EFFECTS OF CONTROL-DISPLAY MAPPING ON 3D INTERACTION IN IMMERSIVE VIRTUAL ENVIRONMENTS

by

Jialei Li

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2017

Approved by:

____
Dr. Zachary Wartell

____
Dr. William Ribarsky

____
Dr. Aidong Lu

____
Dr. Paula Goolkasian

____
Dr. Isaac Cho

ABSTRACT

JIALEI LI. Effects of control-display mapping on 3D interaction in immersive virtual environments. (Under the direction of DR. ZACHARY J. WARTELL)

There is a significant amount of research on the mapping of the interactive control space to the display space in 3D user interfaces. While most work has focused on control-display gain (or ratio), one factor that has received relatively less attention is cursor offset, which is a vector between the input device in physical space and the virtual cursor in display space. Empirical results of the efficiency and usability of a translational offset have been provided by multiple studies, but only for a particular type of interaction, on a specific display system. Anecdotal evidence suggests that results may differ on different types of tasks or in different types of VR systems. Therefore, this research focuses on designing and evaluating virtual cursor offset techniques for 3D interaction in immersive virtual environments in a more comprehensive manner.

Three user studies are carried out to explore the effect of various offset techniques on a 7 degree-of-freedom navigation task in a surround-screen CAVE system, for both one-handed and two-handed interactions. Results show that the Linear Offset technique outperforms other offset techniques for exocentric travel tasks and provides superior performance under a variety of conditions.

To compare the same offset techniques in an HMD environment, an evaluation on 3D object selection tasks that follows Fitts' law model under ISO 9241-9 standard using two different input devices is first presented. The result indicates that direct selection of nearby objects with No Offset remains the most efficient and the Linear Offset technique could enhance selection performance with objects at a distance. Further, two experiments on unimanual and bimanual object manipulation are conducted respectively, using the same input devices. For both studies, No Offset does not reveal advantages over Linear Offset or Go-Go Offset when the object is within reach, while for out of reach condition, Linear Offset outperforms Fixed-Length and Go-Go Offset. In all three studies conducted within Oculus Rift DK2, Razer Hydra proves to be more stable and effective than Leap Motion.

DEDICATION

This dissertation is dedicated to Mom and Dad.

ACKNOWLEDGMENTS

First and foremost, I want to thank my parents, for their unconditional love and endless support. Thank you for always believing in me and supporting me throughout my entire education.

I would like to express my deepest thanks to my academic advisor, Dr. Zachary Wartell, for providing me with the opportunity to pursue a PhD and for his advice and guidance during my research.

I deeply appreciate the valuable suggestions from members of my dissertation committee, Dr. William Ribarsky, Dr. Aidong Lu, Dr. Paula Goolkasian and Dr. Isaac Cho. In particular, I thank Dr. Isaac Cho for his help and encouragement during my PhD study in the Charlotte Visualization Center.

Finally, I would like to thank all the people who have participated in my user studies and contributed to this dissertation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

3DUI	3D User Interface
ANOVA	Analysis of Variance
CAVE	Cave Automatic Virtual Environment
DOF	Degree of Freedom
FOR	Field of Regard
FOV	Field of View
HCI	Human Computer Interaction
HMD	Head-Mounted Display
HTD	Head-Tracked Display
IVE	Immersive Virtual Environment
LSD	Least Significant Difference
MSVE	Multi-Scale Virtual Environment
S+W	Spindle+Wheel
VE	Virtual Environment
VR	Virtual Reality

CHAPTER 1: INTRODUCTION

1.1 Immersive Virtual Environments

Howard Rheingold [108] defined virtual reality (VR) as an artificial environment within which the user is "surrounded by a three-dimensional computer-generated representation, and is able to move around in the virtual world and see it from different angles, to reach into it, grab it and reshape it". A virtual environment (VE) is immersive in the sense described by Ellis [34]: consisting of content (objects and actors), geometry and dynamics, with an egocentric frame reference, including perception of objects in depth, and giving rise to the normal ocular, auditory, vestibular and other sensory cues and consequences. Therefore, immersion of the VR systems depends on the extent to which computer displays accommodate sensory systems. They are surrounding to the extent that information can arrive at the person's sense organs from any (virtual) direction and the extent to which the individual can turn toward any direction and yet remain in the environment [119]. Whether or not a system can be classified as immersive depends crucially on the hardware, software and peripherals (displays and body sensors) of that system.

The first immersive VE display was a Head-Mounted Display (HMD) that was built in 1968 by Ivan Sutherland [128]. It consisted of two cathode ray tubes (CRTs) next to the user's ears and tracking hardware to measure head orientation. Two mirrors were placed in front of the user's eyes to reflect the image from the CRT displays. However, since it had no input device other than a keyboard, it did not allow actions controlled by manual gestures. Current HMDs resemble a helmet with two small displays attached and are often more light-weight. The two displays provide stereoscopic viewing by presenting separate images to each eye. Some HMDs are also equipped with stereo headphones to provide users with audio feedback.

Besides HMDs, various VE displays have emerged, ranging from desktop monitors to surround-screen displays like the CAVE [31]. The CAVE is an immersive VR system that typically consists of multiple screens, including two or more walls and a floor. The wall screens are generally rear projected and the floor screen can be projected from below or from the top. While using the CAVE, users are normally required to wear stereo glasses (either shuttered or polarized), which are synchronized with the projectors to enable stereoscopic viewing.

Virtual environments give users the illusion of being inside a computer simulated 3D world. To enable interaction with the surrounding virtual world, VEs require certain hardware components. Current advancement in tracking technology is able to provide 6 degree-of-freedom (DOF) tracking, which measures the position and orientation of the user's hand and head. The manual input devices allow users to use hand gestures to change the state of the virtual world.

Input devices for human-computer interaction (HCI) have evolved over decades and changed across many forms. Hinckley et al. [50] define spatial input as interfaces based upon free-space 3D input technologies such as electromagnetic or camera-based trackers, as opposed to desktop devices such as the mouse. The manual input may be a hand-held device with push buttons on it, an instrumented glove or just the bare hand. In each case, the position and orientation of the hand must be measured by the tracker to enable manual control of actions. Therefore, input devices possess numerous properties and parameters which can enhance or limit user performance [76].

In particular, touchless gestural controls are able to offer benefits to certain types of user interfaces (UIs) or to augmented systems with multiple modalities of user input, without acquiring a physical input device. Touchless gestural controls track the user's movements in physical space, interpret these motions and then map them to the corresponding surrogate in the virtual scene. Since recent computer vision techniques are able to provide robust real-time tracking of bare hand postures and movements in 3D space, some researchers are beginning to develop and evaluate new interaction techniques using commercial motion tracking systems, either with or without reflective trackers on the user's hand. Figure 1.1 shows hand and finger tracking with a commercially available motion capture device.

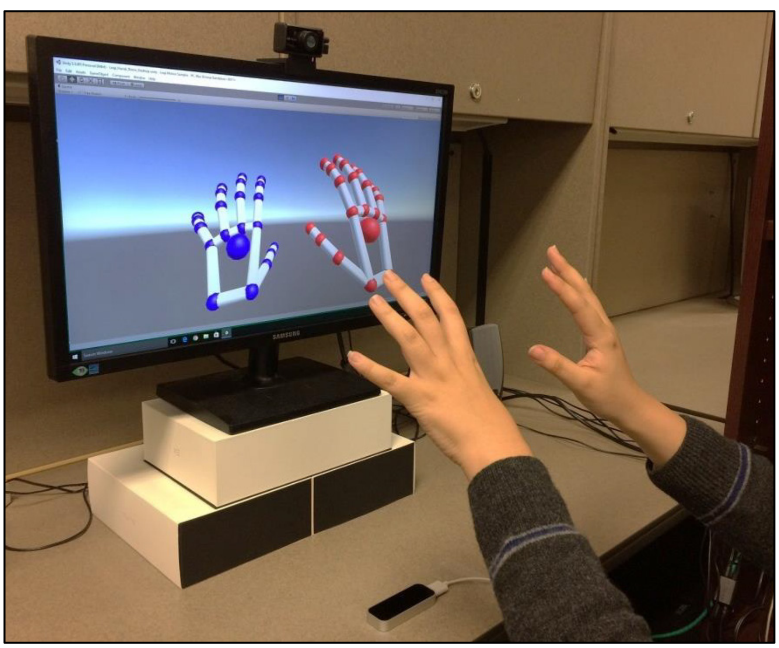


Figure 1.1: Hand and finger tracking using Leap Motion controller.

1.2 3D Interaction

The most inherent and appealing aspect of immersive virtual environments is their ability to allow users to interact with a simulated environment in almost the same way as they do in the real world [37]. 3D interaction refers to the actions of using gestures in free 3D space. Similar to interaction in the physical world, 3D interaction within the VE is affected by user characteristics, task conditions and properties of the environment [124]. In addition, user performance is strongly affected by the characteristics of the interface components, which are the properties of input and output devices and interaction techniques [124] [76]. The design of 3D mappings of interaction techniques, which translate user-operated device motions into object movements in virtual environments, is certainly one of the core issues in designing 3D user interfaces.

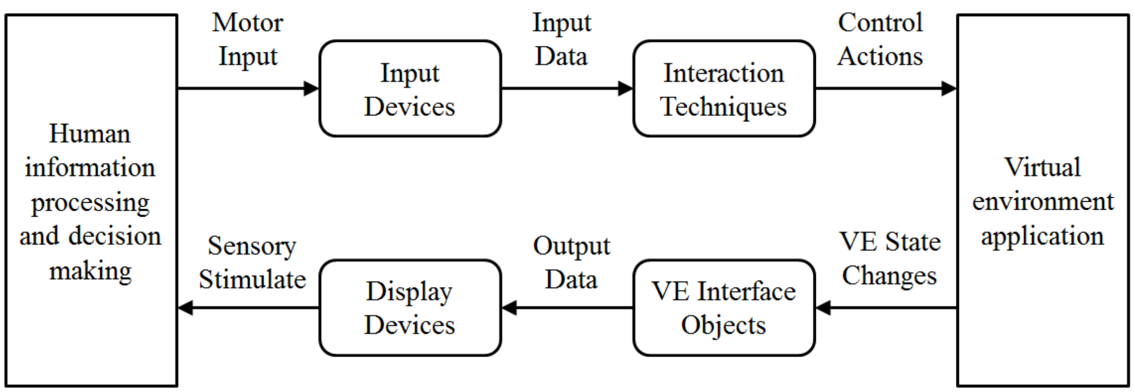


Figure 1.2: Interaction in virtual environments [103].

Poupyrev and Ichikawa [103] placed the major components of VE interfaces into perspective by extending the classic human factor model of human-machine interfaces introduced by Taylor [130] and adopted later for human-computer interaction [76] [65], as shown in Figure 1.2 [103]. The user interacts with VE applications in a closed-loop system by applying motor stimuli to input devices and receiving sensory feedback from display devices. The interaction techniques maps user input captured by input devices, into the control actions. The VE application responds by changing the state of the VE. Display devices provide sensory feedback to users by stimulating their visual, auditory and other perceptual systems, thus closing the interaction loop [103].

A key problem in spatial interaction is identifying aspects of the proprioceptive senses that we can take advantage of when interacting in real space [50]. VR systems require that normal proprioceptive information we use unconsciously to form a mental model of the body be overlaid with sensory data that is supplied by the VE [119]. The consistency between proprioceptive information and sensory feedback, as a result, becomes one of the fundamental requirements for an effective VE. With the help of tracking devices, human body movements are mapped onto corresponding movements of the user's avatar in the virtual world. Direct user interaction depends on natural intuitive mapping between user action and the resulting action in the virtual world, which requires no special action on the user end and provides proprioceptive information that can help the user maintain a better mental model of the current virtual world [87].

Traditional isomorphic techniques keep 3D interaction as a direct physical correlation or one-to-one mapping of motion, which restricts the user's movement within arm's reach. However, human can overcome many limitations of the real world through the use of novel interaction techniques in the virtual environment. For example, an active area of research concerning object manipulation has focused on selecting and interacting with objects at a distance, beyond arm's reach. While a common technique being ray-casting [87], other more powerful and productive techniques have been developed, such as World-in-Miniature [126], Go-Go [102], HOMER [19], image plane interaction [99] and scaled-world grab [88], allowing users to interact directly with distant objects.

1.3 Control-Display Mapping

Along with the increased sense of presence, direct interaction and natural viewing of the virtual world have the potential to increase productivity in 3D interaction tasks. One of the obstacles to realizing this potential is the kinematic design of speed-accuracy balance of interaction techniques, which greatly limits the complexity of the VE with which users can effectively and intuitively interact. Implementing a direct manipulation interface that allows for precise positioning and fine-grain adjustments is a challenging design problem. According to Hinckley et al. [50], such an interface must "effectively integrate rapid, imprecise, multiple degree-of-freedom object placement with slower, but more precise object placement, while providing feedback that makes it all comprehensible".

However, there is a trade-off [154] [78] between speed and accuracy in human motor capacity: a person can complete a task quickly but with low accuracy; or slow down to achieve high accuracy. This trade-off is captured by Fitts' law [35], which is a highly verified model for describing the relationship between movement time, distance and accuracy in selection tasks and has been widely used to design and evaluate interaction techniques and input devices in the HCI community. Originally used to model direct pointing where hand taps physical objects, Fitts' law is also robust for indirect pointing when the control device and display pointer are decoupled [75]. The decoupling of control and display creates two distinct spaces: the motor space and the display space.

There have been numerous studies on different aspects of 3D interfaces, exploring perceptual support of 3D displays [17] [13] [58] [91] [33] [39], evaluating the quality and usefulness of 3D input devices [76] [157] [156] [83] [9], and investigating the effectiveness of 3D interaction techniques [88] [102] [101] [153] [37]. One aspect that has received less research attention is the mapping of the interactive control space to the display space. To explore the effectiveness of 3D interaction, the research presented in this thesis considers mapping from the user's motor space in the real 3D world (*control space*) onto the virtual 3D *display space*.

The study of the relationship between control space and display space originates early in the last century, with the introduction of technology that separated the action from its effect. Control-display mapping was mostly discussed with two components: gain and order. The gain factor is the ratio that determines how much motion in control will be transferred to the effect or motion in the display while the order factor is about whether the control affects the position, velocity or acceleration of the motion in display.

Control to display mapping thus far has mostly been studied in terms of control-display (CD) gain, also referred to as control-display ratio [40] [81] [27] [14]. CD gain is a unit free scale coefficient that maps the motion of the control device to the motion of the display cursor. Since it is generally difficult to simultaneously optimize both speed and accuracy of movements, CD gain is often adapted to facilitate the interaction task. For example, pointer acceleration is the default behavior in common operating systems. It dynamically changes the CD gain between the input device (mouse) and the display pointer (cursor) as a function of the device velocity, based on the assumption that fast mouse movement implies a long distance to the intended target, so the cursor movement should be amplified to quickly cover that distance. Conversely, slow mouse movement indicates that the target is nearby, so the cursor should slow down accordingly for accurate adjustments.

Beyond manipulation of scale, there has been little exploration of the influence of other factors in control-display mapping, such as cursor offset, on 3D interaction. Adding a positional offset to the 3D cursor indicates that there is an indirection between the input device in physical space and the visual pointer in display space, and often enhances efficiency and usability of 3D user interfaces. Empirical evidence of the effects of cursor offset has been provided by several studies [88] [70], but with contradicting conclusions. Additionally, previous research only attempted to evaluate cursor offset techniques for specific types of interaction, mainly object selection and manipulation, on specific display systems, mainly the HMD, using hand-held input devices. Anecdotal evidence suggests that the results may differ in other types of VR systems. For example, for touchless gesture input, the optimal transfer function may be significantly different from those for hand-held input devices. The design of the transfer function is inherently linked to biomechanical theories, since any motion of the fingers, hands and arms that is required by the system could have significant implications for the design of the 3D user interfaces. Therefore, this research focuses on designing and evaluating virtual cursor offset techniques for 3D interaction in immersive virtual environments in a more comprehensive manner.

