless interaction within the virtual space.

Portal Selection was designed for the primary task parameters of targets being single, unoccluded objects, placed at a distance well out of reach of the user with the goal of retrieving these objects by placing a portal and grasping them. Additionally, we placed several objects in our sample scene far away from the floor or any walls to necessitate the use of midair portal placement. The primary tasks involved enabling users to activate, position, and rotate portals using intuitive hand gestures, specifically through pinching motions with either hand. The system was required to accurately detect these gestures in a dynamic virtual environment, allowing for real-time interaction with objects and portal surfaces. Additionally, the task parameters included the development of a visual feedback system that clearly communicates viable portal placements through color-coded cues and a preview feature, enhancing the user's spatial understanding and decision-making process in the virtual space. These parameters were crucial for maintaining an immersive experience while providing precise control over portal mechanics.

3.1.1 Mathematical Model

Notation Summary:

$\mathbf{P}_{rh}, \mathbf{R}_{rh}$: Position and rotation of the real hand.
$\mathbf{P}_{p1}, \mathbf{R}_{p1}$: Position and rotation of the first (sending) portal.
$\mathbf{P}_{p2}, \mathbf{R}_{p2}$: Position and rotation of the second (receiving) portal.
$\mathbf{P}_{vh}, \mathbf{R}_{vh}$: Final position and rotation of the virtual hand.

1. Calculate Relative Position and Orientation: Calculate the relative position and orientation of the real hand to the first portal. This step transforms the real hand's state into the local space of the first portal.

$$\mathbf{P}_{relative} = \mathbf{R}_{p1}^{-1} \cdot (\mathbf{P}_{rh} - \mathbf{P}_{p1})$$

$$\mathbf{R}_{relative} = \mathbf{R}_{p1}^{-1} \cdot \mathbf{R}_{rh}$$

2. Apply Portal Transformation: Apply the transformation through the portal, considering the orientation and position change from the first to the second portal. This step effectively ”teleports” the relative position and orientation from the context of the first portal to that of the second.

$$\mathbf{P}_{vh} = \mathbf{P}_{p2} + \mathbf{R}_{p2} \cdot \mathbf{P}_{relative}$$

$$\mathbf{R}_{vh} = \mathbf{R}_{p2} \cdot \mathbf{R}_{relative}$$

3.2 Implementation

When implementing the scripts for the Portal Selection technique, we split our scripts into two primary groups: scripts for the implementation of portal placement, and scripts for the implementation of the portals themselves.

3.2.1 Portal Placement Implementation

For the management of portal placement and interaction, we implemented a feedback script that determines viable surfaces for portal placement by using ray casts that extend from the user's hand into the virtual environment. Each ray cast dynamically adjusts in length and direction based on the hand's movement, rendering a visual line that changes color based on surface validity as mentioned before. The script also displays the preview of the portal at the potential placement location.

In addition to the feedback script, the logic script handles the interactive aspects of portal management. It uses pinch gestures detected via Unity's XR Interaction Toolkit for engagement. With a pinch from the left hand, users activate and place portals at the designated locations, confirmed by the feedback system. A pinch with the right hand enables users to adjust the portal's orientation dynamically. This adjustment is controlled by storing the initial orientation of the hand at the start of the interaction, which is then compared to subsequent movements to calculate and apply rotational adjustments to the portal.

3.2.2 Portal Implementation

For the management of portals, we started by implementing a feedback script to render the portal surfaces. This was done by moving a camera for each of the portals in the script, and then rendering each camera's view to its respective portal material. Next, we began work on the functionality of portals to warp objects from one portal to the other upon collision. This was done through the main portal logic script, which monitors the objects currently colliding with a portal and warps them to the other portal upon crossing the midway point of the colliding portal.

After this core functionality was established, we further enhanced the feedback of the portals. Firstly, we created a new shader to cut off the portions of objects passing through the portal that would otherwise poke through the back of the portal. Secondly, we updated the feedback script to instantiate clones of interactables that collide with portals during runtime, and position them at the other portal so that the interactables can be seen on both sides of the portal as they pass through.

4 Conclusion

In conclusion, the Portal Selection Technique provides a straightforward and effective way to interact with virtual environments using hand gestures. Users can place and adjust portals quickly, which makes viewing and interacting with objects at a far distance intuitive for the user. This project shows the potential for simple, gesture-based controls to enhance user experience in virtual settings, and introduces a unique take on the hybrid combination of ray casting and grasping for interaction.

4.1 Future Work

For future improvements to the portal selection technique, there are two main changes that we would aim to implement. Firstly, we would like to switch the technique from a hand-tracking implementation to a controller-based one. This is mainly due to the trouble we experienced with the tracking of the pinching input action when placing and manipulating portals, as the tracking would often result in false positives for pinching inputs. Additionally, we would like to further investigate improvements to the shader used for portal surfaces. Although we were able to solve the primary issue of stereo rendering, the current portal surface shader implementation has issues with slight distortions around objects viewed through the portal.

References

[1] D. Ablett, A. Cunningham, G. A. Lee, and B. H. Thomas. Portal rendering and creation interactions in virtual reality. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 160–168, 2022. doi: 10.1109/ISMAR55827.2022.00030 1

[2] D. A. Bowman and L. F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, pp. 35–38, 1997. doi: 10.1145/253284.253301 1