1.4 Research Problem

While 3D interaction has become a well-established research area with continuing improvements, interacting in 3D space can sometimes still be difficult and the design of effective 3D user interfaces is still challenging. In summary, the following aspects of this research problem have been discussed:

- 3D interaction in virtual environments can be affected by different input devices and display systems.
- Control-Display mappings of interaction techniques still need to be investigated, especially in terms of virtual cursor offset.
- Different cursor offset techniques should be designed and evaluated for different types of interaction in different VR systems.

1.5 Expected Contributions

The purpose of the research presented in this dissertation is to investigate the influence of different virtual cursor offset techniques on 3D interactions across different VR systems. In particular, this dissertation will:

- Develop a new virtual cursor offset technique that will benefit user performance in 3D interaction.
- Evaluate different cursor offset techniques on navigation tasks in a multi-scale virtual environment (MSVE) for both unimanual and bimanual interactions using a CAVE system.
- Evaluate cursor offset techniques for object selection and manipulation in a HMD system using both a hand-held device and a motion capture controller.
- Provide user interface design guidelines based on the implication of the evaluation results.

1.6 Dissertation Overview

This dissertation is organized into eight chapters as follows:

Chapter 2 surveys related work in the area of 3D displays, interaction techniques and control-display mapping.

Chapter 3 describes cursor offset techniques and elaborates on their differences.

Chapter 4 presents two user studies that evaluate the effects of varying offset techniques on one-handed 7DOF navigation tasks in a CAVE system.

Chapter 5 discusses the evaluation of offset techniques on bimanual 7DOF navigation in a multi-scale virtual environment.

Chapter 6 evaluates offset techniques on object selection following Fitts' model under ISO 9241-9 standard using two different input devices in an HMD environment.

Chapter 7 presents two experiments to investigate the effects of offset techniques on 7DOF object manipulation in an HMD system using one-handed and two-handed interfaces respectively.

Chapter 8 concludes this dissertation and proposes new directions for future work.

CHAPTER 2: BACKGROUND AND LITERATURE REVIEW

Underlying all current 3D interaction research is the mapping of control space to display space. We are largely limited in the types of interactions possible by the hardware devices available. This chapter describes the state-of-the-art display and input devices, reviews some of the common 3D interaction techniques, specifically those for selection and navigation, and discusses some literature that has studied the effects of control-display mapping on 3D interaction.

2.1 Immersive VE Display

To date, 3D user interface research, particularly in virtual reality, has a large focus on immersive stereoscopic displays. The ability to interact with what is displayed is as important as the display itself. To discuss the control-display mapping, it is necessary to understand the implications of the hardware device on both the display and control sides. Two VE displays that will be used in our user studies are described in this section.

2.1.1 Head-Mounted Displays

Head-mounted displays (HMDs) are the most common head-coupled display devices used for VE applications. A typical HMD consists of a helmet with two small displays and an adjustable lens system, which can produce stereoscopic viewing by presenting separate overlapping images to each eye. Interaction with the VE is achieved through specialized input devices, such as wands, data gloves or motion capture cameras. The HMD and input devices are tracked in real-time to update the position and orientation of the user's head and hands within the VE. Early HMDs were worn on the head and completely blocked out light from the physical world, in order to provide the user with more immersion within the virtual world. Modern HMDs have become quite light-weight, and some even offer see-through options that make it possible to augment the physical world and create mobile applications.

The biggest advantage of HMDs over other VE displays is that users can have complete physical immersion allowing for a 360° field-of-regard (FOR) [21]. Another advantage is HMDs are much lighter and portable than most projection-based displays. However, sometimes HMDs can still be heavy to wear and cause a fair amount of fatigue if used for an extended period of time. In addition, HMDs also suffer from a limited field-of-view (FOV) which may produce distortions in the perception of size and distance, increasing the need for head movement, and thus further fatigue. Another disadvantage that plagues stereoscopic HMDs is that they often do not provide a way to accommodate different interocular distances of users [21].

2.1.2 Surround-Screen Displays

Surround-screen displays are VE output devices that consist of three or more screens that surround the user and completely immerse them in the VE. The screens are typically 8 to 12 feet in width and height, and rear-projected to eliminate the possibility of the user casting shadows on the screens. The first surround-screen system was called the CAVE [31] and consisted of three walls and a floor. It was introduced in the early 1990s as an alternative to the HMD. To make the experience more immersive, the user wears stereo glasses (shuttered or polarized) with small tracking device and navigates in the virtual scene by physically walking within the enclosed area of the screens.

The advantage of using surround-screen displays like the CAVE is that they provide a large FOV, which permits users to utilize their peripheral vision [21]. These displays also provide motion parallax cues and stereopsis when the user is tracked and wearing stereo glasses. On the other hand, surround-screen displays are expensive and the installation of such a system often requires a large amount of fixed space. Although the size of the space would seem to indicate that multiple people can collaboratively interact with the system, the motion parallax through head tracking effectively means that the display system is mostly a single user setup. Additional users will only see the 3D images from the tracked user's perspective and there will be no response when the untracked user moves. However, a new projection technology [64] has been developed to enable multi-user tracking in a stereoscopic display system with each user perceiving correct perspective views of the VE individually, which makes collaboration in the surround-screen display systems possible.

2.2 3D Interaction Techniques

Research of 3D spatial input was initially concerned with the evaluation of input devices for spatial manipulation tasks, starting with the pioneering work by Ware [142]. The usability of 3D input devices and the effect of their properties on user performance were then investigated in various manipulation tasks [16] [156] [157]. For example, Zhai and Milgram [155] compared isometric and isotonic devices for different conditions of spatial interaction. Meanwhile, other studies focused on the influence of output display device characteristics on user performance [144] [141].

While hardware devices have been subject to careful and thorough evaluation, the effect of mappings, i.e. interaction techniques, on manipulation performance also attracts some of the earliest work's attention [50] [87] [19]. They summarized existing techniques, identified problems and discussed possible solutions, with the attempt to systematically evaluate 3D interaction techniques as a distinctive component of VE interfaces. Last decade witnessed enormous improvements in spatial input devices and motion tracking systems. These advances have motivated the development of a plethora of interaction techniques relying on 6DOF input devices and user gestures. To provide a general overview for the context and scope of our work, this section discusses 3D interaction techniques relevant to this research.

2.2.1 Object Selection

Object selection is one of the most fundamental tasks in 3D VR applications. In immersive VEs, object selection can occur either as the sole activity that captures the whole attention of the user or as a component of a more complex task sequence. Recent availability of tracking devices that work in 3D space has enabled the development of direct manipulation interfaces. As a consequence, a significant amount of research that provides effective means for selecting virtual objects has been carried out.

There are two basic metaphors for object selection in virtual environments: virtual hand and virtual pointing [104] [87]. When the virtual hand metaphor is employed in 3D applications, a virtual representation of the user's real hand is always displayed in the VE, which can be used to grab and position virtual objects. The virtual hand acts as a virtual cursor during 3D interaction, although no finger tracking is provided. Consequently, the major design factor that defines a particular technique is the mapping between the real hand's position and orientation and the virtual hand's position and orientation.

The "classical" virtual hand provides 1:1 mapping between the real hand and the virtual cursor. To select an object, the user simply intersects the virtual cursor with the target and presses a trigger to pick it up. The virtual hand metaphor is rather intuitive since it simulates the real world interaction with objects. However, one problem with this metaphor is that only objects within reach can be selected. A number of techniques have been developed to overcome this problem. The Go-Go technique [102], for example, breaks the physical constrains by applying a nonlinear mapping to the user's hand extension. Different mapping functions can be used to achieve different control-display gains between real and virtual hands.

An alternative to using a virtual cursor is the virtual pointing metaphor, which points at objects using a virtual ray emanating from the user's hand position. The user has control over the starting point and orientation of the ray. Objects intersected by the virtual ray can be selected. The major design aspects that distinguish techniques based on this metaphor are the definition of the virtual ray direction, shape of the ray (selection volume) and methods of disambiguating objects the user wants to select [104]. Because multiple objects could be intersected, this technique can cause ambiguities. One solution provided by Hinckley et al. [50] is to augment the virtual pointing technique with a mechanism for cycling through the set of all intersected objects.

Another problem with virtual pointing is that users can have difficulty selecting small objects at a distance, because small hand motions result in large angular displacements for the ray selection spot when the objects pointed at are far away. Several variations of this technique have been designed to solve this problem. For example, the spotlight technique [71] provides a conic selection volume so that objects falling within the cone can be easily selected. However, when more than one object falls into the selection volume, a disambiguation mechanism is required to choose the target object from multiple candidates. In addition, it also depends on sufficient visual feedback indicating the volume of the spotlight cone.

Pierce et al. [99] presents a set of image plane selection techniques for users to interact with the 2D projections of 3D objects in the immersive VE. These techniques allow object selection on an image plane in front of the user by using finger gestures.

The aperture based technique developed by Forsberg et al. [36] is a modification of the spotlight technique that allows for interactive control of the selection volume size through a hand-held aperture. The conic pointer direction is defined by the location of the user’s eye, which is estimated from the tracked head location and the location of a hand sensor represented as an aperture cursor within the VE. The user can simply control the size of the selection volume by bringing the hand sensor closer or moving it further away. Although their work did not explicitly discuss the idea of using 2D image plane during selection, the aperture selection implicitly makes use of the idea by selecting an object whose projection on the 2D image plane falls within the aperture's projection when viewed from the user's eye point.

Another approach to expanding the user’s ability to access virtual objects is through manipulating the relative scale of the virtual world. The scaled-world grab technique [88] allows the user to scale and bring parts of the VE containing remote objects within reach. The environment scales back after the manipulation is finished. The world-in-miniature (WIM) technique [126] provides the user with a miniature hand-held model of the VE. The user can indirectly manipulate virtual objects by interacting with their representations in the WIM.

Occlusion is another major handicap for accomplishing spatial selection tasks. Most interaction techniques for 3D selection require the target objects to be visible. A common solution for selecting occluded objects is to navigate to a location with a clear view of the target. However, this navigate-to-select approach is impractical for dense target environment [25] [26]. Therefore, occlusion management techniques [125] [2] [61] are often essential for helping users discover and access potential targets.

Zhai et al. [153] described the use of a semi-transparent volume, called the "silk cursor", for dynamic target acquisition in 3D VE. The semi-transparent cursor provided occlusion visual cues with the location of the target object positioned behind, inside or in front of the silk cursor. A dynamic 3D target acquisition task was designed to test the effectiveness of their technique and the silk cursor demonstrated superior performance over a comparable wire frame cursor, both in monocular and in stereoscopic display conditions.

Grossman and Balakrishnan [43] presented a novel target acquisition technique based on area cursors, named the bubble cursor, which improves the isolation difficulty problem of area cursors by dynamically updating its activation area depending on the proximity of surrounding targets, so that only the object closest to the cursor center is selected. They conducted two experiments in complex situations with multiple targets of varying densities and the results showed that bubble cursor significantly outperformed other cursor techniques. Later, they explored 3D selection techniques for volumetric displays [44] and found a ray cursor superior to a 3D point cursor in a single target environment. Four variations of the ray cursor with disambiguation mechanisms for multiple intersected targets were designed to address the difficulties associated with selection in cluttered environments. Based on their work, Vanachen et al. [136] presented the evaluation of two new techniques, the 3D depth ray and the 3D bubble cursor, both of which allow for the selection of fully occluded targets in 3D VEs.

Since each different technique has their strengths and weaknesses, there have been a number of attempts to integrate them combining their best features. The HOMER technique [19], which combines ray-casting and virtual hand: after the user selects an object by ray-casting, his virtual hand instantly snaps to the selected object to allow manipulation. The virtual hand returns to its normal position after the manipulation is completed. The depth ray and the lock ray [44] adopt a hybrid approach combining a ray with a 3D cursor constrained along the ray. When the selection trigger is activated, the object intersected by the ray and closest to the 3D cursor is selected.

Some work has systematically evaluated 3D object manipulation techniques in various conditions of object selection and positioning tasks. Hinckley et al. [50] presented a survey of design issues for developing effective free-space 3D user interfaces. They discussed different techniques for 6DOF spatial input and suggested that specifying a target based on the absolute position of the tracker can be a fatiguing, consciously intensive interaction. Although no quantitative data was collected, an informal usability study by Bowman and Hodges [19] provided some useful preliminary observations. They found that naturalness is not always a necessary component of an effective technique, for instance although the Go-Go technique is more natural, users preferred ray-casting since it required less effort. They also observed that ray-casting techniques were more effective for object selection tasks, while arm extension techniques were superior for manipulation. Poupyrev et al. [105] proposed a conceptual framework and experimental testbed for studies of direct object manipulation techniques in immersive VEs, along with a practical implementation of the framework, which provided a systematic analysis of object manipulation techniques and an experimental assessment of immersive interfaces.

A number of approaches have been proposed to improve user performance in terms of task completion time and error rates [18]. A common strategy is to apply human psychomotor behavior models such as Fitts' law, which estimates the time required to select a target. However, as users are constrained by human motor skills, there is a natural trade-off between speed and accuracy. In a typical scenario, high accuracy will result in long task completion time and vice versa. Soukoreff and MacKenzie [122] presented seven recommendations with detailed justifications to HCI researchers who use Fitts' law either as a predictive model in the interface design or as part of the comparison and evaluation of novel pointing devices, in the hope of improving the comparability and consistency of forthcoming research work.

The application of Fitts' law ranges from estimating the time required to perform an assembly operation through the evaluation of different input devices [79], up to estimating times for selecting an object in 3D space [42]. Several studies have extended the Fitts' law formulation to higher dimensional tasks [90] and to account for latency [144]. Other studies [62] [133] [134] [131] analyze whether 3D object selection techniques can be modeled by Fitts' law.

Motivated by insufficient understanding of human factors implications in the design of interaction techniques for object manipulation in virtual worlds, Poupyrev and

Ichikawa [103] gave an overview of several hand based object manipulation techniques and introduced a classification that divides selection and manipulation into exocentric and egocentric techniques based on their metaphors, where egocentric manipulation is further separated into virtual hand and virtual pointing. In my research, I solely focus on the mapping functions of the virtual hand metaphor.

2.2.2 Navigation in Multi-Scale Virtual Environments

Navigation techniques can generally be partitioned into egocentric and exocentric ones [21], and both have their place. There are a wide variety of exocentric techniques including scene-in-hand [146], World-in-Miniature [98], point-of-interest (POI) techniques [82], target-object-of-interest techniques [28], prior defined volume-of-interest (VOI) techniques [63] and user defined VOI techniques [147].

Multi-scale virtual environments (MSVEs) contain geometric details over several orders of magnitude. When the display system supports head-tracking, stereoscopic viewing and/or direct 3D manipulation, MSVEs are best supported by incorporating view scale as an independent 7^{th} DOF [110] [111] [145]. Systems with these characteristics include HMDs and stationary displays with head-tracking such as CAVEs, fish-tank VR [143] and the Responsive Workbench. Southard [123] uses the term HTD (Head-Tracked Display) to distinguish the latter class of displays from head-mounted displays.

Treating view scale as a 7^{th} DOF during view maneuvers was introduced by Robinett and Warren [110]. The utility and importance of optimal spatial perception and interaction in 3D user interfaces running on systems with aforementioned features has been further discussed and motivated by various authors [141] [69] [148] [63] [147]. The view scale adjustment, either manual or automated, can generally be added to any 6DOF navigation technique. For example, the standard scene-in-hand metaphor can be augmented by an additional mode for hand-centered scaling [110].

Various exocentric 7DOF techniques are available, but specifically the scene-in-hand metaphor is chosen for the experiments for several reasons. First there are a large number of related navigation techniques including roughly half-a-dozen bimanual ones and further many object manipulation methods can be converted to view manipulations [84] [32] [86] and thus is affected by the control-display mapping. Secondly, the scene-in-hand approach requires no scene geometry be present at the center-of-rotation/scale, as for instance, POI techniques require, which makes 6DOF scene-in-hand more flexible, although possibly more challenging to learn. The added flexibility is particularly important when there are no definitive points to select for POI type of techniques, which has been remarked and empirically observed by various authors [121] [28]. This becomes particularly acute in volumetric data visualization. Furthermore, as we move from 6DOF to 7DOF navigation, the cursor location becomes important for not just the center of rotation, but also controls the center of scale.

The 6DOF (or 7DOF) scene-in-hand metaphor relies on the user's ability to position a control point at the desired location in space which serves as the center of rotation/scale. This raises the issue about the range of egocentric distance within which the user can position this control point. For this reason, the scene-in-hand metaphor raises similar questions to object selection and manipulation at a distance using various methods of 3D cursor offsets.

However, we anecdotally observe that in the particular case of 7DOF scene-in-hand travel, it is not necessary to extend the cursor's reach to arbitrary distances because

7DOF travel inherently provides a mechanism to quickly traverse large distances in the virtual environment's world coordinate space by adjusting the view scale. This suggests that when investigating cursor offset techniques for 7DOF scene-in-hand navigation, it is sufficient to extend the cursor to a more limited distance range than in other UIs (which do not integrate view scale control). At the same time, even with 7DOF travel, having a non-zero cursor offset appears useful [118] [117].

For MSVEs, scene-in-hand 7DOF navigation is a general navigation method that is usually independent of the choice of HMD vs. HTD and of the choice of a particular HTD size. Most scene-in-hand techniques display a virtual 3D cursor. As mentioned, the cursor often is offset by some amount from the tracked position using various techniques. Our experience indicates that the method used to calculate this offset needs to be modified to accommodate different display types. In particular, as will be detailed in Chapter 3, the common method of using a fixed offset vector perpendicular to the display [118] in fish-tank VR needs to be modified in a multi-display VR system, such as the CAVE. Further, offset techniques developed for HMDs normally do not lead to optimal performance when applied to 7DOF scene-in-hand navigation in a CAVE. Therefore, Chapter 4 and Chapter 5 investigate different cursor offset techniques used for controlling 7DOF travel.

2.2.3 Bimanual Interaction

Using both hands for 3D interaction allows users to transfer ingrained interaction skills, significantly increase performance on certain tasks and reduce training [21]. A great number of systems have explored two-handed interaction techniques for desktop interaction [24] [6], interactive 3D graphics [52] [8] and virtual environments [88] [116].

As a whole, these systems provide a number of case studies that demonstrate compelling bimanual interfaces for 3D applications.

Guiard's Kinematic Chain (KC) model [46] of human bimanual action is a widely accepted framework for asymmetric bimanual interactions, suggesting that the dominant hand works in the reference frame defined by the non-dominant hand. The KC model proposes general principles governing asymmetric bimanual actions where each hand plays a different role. Guiard's analysis of human skilled bimanual action also provides a fundamental theoretical insight that drives much of current experimental design and research in two-handed interaction [32] [51] [53] [56] [68].

Kabbash et al. [56] used Guiard's framework to create bimanual asymmetric techniques for Toolglass, and found that bimanual techniques improved performance by reducing the number of operations and cognitive load. But they also argued that two hands could be worse than one if an inappropriate interaction technique is employed, particularly when the cognitive load is increased. Cutler et al. [32] developed bimanual direct manipulation techniques based on Guiard's framework, and suggested that the most interesting bimanual interactions are coordinated and asymmetric. The Voodoo Dolls technique [101] also takes advantage of the asymmetric division of labor in positioning objects relative to each other, as suggested by Guiard and Hinckley [51].

In favor of symmetric bimanual interactions, Mapes and Moshell [84] designed the Polyshop for scaling, rotating and stretching graphical objects. Investigating factors affecting symmetric bimanual interaction, Balakrishnan and Hinckley [6] reported that parallelism is not a requirement for performance to be symmetric and there is no tendency for the human motor system to devote more resources to the dominant hand when the task difficulty is increased or attention is divided.

Hinckley et al. [53] presented an experimental analysis of a bimanual pointing task which suggested that Guiard's reference principle is correct but the task difficulty is also an important factor. Another experiment by Hinckley et al. [51] suggested that the two human hands together create a perceptual frame of reference, which is independent of visual feedback. Balakrishnan and Hinckley [7] further investigated interactions between visual feedback and kinesthetic reference frames in bimanual interactions, and reported that visual feedback is clearly dominant over bimanual kinesthetic reference frames when both are present, but they can independently guide bimanual actions.

Two-handed input has often been thought to improve HCI efficiency, by enabling the user to perform two sub-tasks in parallel [24], rather than as sequential operations. In a classic Wizard-of-Oz experiment, Hauptmann [48] found that users expressed rotation and scaling tasks using two hands operations spontaneously, suggesting a natural preference for bimanual actions.

Several studies have compared two-handed interaction techniques to one-handed ones. In an early study, Buxton and Myers [24] showed that bimanual methods for navigation/selection outperformed single handed methods for several measures, and the speed of performance in a positioning task was strongly correlated to parallelism or symmetry between the two hands. Balakrishnan and Kurtenbach [8] explored the use of non-dominant hand to control the viewpoint while dominant hand performing other tasks in desktop 3D applications. The bimanual technique showed significant performance advantages over the unimanual technique in the target selection task, but only marginal advantage in the object docking task.

Owen et al. [96] presented an empirical study comparing a two-handed technique to a one-handed technique for a curve matching task. They found that the two-handed technique resulted in better performance than the one-handed technique, and as the task becomes more cognitively demanding, the two handed technique exhibited even greater performance benefits.

Zeleznik el al. [152] explored bimanual techniques using two independent cursors to control camera movement in 3D desktop applications. Controlling 7DOF navigation can be done either by manually controlling all 7DOF [110] [84] [89] [30] or semi-automatically adjusting some of the 7DOFs [141] [148] [63] [147]. Chapter 5 focuses on the case of manual control of the 7DOF using a minor modification of bimanual technique, Spindle+Wheel [30], which is derived from Mapes and Moshell's [84] bimanual 5DOF+Scale technique and the Spindle feedback of Mylniec et al. [89]. Spindle+Wheel is the only technique to allow the user to control all 7DOF simultaneously using an egocentric, scene-in-hand metaphor [146].

Some researchers have applied the usage of an offset in their bimanual UIs, although they did not explicitly introduce offset as a main experimental factor. Cutler et al. [32] developed a framework for two-handed interaction and explored a variety of two-handed 3D tools and interactive techniques on the Responsive Workbench. Users wore a pair of pinch gloves equipped with 6DOF sensors to interact with the VE. They added a constant offset to the position provided by the sensor to estimate the point of action of the pinch gloves for each hand.

Leganchuk et al. [68] compared two bimanual techniques with a one-handed GUI approach using 2D area sweeping tasks, where participants hold a puck device in their left hand and a stylus in their right hand. The tracking point of the puck was offset so that the screen cursor appears at an appropriate location to reduce physical collisions of the puck and stylus on the tablet surface. Balakrishnan and Hinckley [7] explored how the mapping between the input space of the hands and the output space of graphical display influence bimanual interaction. The left hand and right hand cursor positions were constantly offset to prevent the devices from bumping into each other.

Most prior studies have focused mainly on investigating the effect of interaction-display offsets on unimanual UIs, while not much work has been done on evaluating offset techniques using bimanual interaction. Extrapolating previous results is risky because unimanual and bimanual interaction are fundamentally different in both physical motor efficiency and cognitive load. The use of two hands in human-computer interaction results in higher motor manipulation efficiency without imposing significant additional cognitive load [68]. One obvious physical motion difference between one-handed and two-handed techniques lies in the acquisition of the control point. For example in 7DOF manipulation, one-handed scaling requires a separate mode such as rate controlled scaling while the user needs to reach both hands out in front of the body to engage position control based two-handed scaling.

To our best knowledge, the effects of different cursor offset techniques on bimanual interaction have not been studied for two-handed input within an immersive VE. Chapter 3 describes virtual cursor offset techniques used in our experiments. Chapter 5 presents the evaluation of cursor offset techniques on 7DOF bimanual navigation within a

CAVE system. And Chapter 7 includes the effects of cursor offset on two-handed interaction in an HMD environment.

2.2.4 Mid-Air Interaction with Hand Tracking

In many VE applications, objects are manipulated using a hand-held device that is attached with a 6DOF tracker. This approach allows for straightforward mappings where the position and orientation of virtual objects correspond directly to those of the device. The use of physical devices is feasible, since they provide affordances for interaction, but manipulation using bare hands is more similar to real-life scenarios. Direct manipulation by hands without holding an input device means the user needs to interact with the virtual scene using hand gestures, which provides a more natural way of spatial interaction.

Interaction through natural gestures is gaining increasing popularity due to recent development of low-cost tracking systems and gesture recognition technologies. Systems with spatial free-hand input for 3D manipulation support natural interaction of pinching objects to directly grab and move them [15]. While gestural controls are common for video game (e.g. Microsoft Kinect), there has been little research conducted focusing on the usability and ergonomics of touchless gestural input in a common virtual environment. Mid-air hand gesture interaction can be broadly classified into three categories based on the motion capturing mechanism: glove tracking, marker tracking and markerless hand tracking (also refer to as “free hand tracking” or “bare hand tracking” in some literature). The research work in each category is discussed as follows.

Glove tracking has been proposed to ease and speed up the problem of hand tracking [139]. Weimer and Ganapathy [149] described a synthetic visual interface for CAD using a DataGlove for hand tracking. Gestures made by forefinger and thumb are introduced as natural metaphors for picking, clutching and scaling. The Charade system [10] developed by Baudel and Beaudouin-Lafon, provided the user with the ability to control a remote computer display with free-hand gestures by wearing a DataGlove.

Mapes and Moshell [84] presented an intuitive two-handed direct manipulation VE interface using ChordGloves for rapid and repeatable gesture recognition. They also used an offset between the thumb and forefinger to ensure accurate pinching gesture for virtual objects grabbing. Similarly, Voodoo Dolls [101] used Fakespace PinchGloves to detect pinches through the contact of forefinger and thumb. They tracked the position and orientation of the user's hands by placing a 6DOF tracker on the index fingers of these gloves.

Osawa [95] introduced automatic adjustment techniques for precise positioning and releasing 3D virtual objects based on the speed of hand movement in the immersive VE. The motions of the hand and fingers were captured using a sensor glove. Wang and Popović [140] presented a hand-tracking input system that facilitates 3D articulated user input using a single camera and an ordinary cloth glove that is imprinted with a custom pattern. They demonstrated this device for several canonical 3D manipulation tasks with precise finger and pose tracking.

Instrumented glove systems have demonstrated precise capture of 3D input for real-time control. Like other hand tracking techniques, extended use of gloves may cause fatigue for users. An interesting solution is Balloon Selection [11], which supports 3D selection using a pair of specially constructed gloves through a two-handed multi-finger technique, where the hands can rest on a multi-touch surface, minimizing fatigue.

Another type of mid-air interaction system needs fiducial markers on the user to enable hand tracking. Grossman et al. [45] explored the design of 3D interaction techniques for direct manipulation within a 3D volumetric display by leveraging thumb trigger gestures. Their prototype used a commercial motion tracking system to track the position of markers placed on the user's fingers. Kim et al. [59] presented an immersive 3D modeling system that allows for performing non-trivial tasks using only a few well-known hand gestures, with four markers worn at the fingertips.

Vogel and Balakrishnan [137] developed and evaluated three gestural pointing and two clicking techniques for distant free-hand interactions on high resolution displays. Passive reflective markers were used for hand tracking and gesture recognition through a commercial motion tracking system. Bogdan et al. [15] presented a hybrid user interface by combining 2D display and input with 3D spatial interactions, which used a 6-camera OptiTrack motion tracking system to track the 3D position of two markers placed on the user's thumb and index fingers.

Traditional marker-based motion capture systems have been exploited in several interactive systems and prototypes. However, the requirements of passive retro-reflective markers or obtrusive LEDs and expensive multiple camera setups have limited their usage, since they focus on accuracy at the cost of ease of deployment and configuration.

Markerless hand tracking continues to be an active area of research. Edge detection and silhouettes are the most common features used to identify the pose of the hand [140]. While these cues are generally robust to various lighting conditions, reasoning from them requires computationally expensive algorithms that search the high-dimensional pose space of the hand. Sato et al. [112] introduced a computer vision method for tracking hands in 3D space and recognizing gestures in real-time without using any invasive devices attached to the hand through the use of multiple cameras.

Schlattmann and Klein [114] developed a vision based hand-tracking method that can simultaneously track 6DOF+4 gestures of each hand in real-time using three cameras. Four stiff gestures are predefined to ensure synchronized markerless tracking. Using this method, they later proposed a technique [113] for efficiently grabbing and releasing virtual objects through markerless bimanual 6DOF tracking, based on the velocities of the hands. Hackenberg et al. [47] also implemented a technique for touch-based bare-hand interaction with 3D virtual content by employing a time-of-flight camera.

More recently, Wang et al. [139] presented a bimanual hand tracking system for assembling CAD components by using two consumer market depth cameras, with pose detection, hands and fingers tracking in a non-invasive manner. They developed an input metaphor that maps unimanual translation to 3D translation and bimanual translation to 3D rotation.

Song et al. [120] proposed a handle bar metaphor for virtual objects manipulation with mid-air interaction using hand gestures. It allows users to manipulate single object or pack multiple objects through relative 3D motion of both hands. The bimanual handle bar metaphor is very similar to Mapes and Moshell [84]'s 5DOF+scale interface, with the addition of incremental pitch rotation in a constrained fashion. As a result, Unlike [139], their handle bar metaphor design provides precision control for all 6DOF manipulations in a unified bimanual manner.

Mendes et al. [86] designed and implemented four mid-air techniques and one multi-touch gestures based technique for 3D virtual object manipulation on stereoscopic tabletops. Head and hands tracking were provided by integration of depth cameras. The results of evaluation indicated that 6DOF mid-air interaction with direct manipulation performed the best.

Jang et al. [55] presented a framework for simultaneous detection of click action and estimation of occluded fingertip positions from egocentric single-viewed depth image sequences. Their probabilistic inference of the detection and estimation was based on the knowledge priors of clicking motion and clicked position. Experimental results showed that their method delivered promising performance under frequent self-occlusions in AR/VR space.

In free-hand gestural interaction, hand motions are mapped directly to the object movements, only after the object has been selected. Hence, a classic problem in device-free interaction is how to signal a selection in the absence of any buttons. One solution, which has been adopted by many eye tracking systems, is to use a cursor dwell time threshold as a clicking event. But this simple method could introduce a constant lag to the interaction. Another approach that has been shown to be an effective way of "picking" in 3D space is to use the pinching gesture, which has been used by glove-based finger tracking systems for a long time. Hilliges et al. [49] used a single depth camera to detect pinches above a table top. Benko and Wilson [12] tracked pinches using an infrared camera above a projector. Wilson [151] used a webcam to detect pinches above the keyboard.

When pressing a physical button or tapping a display surface, we receive instant kinesthetic feedback confirming that the click has been triggered [137]. But for mid-air interaction, bare-hand gestural tracking techniques employ multiple-hypothesis inference to overcome the lack of strong correspondences, which may not work well in some cases. For example, in object selection tasks, the user needs to click a button to confirm the selection when holding a physical input device. The system could not miss this clicking signal whatsoever. However, with a motion capture device, using a gesture to confirm the selection could result in multiple issues such as false positive, false negative or mismatch of gestures due to various tracking capabilities of the input device. Therefore, with free space hand gestures, we need to investigate other sensory replacements in an attempt to mitigate the effects of lost kinesthetic feedback.

2.3 Control-Display Mapping

Two interaction spaces are involved in control-display mappings: the control space and the display space. The control space is the physical motor space available for the user to operate, which is constrained by the degrees of freedom available and the VR system setup. On the other hand, the display space is the visual representation of the VE within which the virtual scene may afford direct manipulation. In other words, the display space defines the scope of the user's effected actions.

The mapping between 3D control space and 3D display space has been an active research area and the separation between control and display space has been noticed since early exploration of the immersive VE. Schmandt [115] discussed his attempt to have the control space and the display space coincided, which he called spatial input/display correspondence, by using a half-silvered mirror. He discovered that improper occlusion of the hand by the virtual objects, detrimentally affected the realism of the virtual scene.

Keijser et al. [57] presented a conceptual framework for a more general and mathematical control-display description that included scale, order, lag, orientation, flip and skew as alternative mappings. They conducted a user study to explore 3D selection and manipulation tasks in special combinations of flip and skew mappings: Perspective, Mirrored Perspective, Orthographic and Mirrored Orthographic. The results showed that all three non-standard mappings could be considered as viable alternatives and the de facto perspective mapping was not the favorite in user preference.

The display space can be superimposed onto the control space, as in the classical virtual hand metaphor. Or it can be nonlinearly mapped to the control space, as in the Go-Go technique. In the former case, direct manipulation is accomplished by relying solely on the proprioceptive feedback of the hand. However, when they do not overlap, proprioceptive feedback no longer suffices and visual feedback is critical. The user has to adapt to the new motion pattern to execute accurate corrective movements.

Traditional isomorphic techniques keep 3D interaction as a direct one-to-one mapping of motion, but the development of non-isomorphic techniques opens up a whole new realm of interaction techniques and provides the power and potential to create more effective 3D interaction by changing the control-display mapping.

When creating techniques for manipulating distant objects in immersive VEs, researchers have primarily focused on increasing selection range, placement range and placement accuracy [100]. This focus has led to the rise of a series of arm-extension techniques, which dynamically scale the user's arm to allow for manipulation of distant objects. However, none of them allow the user to work seamlessly at multiple scales.

The Go-Go technique [102] scales the motion of the user's hand so objects out of reach can be grabbed, but the user is still limited by how far the arm can be extended, which makes working at large scales impossible without additional navigation. The ray-casting in the HOMER [19] technique makes it very difficult to select small objects and impossible to select occluded objects. The scaled-world-grab technique [88] shrinks the world after an object is selected using an image plane technique so that the object becomes within the reach. This explicit change in control-display mapping allows the user to manipulate any object in view, which means only visible objects can be moved.

Researchers have also developed representation-based techniques, which allow users to manipulate a distant object by manipulating its copy in a handheld representation, thus providing users with better feedback than arm-extension techniques by allowing them to view the virtual object both up close and at a distance.

The World-in-Miniature technique [126] provides the user with a miniature replica of the virtual world, so that the user can manipulate the objects in the miniature world to affect those in the larger world. However, shrinking the entire world to a small, hand-held model makes it almost impossible to select small objects and place them precisely. Also, the WIMs are pre-defined, which fixes the scale of the interaction, rather than allowing the scale to change on the fly. Similarly, the Voodoo Dolls technique [101] enables users to dynamically create transient copies of the target object using an image plane technique [99] and interact with them directly. Thus, this technique allows dynamic change of scale in control-display mapping to facilitate manipulation, as well as the ability to select both visible and occluded objects.

Pierce and Pausch [100] presented a formal experiment comparing Voodoo Dolls technique with HOMER technique for manipulating objects at distance in immersive VE. The result of their study showed that the Voodoo Dolls technique allowed users to both position and orient objects more accurately than the HOMER technique. The results also suggested that the visual feedback is important for 3D interaction techniques.

The study of control-display gain [81] was incorporated into HCI research for a long time. In the HCI community, it is a tradition to conduct Fitts' law based studies to evaluate and improve target acquisition (i.e. object selection) techniques, such as Bubble Cursor [43], Semantic Pointing [14], Adaptive Pointing [60], SQUAD [61], etc.

As an important factor, CD gain has been explored extensively in the context of physical and virtual control device design. MacKenzie and Riddersma [81] compared CRT and LCD displays with different settings of CD gain on interactive systems on a routine 2D target acquisition task using a mouse. Movement times were significantly lower in medium CD gain than in low and high gains.

CD gain has also been exploited in the studies on the effects of different body parts in manual control. Gibbs [40] compared task performance of three different body parts: the thumb, the hand and the forearm in a one dimensional target acquisition task, using joystick controls of both position and rate control systems with various control gains and time delays. They found that hand and forearm were superior to the thumb with short time lags and high CD gains could improve pointing performance although it tends to amplify undesirable effects such as limb tremor.

Based on Gibbs' work, HCI researchers have been studying the differences in performance between the muscle groups controlling various segments of human upper limb. In particular, Zhai et al. [157] investigated the effect of small muscle groups and joins of fingers on human performance differences in 6DOF input control. Balakrishnan and MacKenzie [9] conducted Fitts' law analyses to determine the relative bandwidths of fingers, wrist and forearm in a typical serial pointing task. In order to accommodate different ranges of motion of the limb segments, they varied the CD gain to maintain the same visual stimuli across all limb conditions.

Casiez et al. [27] theoretically and empirically examined the impact of CD gain on mouse pointing performance on a standard desktop display and on a very large high-resolution display, using two modifications of CD gain: constant gain where CD gain is uniformly adjusted by a constant multiplier and pointer acceleration where CD gain is adjusted using a non-uniform function depending on movement velocity, at various levels. They found that low levels of gain had a noticeable negative effect on performance due to the increased clutching and maximum limb speeds. High gain levels had little impact on performance, with only slight increase in time when selecting very small targets.

In order to improve ergonomic design of touchless gestural interfaces, Riyal et al. [109] evaluated the effect of different CD transfer functions on user performance using a standard 2D Fitts' pointing task in a touchless gestural control system. The system allows the user to clutch and click through movements of their thumb and middle finger tips and thumb and index finger tips respectively. Motion tracking is achieved by LED markers placed on the dominant hand of the user. Results indicated high level of CD gain in the gesture control incurred significant increase in error rates while lower gain showed the best performance and lowest error rates. However, CD gain had no significant effect on movement time.

The first 3D selection technique proposing a dynamic CD gain is the Go-Go technique, which allows the user to stretch the virtual arm to select distant objects, but the precision decreases as the hand is moved further from the torso because movements are magnified. Since the CD gain depends on the distance between the user's hand and torso, some studies show that people tend to judge their hand movements mainly on the basis of the displayed cursor and adapt their motion accordingly [60].

König et al. [60] divided the solutions to human precision limit of mid-air distant pointing on a high resolution display with an infrared laser pointer into three categories based on the way CD gain is modulated: target-oriented [14], manual-switching [137] and velocity-oriented [37] approaches. To improve pointing performance for absolute input devices, they introduced the Adaptive Pointing technique, which implicitly adapts the CD gain to the current user's needs without violating the user's mental model of absolute-device operation by dynamically adjusting the CD gain depending on the movement velocity and current offset between the motor-space position and display-space position.

Blanch et al. [14] introduced the semantic pointing technique which decouples the motor space from visual space to facilitate pointing movements by dynamically adapting the CD gain to control the mapping between motor and visual space. They also showed that the difficulty of a pointing task is defined by the size of the target in motor rather than visual space, thus making the motor size a function of the CD ratio that is used to reflect the local semantics of the system.

Vogel and Balakrishnan [137] explored the design space of freehand pointing and clicking interaction with very large high resolution displays from a distance. They created two hand gestures and three pointing techniques to enable the user switch the CD gain manually between a constant value for absolute mode and a conventional acceleration function for relative mode. The results showed that while the absolute mapping technique was significantly faster, it also caused higher error rates.

A specific design challenge for direct manipulation is to provide the user with the ability to perform precise adjustments to objects as well as the ability to move objects quickly in the 3D space. To meet this challenge, Frees and Kessler [37] proposed PRISM (Precise and Rapid Interaction through Scaled Manipulation), a technique that adjusts CD gain in correspondence with the user's behavior. PRISM seamlessly switches between precise and direct mode by dynamically adjusting the CD ratio based on the current velocity of the user's hand. The results of the user study showed that PRISM performed significantly better than the traditional direct manipulation approach. Later, Frees et al. [38] extended the PRISM technique to object rotation and developed an enhanced version of PRISM with ray casting to increase the speed and accuracy of object selection, with evaluations demonstrating their effectiveness. Techniques using a dynamic CD gain, such as PRISM and Adaptive Pointing, require an offset recovery mechanism to avoid an excessive decoupling between control and display space.

Inspired by PRISM technique, Wilkes and Bowman [150] studied the relationship between HOMER and PRISM, and applied the velocity-based scaling principle to the HOMER technique. The results of the user study showed that the addition of scaling to HOMER significantly improved user performance on 3D manipulation tasks. Similarly, Auteri et al. [4] combined Go-Go and PRISM technique, to increase the precision of object manipulation in 3D immersive VE. They found that the combination of two techniques yielded a nearly 2:1 improvement in precision over traditional Go-Go when attempting to align two objects at a distance.

Direct manipulation could be inefficient when the user need to reach for a distant object, due to the natural limitation of the physical arms. In addition to the Go-Go technique, researchers have developed several nonlinear motion control techniques for both object manipulation and navigation in the virtual environment.

Song and Norman [121] presented nonlinear motion control techniques for both viewpoint movement and hand positions in order to help the user get a panoramic view of the virtual scene. The principle of their idea is to divide the working space of the input device into several regions and use different mapping functions to transfer the motion of the device into the virtual space for each region.

McMahan et al. [85] presented a study that separated the effect of level of immersion and 3D interaction technique for a 6DOF manipulation task in a CAVE environment. Three techniques were compared in their experiment: HOMER, Go-Go and DO-IT (Desktop Oriented Interaction Technique). The results indicated that there was no significant difference of object manipulation time between Go-Go and HOMER.

Plenty of work has been done in the evaluations of various navigation techniques under different VE settings [20] [129] [63] [127], but few of them compare the Go-Go technique and the HOMER technique directly. Chen et al. [28] compared these two techniques in an Information-Rich VE, but as navigation techniques. The user grabs the world (Go-Go) or grabs an object (HOMER) to change the viewpoint. The result showed that Go-Go performed significantly better than HOMER and thus is better suited for navigation that requires easy and flexible movements. They also inferred that for manipulation based navigation techniques, those who use ray-casting and involve object selection for viewpoint movement would be less usable.

The control space and display space are not always overlapped. When tracking the user's hand position, an offset can be added to the virtual representation of the hand. For example, in image plane selection [99], the pointing direction is defined by roughly aligning the hand with the eye position, which requires the user to keep his arm extended. Introducing a vertical offset allows the user to keep his hand in a lower position, thus reducing fatigue levels. However, decoupling control and display spaces may also result in performance loss. Human seem to achieve optimal manipulation performance when haptic and graphic displays of objects are superimposed [138].

Several studies have explored the effect of offset between measured distance in physical space and controlled distance in the virtual scene. Poupyrev et al. [104] evaluated two generic metaphors, the virtual hand and the Go-Go technique, for egocentric object selection and manipulation in an HMD. They indirectly addressed the problem of direct and distant manipulation by comparing two techniques. They found that there was no significant difference between these two techniques in local selection conditions, whereas for object repositioning at a constant distance, classical virtual hand was 22% faster than the Go-Go technique on completion time. In this dissertation, the Go-Go technique is used for a comparison of different cursor offset techniques. Although it is designed for one-handed 6DOF object manipulation, I extend its usability to both unimanual and bimanual 7DOF view manipulations.

Mine et al. [88] presented a framework to investigate the effect of proprioception on various interaction techniques using an HMD. They carried out a study to explore the difference between manipulating virtual objects that are collocated with the user's hand and those that have a translational offset on an object docking task. The experiment has three conditions for the main independent variable: manipulation of objects held in one's hand, objects held at a fixed offset and objects held at an offset varying with the subject's arm length. Their results showed that users had better performance with manipulation of objects that are collocated with their hands than with manipulation of objects at a fixed or varied offset.

Paljic et al. [97] conducted a study about close manipulation using a two-screen Responsive Workbench. The experiment explored the influence of manipulation distance on user performance in a 3D location task, which consists of clicking on a start sphere and then clicking on a target sphere that appears at one of nine locations. The subjects were asked to hold a tracked stylus in their dominant hand to control the virtual pointer. The offset between the tip of the stylus and the virtual pointer is introduced as the main factor with four levels: 0, 20, 40 and 55cm. The target sphere position is another factor. The results of the analysis indicated that task completion time using 0 and 20cm were significantly shorter than using 40 and 55cm.

Due to the fact that both Mine's work and Paljic's study revealed that distant manipulation impairs user performance, Lemmerman and LaViola [70] conducted an experiment to explore the effect of a positional offset between the user's interaction frame-of-reference and the display frame-of-reference on a different type of task in a surround-screen VE. In their experiment, the subjects were first asked to perform a centering task to ensure they begin with the same position for each trial, and then they needed to match colors using a 3D color-picking widget. Three different positional offsets between the input device and the graphical feedback were presented as the main factor: zero offset, 3 inches offset and 2 feet offset. For the centering task, their results showed that collocation or a short offset could increase user performance, which complies with Mine and Paljic. However, the results from the color matching task indicated that zero offset condition could reduce the performance accuracy. Their explanation is that object docking is a coarse task while color matching task requires close attention and precise operation.

More recently, Bruder et al. [23] evaluated effect of visual conflicts for mid-air 3D selection within arm's reach on a stereoscopic table with a Fitts' law experiment. They compared three different levels of visual conflicts for selecting a virtual object: real hand, virtual offset cursor and virtual offset hand. Their results showed that the real hand condition resulted in highest error rate, but also the highest effective throughput, which suggests that virtual offset-based techniques do not improve overall performance.

CHAPTER 3: VIRTUAL CURSOR OFFSET TECHNIQUES

3.1 Motivation

This chapter describes cursor offset techniques and elaborates on their differences.

In a prior study that I'm involved, a one-handed 7DOF exocentric technique was used for travelling in a fish-tank MSVE [29]. The travel technique required precise cursor placement relative to scene objects. Users tended to adjust the view scale so that the scene locations they wanted to navigate around remained within reach of the 3D cursor's range of motion. Importantly, a fixed translational offset (perpendicular to the screen) was used between buttonball and the cursor and there was no gain factor between device motion and cursor motion [118]. When porting the same navigation technique to a CAVE, the question arose of how to handle the offset between the buttonball and the cursor.

Compared to fish-tank VR, the user tends to stand further away from the screen in a CAVE environment due to the larger display size. This implies at least the magnitude of the translational offset needs to be increased. However, our informal study showed that this alone was not enough. In the CAVE, the direction of the offset is also important (recall that the approach in fish-tank VR is to translate perpendicular to the lone screen). Therefore, the offset algorithm designed for the CAVE needs to control both the magnitude and the direction in order to support a cursor offset in any direction to be used in different screens that has different orientations (i.e. 360°). This brings us into the research area of arm-extension techniques reviewed in Chapter 2.

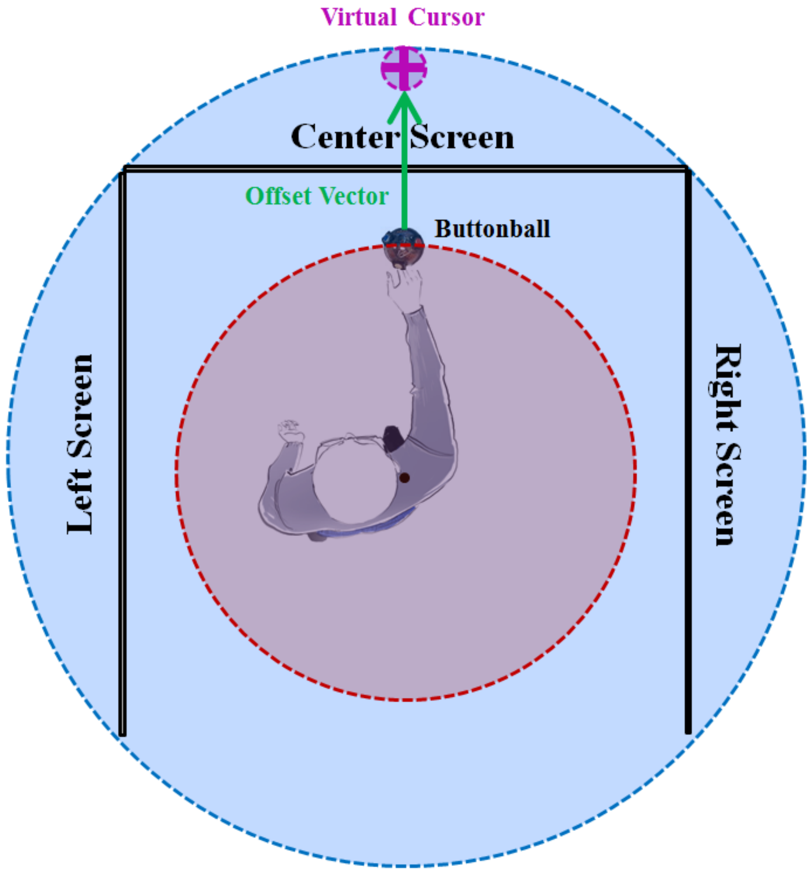


Figure 3.1: Range of the user's hand (red circle) and the 3D cursor (blue circle).

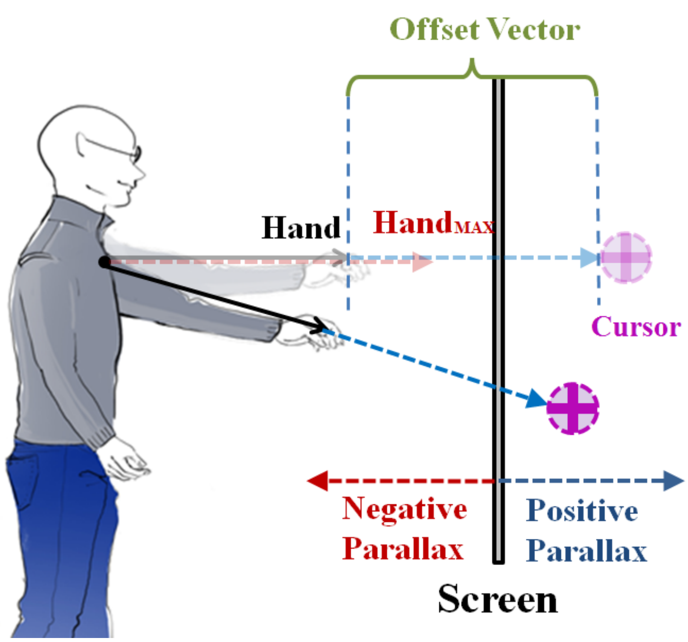


Figure 3.2: A side view of Figure 3.1.

Figure 3.1 and Figure 3.2 illustrate the goal of such offset techniques in a CAVE system. The red circle indicates the range of the hand tracker (buttonball in our case) and the blue circle shows the larger range of the cursor. This cursor range could vary considerably depending on the offset calculation used. The green arrow represents the offset vector, which starts from the hand tracker and ends at the center of the cursor:

$$P_{cursor} = P_{hand} + \hat{v}_{offset} \qquad \text{(Equation 3.1)}$$

The $\hat{v}_{offset}$ calculation is discussed below for three techniques.

3.2 Fixed-Length Offset Technique

In fish-tank VR, $\hat{v}_{offset}$ is perpendicular to the screen and of fixed size. Several algorithms were tested informally to enable dynamic switch between using the various CAVE screens' orientations for $\hat{v}_{offset}$ while the user was interacting with geometry across multiple screens. None of the methods were proved to be satisfactory. Each time any algorithm switched the chosen screen, the cursor would abruptly change its position as $\hat{v}_{offset}$ instantly changed ±90 degrees. If the user was interacting with objects whose 2D projections straddled a screen corner, the algorithms tended to bounce back and forth between the different $\hat{v}_{offset}$ directions causing the cursor to bounce around. Further, trying to choose which screen should determine $\hat{v}_{offset}$ proved difficult. The tracked head orientation is not an accurate predictor of which screen the user is looking at. Various heuristics based on which screen the cursor's projected 2D image fell on worked poorly as well. During some bimanual operations each one of the two cursors would briefly appear on a different screen. In general, heuristic approaches for dynamically picking a screen on which to base $\hat{v}_{offset}$ did not match user expectations with a high enough frequency [29].

For these reasons, our fixed-length offset technique is independent of any particular screen. In the fixed-length offset condition, the direction of $\hat{v}_{offset}$ is the same as the vector $\hat{v}_{chest \to hand}$ (the "hand vector") which points from the user's chest to the hand. (If only the head and hand are tracked, the position of the user's chest is approximated based on the position and orientation of the head tracker). The formula has a constant coefficient C:

$$\hat{v}_{offset} = C \cdot \frac{\hat{v}_{chest \to hand}}{\|\hat{v}_{chest \to hand}\|} \qquad \text{(Equation 3.2)}$$

C should be determined empirically and perhaps adjustable by the user.

3.3 Go-Go Offset Technique

The Go-Go technique [102] allows the user to directly manipulate both nearby objects and those at a distance by using a nonlinear mapping between the user's hand and the virtual hand. I adapted their method to the calculation of the offset vector:

$$\hat{v}_{offset} = \begin{cases} \hat{0} & if\ L_H < D \\ k(L_H - D)^2 \cdot \frac{\hat{v}_{chest \to hand}}{L_H} & otherwise \end{cases} \qquad \text{(Equation 3.3)}$$

Where $L_H = \|\hat{v}_{chest \to hand}\|$ and k is a coefficient: $0 < k < 1$. This indicates that as long as the user is reaching for nearby areas ($L_H < D$), there is no offset and the cursor is coincident with the user's hand. I use the same value for D as 2/3 of the user's arm length. When the user reaches her hand further than D, the mapping becomes nonlinear and the movement of the cursor becomes quadratic to the movement of the user's hand, but the offset vector $\hat{v}_{offset}$ and the hand vector $\hat{v}_{chest \to hand}$ still have the same direction.

3.4 Linear Offset Technique

In the informal test, I observed that sometimes under the fixed-length offset condition, it was not very convenient for the user to navigate in the negative parallax area.

Especially when the targeted location was very close to the user's body, the user could not directly put the virtual cursor anywhere near the target. Also under the Go-Go offset condition, I noticed that the position of the virtual cursor became more sensitive to the motion of the physical input device when the user reached out further due to the nonlinear mapping function. Therefore, a more dynamic offset technique is desirable to overcome the disadvantages from the previous two techniques.

I implemented a new technique called the linear offset technique, which enables the user to travel more effectively in the VE by creating an intuitive linear mapping between the user's hand and the virtual cursor. In the linear offset approach, the direction of $\hat{v}_{offset}$ remains the same with $\hat{v}_{chest\to hand}$. The magnitude of $\hat{v}_{offset}$ depends on two preset parameters: maximum arm reach M_{arm} and maximum offset length M_{offset}, as well as the magnitude of $\hat{v}_{chest\to hand}$:

$$\hat{v}_{offset} = \left(M_{offset} \cdot \frac{\|\hat{v}_{chest\to hand}\|}{M_{arm}}\right) \cdot \frac{\hat{v}_{chest\to hand}}{\|\hat{v}_{chest\to hand}\|} = \frac{M_{offset}}{M_{arm}} \cdot \hat{v}_{chest\to hand} \quad \text{(Equation 3.4)}$$

In Equation 3.4, the offset vector $\hat{v}_{offset}$ changes linearly with the hand vector $\hat{v}_{chest\to hand}$, which implies that when the user's hand is close to the body, the offset added to the virtual cursor will be short; vice versa, when the user tries to move her hand away from the body, the offset length will increase accordingly. This design provides a natural extension to the user's arm by dynamically adjusting the offset length based on the arm motion.

Figure 3.3 shows offset distance of the four offset techniques by a hand position. According to the graph, only the Go-Go offset technique has a nonlinear mapping function. By adjusting the coefficients, all techniques allow the cursor to reach a predefined max distance position when the user's hand reaches to her maximal arm extent $Hand_{MAX}$ except for the no offset technique. The maximum distance (from the virtual cursor to the user's body) is approximately 72" (≈1.83m), but it is varied with the user's arm reach.

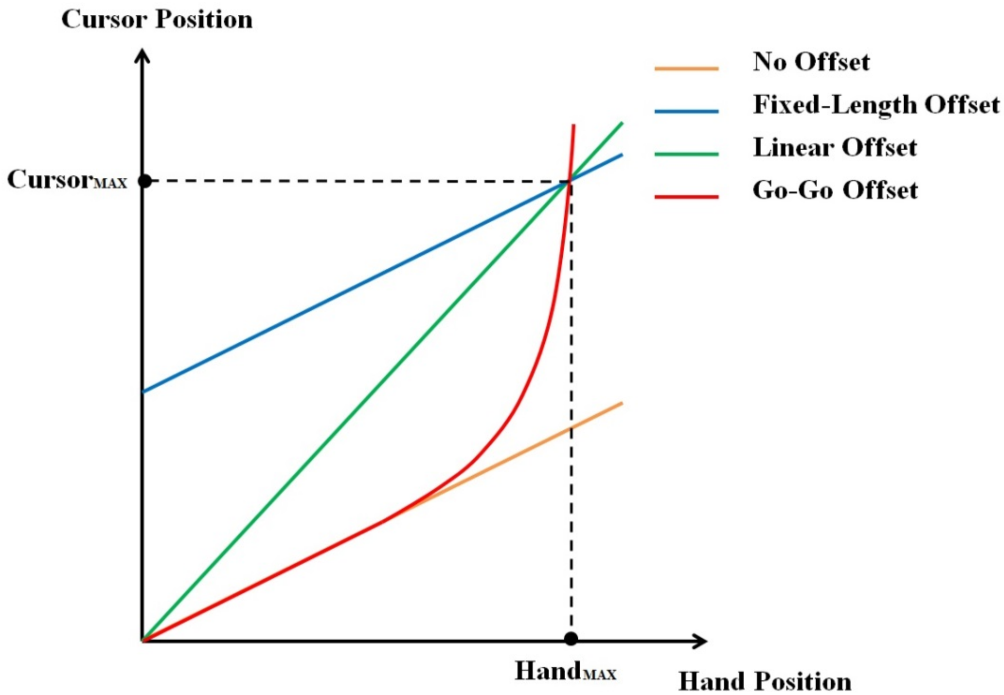


Figure 3.3: Mapping functions of the four offset techniques.

It is not intuitively clear which offset technique is more effective within the VE. Therefore, systematic analysis and experimental evaluation are critical for understanding the usability characteristics of the offset techniques and for providing developers with guidelines to allow informed design decisions.

CHAPTER 4: EVALUATION ON ONE-HANDED 7DOF NAVIGATION IN A MULTI-DISPLAY VIRTUAL ENVIRONMENT

4.1 Introduction

Immersion in a VR application can be enhanced by giving the user the ability to move around in the VE with natural physical motions [19]. By using head tracking and 3D spatial input devices, the user can navigate in the VE in order to obtain different visual perspectives of the scene. Since 3D input devices usually have larger working ranges than traditional 2D devices, most travel techniques that employ direct positioning metaphors for 3D viewpoint movement control typically involve a gain factor parameter for the input device [82] [121]. The gain factor should be carefully chosen when building the map between the position of the 3D input device in the physical world and the position of the 3D virtual cursor in the virtual world.

This chapter examines the effects of cursor offset techniques in a CAVE system for scene-in-hand 7DOF navigation (the content of this chapter is published to 3DUI 2015 [72]). It is common practice in desktop VR systems to have a fixed translational offset between the hands and the virtual cursors [118] to allow the user to maintain an elbow-resting posture. The offset is perpendicular to the display screen. Within a CAVE system, such an offset could allow the shoulders to stay relaxed during a broader range of cursor manipulation. Naive porting of this offset technique proved problematic.

Ease of scene-in-hand 7DOF navigation depends on the ability to place the cursor, which defines the center-of-rotation as well as the center-of-scale, at strategically optimal locations within the scene during navigation maneuvers. In order to explore the effect of cursor offset techniques on user performance, two experiments are conducted on a 7DOF navigation task using a one-handed scene-in-hand [146] travel technique. Experiment 1 compares four different offset techniques: no offset, fixed-length offset, Go-Go offset and linear offset. As an extension of Experiment 1, Experiment 2 investigates which offset length in the linear offset technique yields optimal user performance.

4.2 Evaluation

Two formal user studies are conducted to evaluate cursor offset techniques on user performance in a CAVE system when navigating in a MSVE using an exocentric scene-in-hand travel metaphor. This section describes the designs and procedures.

4.2.1 Environment

Our CAVE system consists of three large displays and a Polhemus Fastrak tracker with a wide range emitter (shown in Figure 4.1).The physical size of each display is 8ft×6.4ft (2.44m×1.95m) with a screen resolution of 1280×1024. The overall dimension of the CAVE is 8ft×8ft×6ft (2.44m×2.44m×1.95m) and screen resolution is 3840×1024. The head tracker is attached to the side of the polarized glasses. For hand tracking and operations, the user holds a precision-grasped buttonball that has a 6DOF receiver fixed inside (Figure 4.2). The virtual environment used for the experiments is written with OpenSceneGraph [94] and a custom VR API.

4.2.2 Experimental Design

A 7DOF navigation task is used in both experiments to evaluate the effect of varying the offset between the physical tracker and the virtual cursor. The 3D virtual cursor is a transparent 3D sphere in the scene that represents the buttonball (Figure 4.2).

The user is asked to perform the navigation task by holding a buttonball with her dominant hand. A scene-in hand travel technique [146] is used for the view manipulation. The top left button engages 6DOF navigation using the scene-in-hand metaphor and the top right button engages rate controlled scaling [21] (Figure 4.3). The center of scale is determined by the cursor's position when the top right button is first pressed [110]; a separate, small red sphere will appear to indicate the center of the scale.

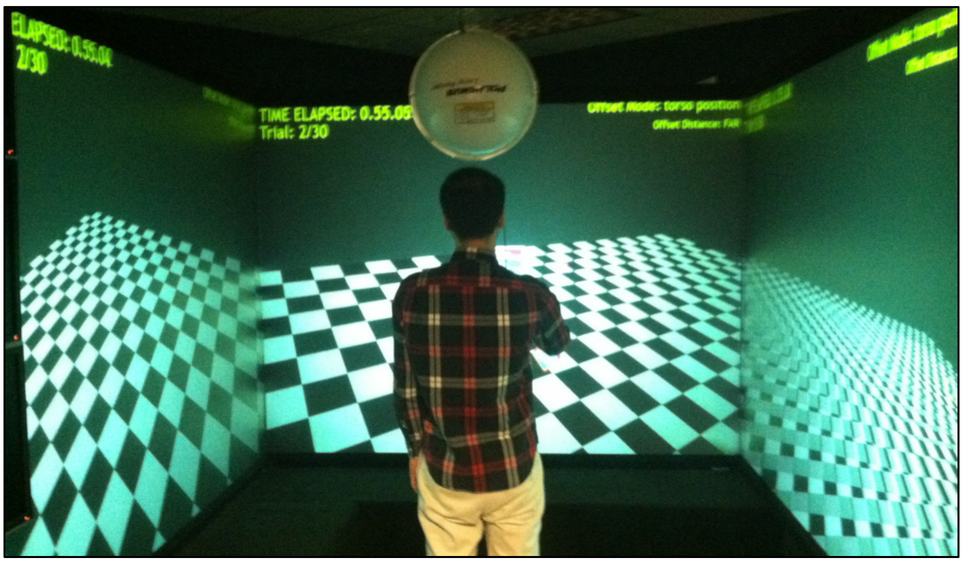


Figure 4.1: Our three-side CAVE system. Polhemus Fastrak tracks the position and orientation of the user's head and 3D input.

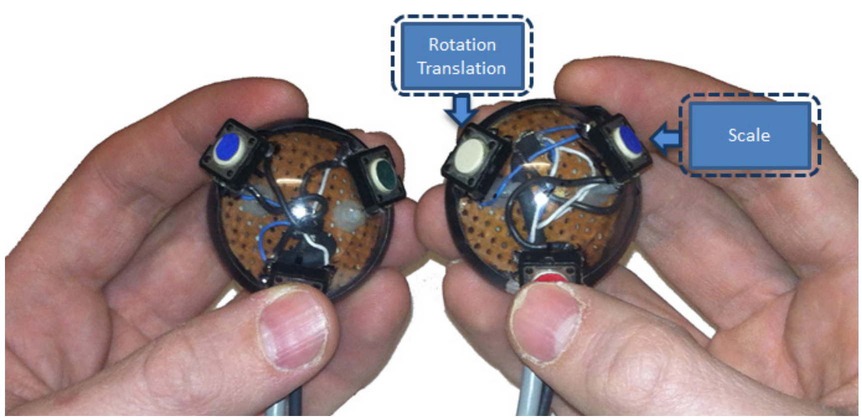


Figure 4.2: A pair of the buttonball devices. Each buttonball has three buttons on its surface and a Fastrak receiver inside of the ball to track the user's hand position.

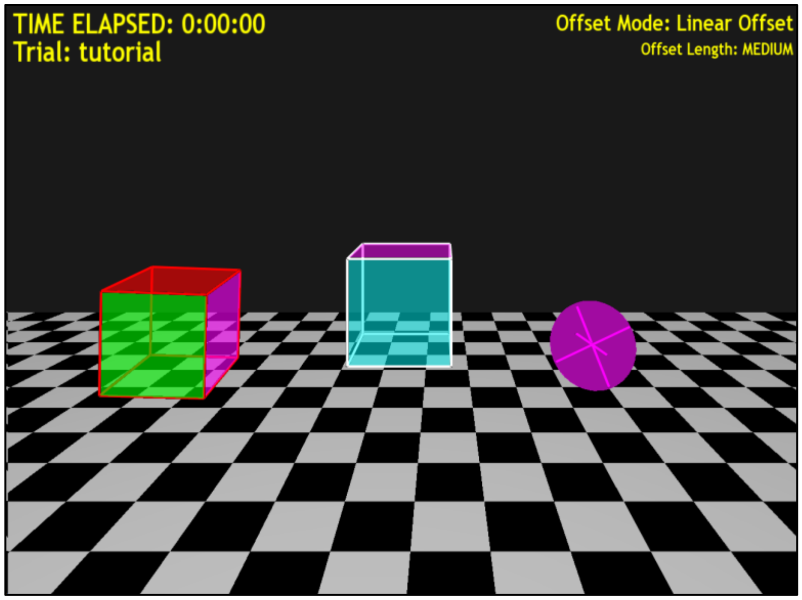


Figure 4.3: A screenshot of our virtual scene. The target box (white outline) is placed at the center of the center display and the docking box (red outline) is located at a random position above the grid ground.

A screen capture of our virtual environment is shown in Figure 4.3. The VE consists of a checker-board ground plane and two transparent color boxes. The size of the ground is 8ft×8ft. The initial position of the ground plane is set in a manner that half of it appears in front of the center screen and the other half appears behind the center screen. At the center of the center screen is the target box, which is a transparent cube with a side length of 1 inch. This cube has a white outline and a different color at each face. It remains stationary relative to the screen during travel. For each trial, a docking box with a red outline appears at a random location above the ground plane. This cube can show up in any one of three sizes: 25%, 100% or 400% of the target box's size, and at any location within the range of the ground plane. The position, orientation and size of the docking box are randomly generated across the trials. The goal of the task is to align the docking box with the target box. To finish the task, the user must travel to maneuver the view pose and view scale to match the size and orientation by using the buttonball device.

A timer appears at the upper left of the screen indicating how much time has elapsed since the start of the current trial. Right below the timer is the trial indicator, which tells the user how many trials have been completed. Upper right of the screen shows the offset mode for Experiment 1 and the offset length for Experiment 2. When the distance of corresponding vertices between the target box and the docking box is within a tolerance (0.84cm) [157], the outline of the target box turns green and a chime sound plays. The user must release the navigation engagement button to stop the timer. Once the outline of the target box becomes green, the user can press the third button (the bottom one) to finish the trial, and next trial will start immediately, in which case the timer will be reset to zero.

4.2.3 Procedure

Upon arrival at the study location, each subject is first asked to sign the informed consent form and then complete a short prequestionnaire. Next, the subject is briefed with the purpose of the experiment and is introduced the VE and tracking devices.

After the experimenter has demonstrated the docking process, the subject is asked to wear the stereo shutter glasses and begin a short training session where she learns how to use the buttonball and to engage the view manipulation. Each practice trial is identical to those performed during the experiment and the ordering of the practice condition blocks is the same as it would be in the actual trials. During the practice, the experimenter remains in the study environment with the subject to act as a guide and the subject is encouraged to ask any clarifying questions. The entire training session lasts for approximately ten minutes.

In Experiment 1, the subject is also asked to complete a calibration step before she can advance to the actual trials. The calibration step measures the foremost reach of each subject and sets the parameters so that the virtual cursor can reach the same point when the subject straightens her arm forward under fixed-length offset, Go-Go offset and linear offset conditions. To acquire the arm measurement, the subject is asked to stand in the center of the environment, reach straight forward while holding the buttonball. The experimenter watches the subject perform this calibration step to ensure that a proper measurement is recorded. The foremost distance of the virtual cursor using fixed-length offset and linear offset can be determined by the arm reach alone, but for Go-Go offset, the gain factor k is also needed to be adjusted based on the value of the arm extension in order to reach the same distance.

When the subject is ready and parameters are all set, she can start the actual trials. As described in the experimental design section, there are four sessions in either of the studies. Each session contains 30 trials and uses a different offset technique or offset length. Each subject is instructed to align the target cube with the docking cube as quickly as possible, but no time limit is imposed. The subject can take a short break between the sessions. The application records the task completion time and number of button clicks for each trial. At the end of the experiment, the subject is asked to fill out a post-questionnaire regarding subjective preferences on the offset techniques or offset lengths, as well as opinions on how the target box size and parallax condition affect the interactions.

The repeated measures ANOVA (analysis of variance) with per-trial mean of task completion time are used for quantitative analysis for both experiments. The reported F tests use α=.05 for significance and use the Greenhouse-Geisser correction to protect against possible violation of the sphericity assumption. The post-hoc tests are conducted using Fisher's least significant differences (LSD) pairwise comparisons with α=.05 level for significance.

4.2.4 Experiment 1: Unimanual 7DOF Navigation in CAVE

Experiment 1 compares four different offset techniques for navigation tasks: No Offset (NO), Fixed-Length Offset (FO), Go-Go Offset and Linear Offset (LO). Each participant should complete 120 trials (5 trials × 4 offset techniques × 3 box sizes × 2 parallax conditions) in a within-subject design repeated measures ANOVA. Sixteen participants are recruited (twelve male and four female; four CS major and twelve non-CS major). All participants have 20/20 (or corrected 20/20) eye vision and no disability using their arms and fingers. One participant is left-handed and the other eleven are right-handed. Participants have high daily computer usage (6.38 out of 7) and nine of them have experience with 3D user interfaces, such as Microsoft Kinect or Nintendo Wii mote.

The experiment has three main factors: offset technique, target box size and target box's initial position. The target box can appear either in the positive parallax part of the ground plane which is the space behind the center screen, or in the negative parallax part which is the space in front of the center screen. The order of offset techniques is counterbalanced between subjects using a Latin square.

The primary hypotheses of Experiment 1 are:

H1: The fixed-length offset, Go-Go offset and linear offset techniques are expected to have faster completion time than no offset because they increase the 3D cursor distance.

H2: The linear offset technique is expected to have faster task completion time than the fixed-length offset, because it is easier to navigate to the negative parallax area.

H3: The linear offset technique is expected to outperform the Go-Go offset technique, because the Go-Go technique increases the cursor distance quadratically so that it makes the view pose more sensitive to control.

4.2.4.1 Quantitative Results

Table 4.1 shows average of task completion time (*CT*) and standard deviation (*SD*) by offset technique and box size conditions of Experiment 1. The result of a three-way (Offset Technique × Box Size × Parallax) repeated measures ANOVA shows a significant main effect on task completion time for the offset technique factor ($F(1.81,27.16)=10.92$, $p<.001$, $\eta_p^2=.421$, see Figure 4.4). Pairwise comparisons show that the completion time of LO ($M=15.06$) is significantly faster than NO ($M=26.33$, $p<.001$), FO ($M=19.48$, $p=.013$) and Go-Go ($M=23.55$, $p=.001$). In addition, the completion time of FO is significantly faster than NO ($p<.001$). The task completion time of Go-Go is not significantly different from either NO ($p=.306$) or FO ($p=.176$). As hypothesized (*H1*, *H2* and *H3*), the linear offset technique outperformed other offset techniques. These results indicate that the user takes advantage of LO for the traveling task. Compared to NO, however, adding a quadratic offset to the virtual cursor (i.e. Go-Go) does not enhance user performance of the traveling task while FO and LO do. Interestingly, the results indicate that FO is not better than Go-Go.

The main effect for Box Size is also significant ($F(2,30)=104.48$, $p<.001$, $\eta_p^2=.874$). LSD tests show that completion time of 100% box size ($M=12.97$, $SD=2.73$) is significantly faster than 25% box size (*M*=27.89, *SD*=7.36, *p*<.001) and 400% box size (*M*=22.45, *SD*=6.08, *p*<.001). In addition, completion time of 400% box size is significantly faster than 25% box size (*p*<.001). This is because 100% box size only requires 6DOF while others require 7DOF (6DOF+scale) for the navigation task.

Table 4.1: Average completion time (CT) and standard deviation (SD) of each condition in Experiment 1.

	NO		***FO***		***GoGo***		***LO***	
Size	***CT***	***SD***	***CT***	***SD***	***CT***	***SD***	***CT***	***SD***
25%	32.9	9.9	28.5	15.7	29.9	11.3	20.4	8.0
100%	17.0	5.2	10.2	4.3	15.8	9.8	8.8	3.0
400%	29.1	9.6	19.7	7.0	25.0	11.4	16.0	5.1
All	26.3	10.8	19.5	12.6	23.5	12.2	15.1	7.4

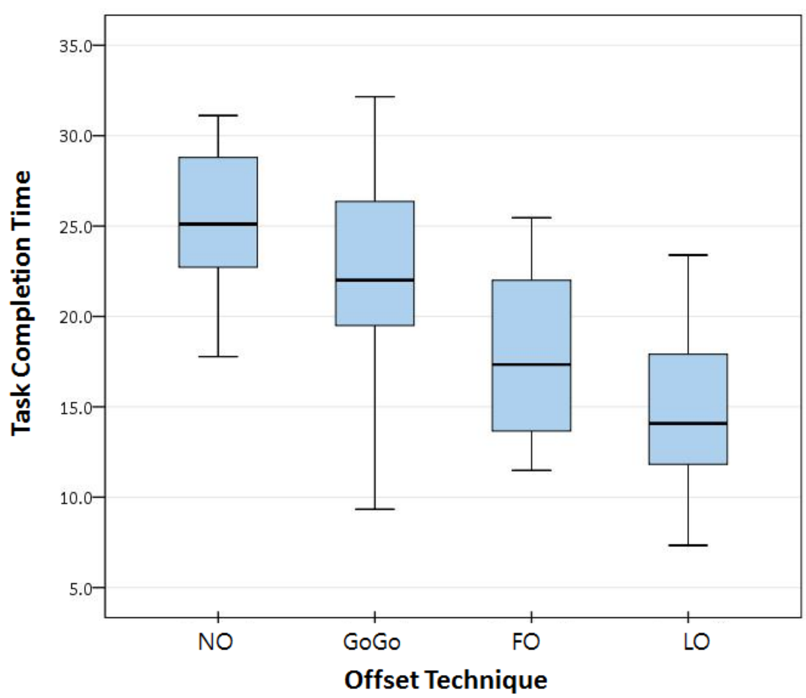


Figure 4.4: Boxplot of task completion time of offset techniques (No Offset, Go-Go Offset, Fixed-Length Offset and Linear Offset).

6DOF vs. 7DOF

To clarify the effects of offset techniques for different DOFs, two-way ANOVAs were performed on different box sizes respectively (25% vs. 100% and 400% vs. 100%). The results reveal a significant interaction effect on task completion time for Box Size × Offset Technique (25% and 100%, $F(3,45)=3.106$, $p=.036$, $\eta_p^2=.172$). There is a simple effect of the offset technique condition in 25% box size ($F(3,45)=6.067$, $p=.001$, $\eta_p^2=.288$), and there is also a simple effect of the offset technique condition in 100% box size ($F(1.633,24.491)=11.584$, $p=.001$, $\eta_p^2=.436$). In 6DOF tasks (100% box size), LO is faster than NO ($p<.001$) and Go-Go ($p=.003$). FO is faster than NO ($p<.001$) and Go-Go ($p=.021$). But there is no difference between Go-Go and NO ($p=.615$) and LO and FO ($p=.147$). In 7DOF tasks (25% box size), however, only LO is faster than all other techniques (NO ($p<.001$), FO ($p=.017$) and Go-Go ($p<.001$)).

There is an interaction effect between DOF and offset technique conditions (400% and 100%, $F(3,45)=3.662$, $p=.019$, $\eta_p^2=.196$). The results show a simple effect of the offset technique factor in 400% box size ($F(1.840,27.594)=12.012$, $p<.001$, $\eta_p^2=.445$). LO is faster than other three techniques (NO ($p<.001$), FO ($p=.009$) and Go-Go ($p=.002$)). In addition, FO is faster than NO ($p<.001$). Overall, the results indicate that users perform 7DOF tasks faster with LO than with other three techniques, while in 6DOF tasks, the difference between offset techniques is less significant.

4.2.4.2 Subjective Preferences

Participants rate arm fatigue level on a 7-point Likert scale from 1 ('Not at all') to 7 ('Very Painful'), after finishing each offset technique session. The Friedman test shows a significant main effect on fatigue rate (χ^2 (3)=7.992, $p=.046$). However, Wilcoxon signed-rank tests with a Bonferroni correction ($p<.008$) do not show any significant difference between levels (FO vs. NO: $p=.041$, Go-Go vs. FO: $p=.030$, and LO vs. Go-Go: $p=.042$).

When asked which offset technique is the easiest when the target box appears in the positive parallax area, eleven out of sixteen answered LO, two answered Go-Go, one answered FO, one both FO and LO, and one did not choose any technique. For the negative parallax, twelve selected LO as the easiest technique, two selected FO, one answered both FO and LO, and one did not choose.

When asked to choose the easiest offset technique overall, twelve out of sixteen preferred LO, one preferred FO, one preferred Go-Go, one chose both FO and LO and one chose both FO and Go-Go.

4.2.5 Experiment 2: 7DOF Travel with Different Linear Offset Lengths

The results of Experiment 1 show that the linear offset technique outperforms other offset techniques. Based on this, the effect of four different offset lengths: 0” (0cm), 24” (60.96cm), 48” (121.92cm) and 96” (243.84cm) of the linear offset technique on the same navigation task is evaluated. These four offset lengths are chosen based on the dimension of our CAVE environment. The distance from the center of the CAVE to a screen is 4ft (48”). With the 48” offset length, the user can move the cursor in a negative or positive parallax area with little arm movement. I speculate that if the offset length is shorter or longer than 48”, then the user performance will decrease because it requires more arm movement to move the cursor to a certain parallax area.

Another sixteen participants are recruited for Experiment 2 (nine male and seven female; ten CS major and six non-CS major). Each participant performs 120 trials (5 trials × 4 offset lengths × 3 box sizes × 2 parallax conditions). Two participants are left-handed and the other ten are right-handed. Participants have high daily computer usage (6.56 out of 7) and seven of them have experience with 3D UIs.

The primary hypothesis of Experiment 2 is that adding a translational linear offset to the virtual cursor would help user perform better than without it. But I do not have a definitive conjecture about which offset length is the most effective under our virtual environment setting, because the short offset condition and the long offset condition are expected to work better in negative parallax area and positive parallax area respectively, while the medium offset condition could potentially excel on average.

Table 4.2: Average completion time (CT) and standard deviation (SD) of each condition in Experiment 2.

	0”		**24”**		**48”**		**96”**	
Size	***CT***	***SD***	***CT***	***SD***	***CT***	***SD***	***CT***	***SD***
25%	34.5	14.0	20.9	5.9	21.3	8.2	18.1	3.6
100%	14.9	4.4	9.7	3.0	8.6	3.3	8.0	2.1
400%	30.3	16.5	18.9	6.9	17.7	5.1	15.9	5.2
All	26.6	15.2	16.5	7.3	15.9	7.9	14.0	5.8

4.2.5.1 Quantitative Results

Table 4.2 shows average of task completion time (*CT*) and standard deviation (*SD*) by box size and offset length conditions of Experiment 2. The result shows a significant interaction effect for Box Size × Offset Length ($F(2.73,40.89)=4.23$, $p=.013$, $\eta_p^2=.220$, see Figure 4.5). There is a simple effect on completion time of offset length for 25% box size ($F(1.869,28.034)=17.925$, $p<.001$, $\eta_p^2=.544$). Completion time of 0” is significantly slower than 24” (p<.001), 48” (p<.001) and 96” (p<.001). In addition, completion time of 24” is significantly slower than 96” (p=.040). For 100% box size, there is also a simple effect on completion time of offset length (F(3,45)=26.512, p<.001, η_p^2=.639). Same as 25% box size, completion time of 0” is significantly slower than 24” (p<.001), 48” (p<.001) and 96” (p<.001) and 24” is significantly slower than 96” (p=.047). Moreover, there is a simple effect on task completion time for 400% box size (F(1.394,29.911)=10.806, p=.002, η_p^2=.419). Completion time of 0” is significantly slower than 24” (p=.006), 48” (p=.002) and 96” (p=.003) and 24” is significantly slower than 96” (p=.046). Overall, 96” is the fastest offset length for all three box sizes and it is also significantly faster than 24”. However, there is no statistical difference between either 48” and 96” or 48” and 24”.

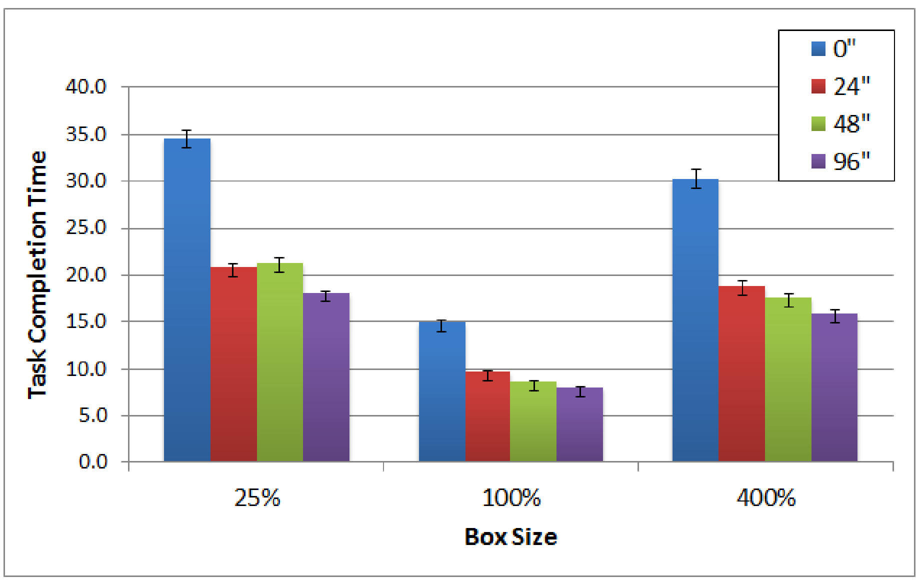


Figure 4.5: Task completion time by box size and offset length. The error bar represents ±1.0 standard error.

There is a significant main effect of box size on task completion time (F(2,30)=83.58, p<.001, η_p^2=.848). Pairwise comparisons show that the completion time of 100% box size (M=10.28, SD=1.74) is significantly faster than 25% box size (M=23.73, SD=5.76, p<.001) and 400% box size (M=20.70, SD=5.46, p<.001). Also, 400% box size is significantly faster than 25% box size (p=.009). This result indicates that users would perform better if no scaling operation is required (i.e. 6DOF) and scaling down the virtual scene is easier than scaling the scene up.

The main effect of offset length on task completion time is also significant (F(1.68,25.20)=19.67, p<.001, η_p^2=.567). Pairwise comparisons show that the completion time of 0" (M=26.58) is significantly slower than 24" (M=16.49, p<.001), 48" (M=15.86, p<.001) and 96" (M=14.01, p<.001). Task completion time of 24" is also significantly slower than using 96" (p=.016). However, the completion time using 48" is not significantly different from either 24" (p=.670) or 96" (p=.125). This result indicates that adding an appropriate length to the virtual cursor would be helpful to enhance user performance for the navigation task.

4.2.5.2 Subjective Preferences

The Friedman test shows a significant main effect on fatigue rate (χ^2 (3)=12.520, p=.006). Followed up Wilcoxon signed-rank tests with a Bonferroni correction (p<.008) show that users felt more arm fatigue with 0" than with with 24" (Z=-2.804, p=.005, r=5.66). They also felt more arm fatigue with 0" than with with 96" (Z=-2.698, p=.007, r=5.66).

When asked which offset length is the easiest when the target box appears in the positive parallax area, twelve out of sixteen answered 96", three answered 24" and one answered 48". For the negative parallax, six chose 96", five 24", four 48" and one 0". Overall, ten out of sixteen preferred 96", three 48" and three 24".

4.3 Discussion

The result of Experiment 1 shows that the linear offset technique performs better than both no offset and Go-Go offset techniques. However, there is not any statistically significant difference between the Go-Go and no offset techniques although the Go-Go technique has the same maximum offset length of the cursor as the linear technique. This could be explained by different levels of sensitivity due to the gain factor. The Go-Go technique changes the cursor position quadratically in the nonlinear mapping area, which increases the sensitivity of the gain factor. While the previous research shows the advantage of the Go-Go technique for object selection and manipulation, it did not bring any advantage to the user for the direct view manipulation technique. Furthermore, the linear offset technique outperforms other techniques, including the fixed-length offset technique, when the navigation task requires 7DOF (pose+scale) interaction.

Previous research shows that minimal offset is optimal for object selection or manipulation tasks in a surround-screen VE [70], an HMD [88] and a Responsive Workbench [97]. The results of Experiment 2, however, indicate that the 96” offset length enhances user performance the most. I conducted an informal study that extended the offset length to 144” (365.76cm) but the result did not show any statistical difference between 96” and 144”. The main difference between our navigation task and their selection or manipulation task is that their task does not allow the user to release and re-grab a target object during the trial. For our navigation task, the user is able to freely relocate the cursor without having to manipulate the view. In addition, the user does not need to select a specific object for view manipulation, which gives her the ability to engage in view manipulation anywhere in the virtual world. This freedom requires relatively less accuracy of the interaction technique which is affected by the gain factor.

Our study's task is 7DOF navigation. User controlled view scale adjustment is a fundamental part of the interaction, which makes it possible that an offset technique that only allows the cursor to extend to, say, 10ft in physical space is sufficient, because this translates to range of 10ft × View Scale in virtual space. If view scale is not changeable, for instance in a system with 6DOF navigation and a selection task, then being able to extend the cursor 100's or 1000's of feet in physical space becomes necessary. It is our experience and of others, however, that when performing 7DOF navigation in MSVEs using an exocentric navigation technique (such as scene-in-hand or the Mapes-Moshell bimanual technique [84]), users normally need and use much smaller motion range of the cursor than in this selection example. For example, Wartell et al. [147] navigate MSVEs in the Responsive Workbench with cursor based 7DOF navigation techniques with only a fixed-offset. However, when trying the similar technique in the CAVE, fixed-offset is not optimal; yet prior experience suggested that it is not necessary to be able to reach 100's of feet in physical space. Therefore a linear offset technique appears to be optimal.

The results of both Experiment 1 and 2 do not reveal any statistical differences of the parallax factor. However, based on the subjective results and our observation during the experiments, users have difficulty manipulating the view with the fixed-length offset technique when a target box is close to the user. Most users would step backwards under this circumstance in order to bring the cursor closer to the target box. This may be the reason why it took the user more time to finish the navigation task with the fixed-length technique than with the linear offset technique. Using the fixed-length technique, the user cannot bring the cursor all the way towards her body.

One of the important factors in the measurement of usability and efficiency of an interaction technique is accuracy evaluation. This could be done by separating DOFs (translation, rotation and scale). This study, however, solely focuses on how the offset techniques help accomplish 6DOF and 7DOF navigation tasks. The efficiency and usability across offset techniques likely differs depending on the type of interaction technique with which it is combined and the task. As some previous research reported, a nonlinear arm extension technique outperforms other techniques for selection. It is also possible that the offset techniques discussed here may perform differently with other navigation techniques or navigation tasks.

4.4 Conclusion

This chapter presented two user studies of 3D cursor offset techniques in a head-tracked, stereoscopic three-side CAVE system. Experiment 1 compared four different 3D virtual cursor offset techniques and Experiment 2 compared four different offset lengths for navigation tasks in the CAVE system. The results suggest that using the linear offset technique could reduce task completion time for 6DOF and 7DOF navigation tasks. Furthermore, a longer offset distance (96”) is more helpful to the user to complete the task than a shorter offset distance.

CHAPTER 5: EVALUATION ON BIMANUAL 7DOF NAVIGATION IN A SURROUND-SCREEN VIRTUAL ENVIRONMENT

5.1 Introduction

The capability of efficiently interacting with the VE is often achieved by utilizing a real-world metaphor, in which users can reach out their hands and manipulate the virtual objects or viewpoint using natural physical motions [110]. However, this real-world metaphor does not allow large-scale interaction without further navigation in the virtual scene, for which arm extension techniques [102] [19] are developed that add an offset between the user's hand and the virtual surrogate. While most of prior studies on offset and distance of manipulation have focused on one-handed interaction, the design and influence of such an addition on two-handed interfaces are mostly left unexplored. Compared to unimanual operation, it is more natural and intuitive to reach out with both hands to operate in the real world [84]. To achieve this ease of manipulation, a number of VR systems have provided with facile bimanual 3D interfaces.

Chapter 4 investigates the effect of virtual cursor offset on a one-handed scene-in-hand [146] 6DOF travel technique for 7DOF navigation tasks in a CAVE. Due to the inherent differences between one-handed and two-handed interactions in both physical motor efficiency and cognitive visualization, whether their conclusion can be applied directly to two-handed navigation under similar environment setting remains unknown. This chapter conducted a formal user study with the objective of exploring the effect of translational offsets between the positions of 3D input devices in the physical world and the positions of 3D virtual cursors in the virtual world during two-handed 7DOF navigation. The experiment compares four different offset techniques: no offset, fixed-length offset, Go-Go offset and linear offset. A 7DOF navigation task using a two-handed travel metaphor [30] is used in all conditions.

5.2 Bimanual Offset Technique

Bimanual interaction refers to using both hands in a continuous and coordinated manner, while performing a particular manipulation task in the VE. Consequently, when a translational offset between reference frames is needed in order to enlarge the space of motion, the mechanism of adding such offsets should be designed carefully to ensure the respective interaction of left and right hands with the virtual scene will not bring too much cognitive load altogether to the user. The rationale for the choice of techniques evaluated in this study is explained next.

After having studied everyday activities to understand how humans distribute work between their left and right hands, Guiard [46] classifies manual activities into three categories: unimanual, bimanual symmetric, and bimanual asymmetric. The most common activities involve an asymmetric division of labor between the left and right hand [32], which have been studied in various VR systems concerning asymmetric bimanual interaction techniques [84] [118] [51] [7] [30].

Compared to one-handed interaction, using both hands can help users obtain a better sense of the space they are working in, which may change the way users think about the task. This could lead to exploring alternative strategies for problem solving when both hands can be used. To evaluate the effect of cursor offsets on bimanual interaction, the simultaneous 7DOF interaction technique [30] is adopted for view manipulation in the CAVE environment. This technique allows the user to control 7DOF (6DOF pose + 1DOF scale) continuously. What's more, this technique has been modified to allow the user to switch between her dominant and less-dominant hands to control travel, which adds more flexibility and sense of symmetry to the interaction.

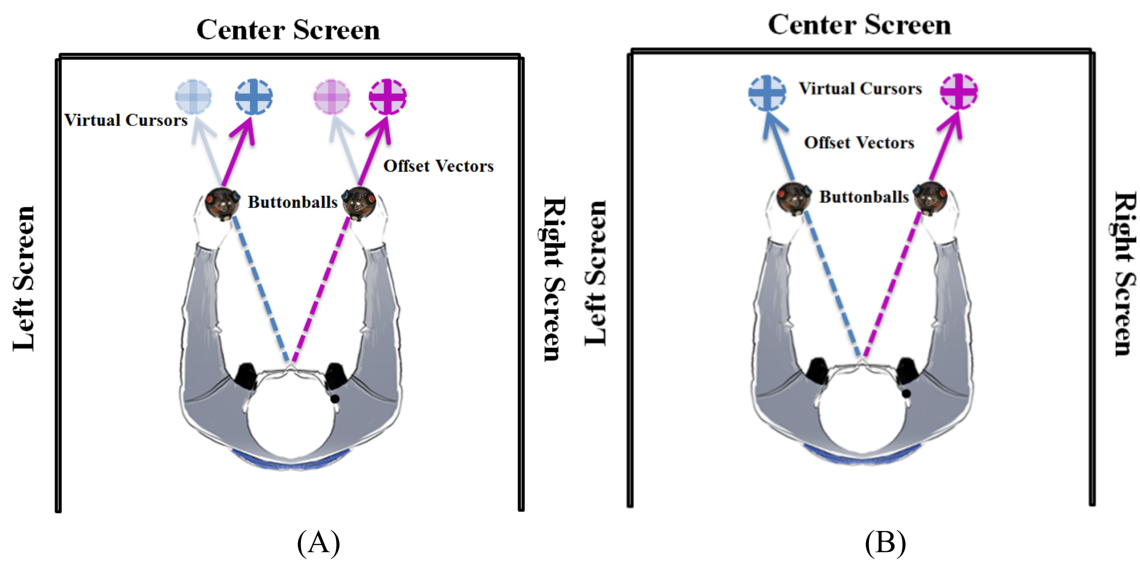


Figure 5.1: Two options of adding offsets to bimanual interaction. (A) The same offset vector used for both hands. (B) A distinctive offset vector added for each hand.

In the experiment and illustrations, two-handed interaction uses hand-held devices as opposed to direct 3D hand tracking. The offset issues are similar whether the bimanual interaction uses virtual cursors with tracked held devices or with directly tracked hands. In CAVE and HMD systems, for one-handed input the offset vector direction is generally computed as a function of the displacement between the tracked device position and the user's torso. When considering how to compute the offsets in a bimanual case, there are two possible options: 1) adding the same offset vector to both hands or 2) using different offset vectors for left and right hands. These two cases are both illustrated in Figure 5.1, where a pair of virtual cursors represents the positions of the physical input devices (a pair of buttonballs in our study). In Figure 5.1(A), the virtual cursors for left and right hands are both translated by the same offset vector. Two pairs of offsets are illustrated in blue and purple. These indicate that the offset could be calculated based on either left hand position (the blue vector pair) or right hand position (the purple vector pair). Figure 5.1(B) shows the alternative case where the two virtual cursors are translated by different offset vectors, with each vector calculated based on the position of left hand and right hand respectively.

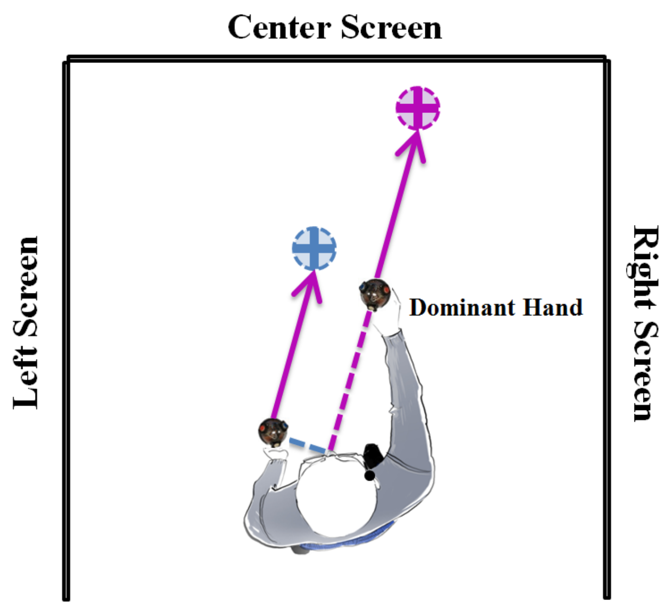


Figure 5.2: Inconsistency caused by adding the dominant hand offset vector to the non-dominant hand.

An informal study was conducted to test these two options. The participants used the two-handed 7DOF interaction technique to perform a 7DOF travel task that requires navigation to 'dock' a target box with a docking box (Figure 5.4). During the informal study, I observed some disadvantages of using the same offset vector. The offset vector used was that of the dominant-hand. When two hands are not equally away from the body, the single offset vector can cause confusion, as illustrated in Figure 5.2. In this case, the non-dominant hand is close to the body while the dominant hand is farther away. While the left and right hands are on opposite sides of the body, the offset direction is biased to the right, which pushes both cursors to the right side of the body. This tends to introduce an unnatural, additional asymmetry into the bimanual interface.

Another issue brought by using the initiating hand's offset is that when the user switches control between two hands, it causes a sudden change of direction of the offset for virtual cursors. This process can be seen from Figure 5.1(A). When the control changes from the right hand to the left hand, the offset vector being used switches from the solid purple arrow to the semi-transparent blue arrow, which causes a sudden shift of both virtual cursors to the left. This abruptly change degrades the user's ability for precise cursor placement.

Both problems described above could cause cognitive overload to the user, whereas option 2 overcomes such drawbacks by assigning a different offset vector to each hand (Figure 5.1(B)), thus it enables a smooth and intuitive addition of offset without introducing inconsistency to the graphical feedback.

5.3 Experiment 3: Bimanual 7DOF Navigation in CAVE

A formal user study is conducted to evaluate four different virtual cursor offset techniques: No Offset (NO), Fixed-Length Offset (FO), Go-Go Offset (GO) and Linear Offset (LO), on user performance in a CAVE environment when navigating an MSVE by a modified Spindle+Wheel travel technique [30]. Sixteen unpaid volunteers (twelve male and four female) are recruited for the study, with ages ranging from 23 to 30 (M=26.06). All participants have 20/20 (or corrected 20/20) eye vision and no disability using their arms and fingers. Participants have high daily computer usage (5.69 out of 7) and four of them have experience with 3D UIs, such as Microsoft Kinect or Nintendo Wii mote.

5.3.1 Environment

This experiment uses the same environment setup the one-handed study in Chapter 4. Figure 5.3 shows the bimanual interface. The same pair of buttonball devices are used (Figure 5.4), but with different button functions.

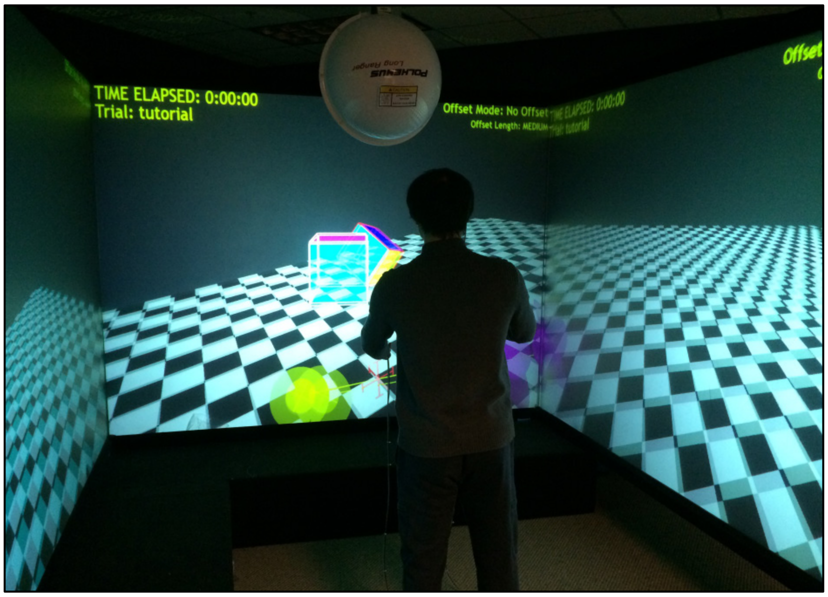


Figure 5.3: The three-side CAVE environment used for the bimanual experiment.

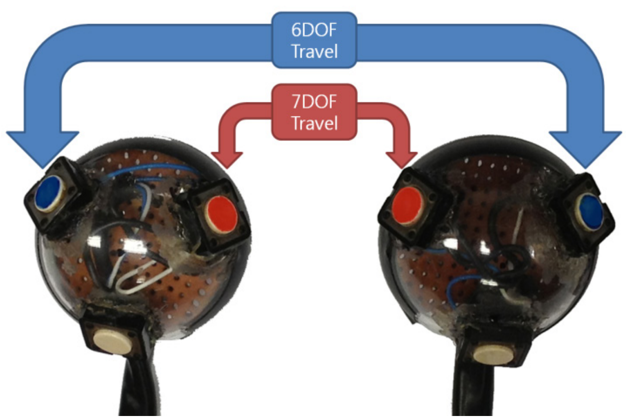


Figure 5.4: The buttonball devices with Fastrak trackers fixed inside.

5.3.2 Experimental Design

The effect of varying offset between the physical tracker and the virtual cursor is evaluated in the experiment using a 7DOF navigation task. We use a modified two-handed 7DOF travel technique [30], Spindle+Wheel for the view manipulation. The Spindle+Wheel technique allows simultaneous 7DOF (6DOF+scale) interaction with multi-scale VEs by adding an additional pitch DOF control to Mapes and Moshell's [84] original bimanual 5DOF+Scale technique with further augmented Spindle visual feedback [89]. In Spindle+Wheel, translating the hands rigidly translates the view point. Rotating one hand about the other while keeping their distance constant, rotates the view in yaw and roll. Moving two hands closer or farther apart scales the view. The Spindle+Wheel visual feedback is shown in Figure 5.5. The yellow and purple semi-transparent spheres are the 3D virtual cursors. A thin cylinder (the 'Spindle') is drawn between the virtual cursors with a small red sphere at the center point indicating the precise center of scale (or rotation). The wheel on each sphere is the visual feedback for the pitch operation. Spinning the less-dominant hand's buttonball around the axis of the wheel with the fingers rotates the view around the spindle axis.

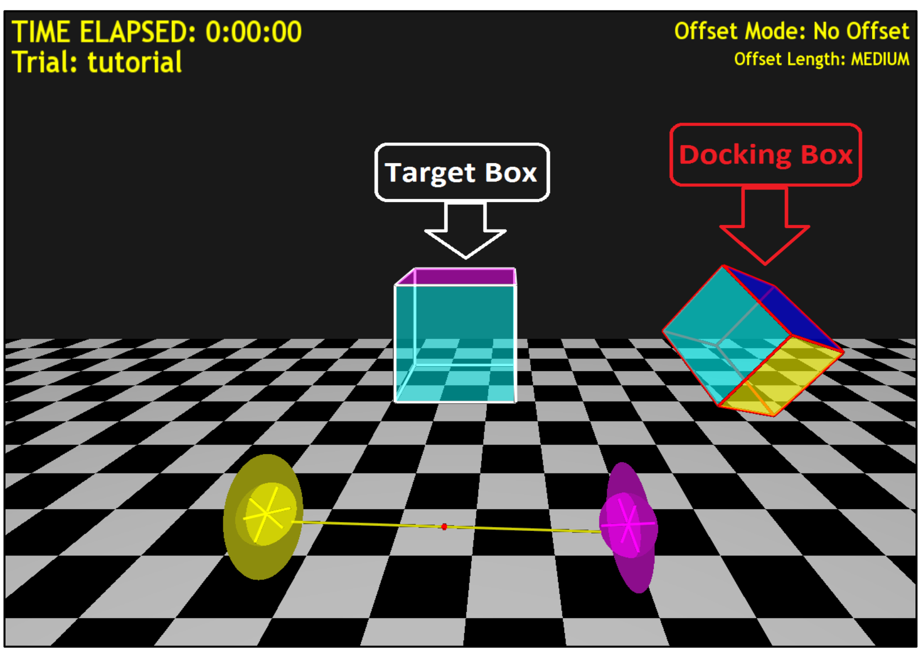
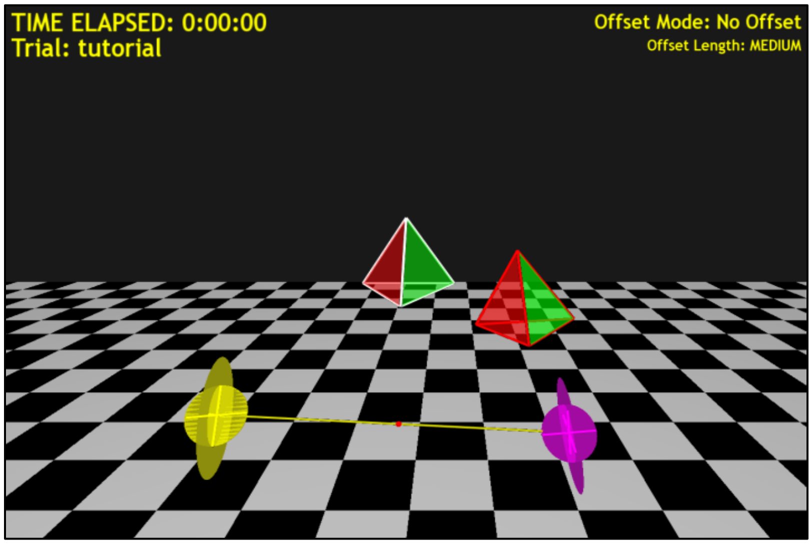


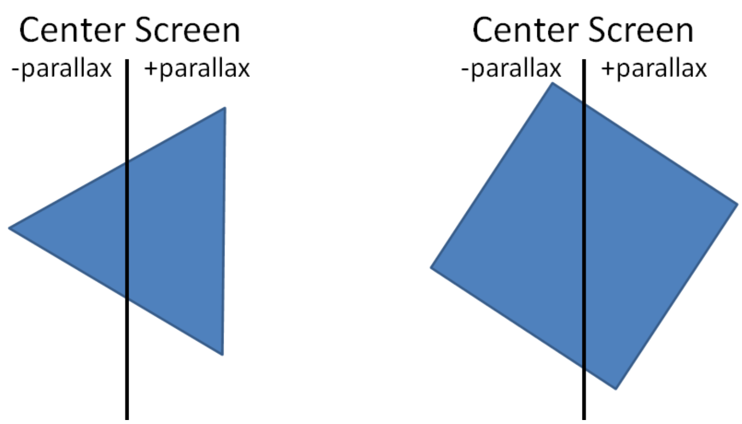
Figure 5.5: A screen capture of the VE. The target box is at the center of the center display and the docking box is located at a random position above the grid ground.

This experiment implements Spindle+Wheel with two modifications. First, pressing and holding the red button on either buttonball (Figure 5.4) engages 7DOF Spindle+Wheel navigation. The buttonball whose red button is pressed controls the pitch DOF. (In the original Spindle+Wheel only the user's less-dominant hand's buttonball's red button is activatable). Hence, in our implementation, the user can freely choose which hand controls the pitch operation. Second, prior work shows that when 5DOF+Scale derived techniques are used for 7DOF navigation and when the user performs a view maneuver where she does not desire to change the scale, having the scale DOF always engaged leads to accidental scale changes [30]. Therefore, a second separate mode is added, pressing and holding the blue button on either buttonball enables only the 6DOF (translation+rotation) Spindle+Wheel.

Figure 5.5 shows a screen capture of the virtual scene being used, which is exactly the same as in previous experiment in Chapter 4. The scene consists of a checker-board ground plane and two transparent boxes. The plane is 8ft (2.44 m) square with half appearing in front of the center screen and half behind. At the center of the middle screen is the target box, which is a cube with an edge length of 1ft (0.3 m). This cube has a white outline and a different color for each face. It remains stationary relative to the screen during travel. At each trial, a second cube appears at a random location on the ground plane, which is the docking box with a red-outline. This cube comes in three sizes: 25%, 100% and 400% of the target box's size, and can show up anywhere within the range of the ground plane. The docking box's location, orientation and size vary randomly across the trials. The user must travel to align the docking box with the target box, which requires view pose and view scale maneuvers to match the size and orientation. The 25% and 400% cases require the user to scale up or down the scene by 4 or 1/4. Given our ground plane size, the 25% cube remains easy to see and its orientation is discernible at the farthest distance.



(A)



(B)

Figure 5.6: (A) Test environment of using tetrahedron in the docking task. (B) A top-down view comparison between a tetrahedron and a cube as the docking geometry.

A timer at the upper left corner of the screen informs the user how much time has elapsed since the start of the current trial. The progress indicator is right below the timer, which shows how many trials have been completed. The current offset condition is shown on the upper right of the screen. When the distance of corresponding vertices between the docking box and the target box is within a tolerance (0.84cm) [157], the outline of the docking box turns green and a chime sound plays. The user must release the navigation engagement button to stop the timer. Then the user presses the bottom white button to advance to the next trial, with the timer being reset.

In the initial design of the docking task, a tetrahedron was used as the docking geometry, as shown in Figure 5.6(A). Compared to a cube, a tetrahedron has fewer faces. Along with its triangular shape, a tetrahedron occupies smaller space volume and provides fewer perceptual cues, such as visible edge and faces, than a cube does (Figure 5.6(B)). Prior work [157] uses tetrahedra for the docking task, but their test environment is a fish tank VR system, compared to which, the CAVE system provides more movement space for the convenience of the user's spatial judgment. Furthermore, our research group has a history of using cubes for user studies, which leads to the final decision of using a cube as the docking geometry for this experiment.

Our experiment has three main dependent variables: offset technique, docking box size and whether its initial position is in front or behind the screen. When the new trial begins, the docking box can show up in either positive parallax or negative parallax region. The offset technique condition order is counterbalanced between all subjects using a Latin square.

Our primary hypotheses for the experiment are:

H1: Fixed-length offset, Go-Go offset and linear offset techniques are expected to have faster completion time than no offset. This is because all three techniques extend the area of direct manipulation, thus increasing the efficiency of navigation.

H2: The linear offset technique is expected to outperform the fixed-length offset. This is because linear offset provides a flexible mechanism to dynamically adjust the offset length based on arm stretching, instead of using a constant offset.

H3: The linear offset technique is expected to outperform the Go-Go offset technique. This is because the nonlinear mapping of the Go-Go technique makes the view pose more sensitive to control due to the quadratic gain factor.

5.3.3 Procedure

Upon arrival at the study location, each subject is asked to sign the informed consent form followed by a short pre-questionnaire. Then the subject is briefed with the purpose of the study. After the experimenter has introduced the VE and demonstrated the docking process, the subject puts on the polarized glasses and grasps the buttonballs to begin a short training session where she learns how to use the Spindle+Wheel view manipulation. During the practice, the experimenter remains in the study area to answer any of the subject's questions. The training session is approximately 10 minutes.

Before the formal study starts, the subject is required to complete the calibration step which measures the subject's foremost reach, in order to ensure that the virtual cursor can reach the same distance when the subject maximally out stretches her arm during the fixed-length offset, Go-Go offset and linear offset conditions. This point is approximately 72" (≈1.83m) from the user's body, but the distance is varied with the user's arm reach. I choose this length based on the dimension of our CAVE environment; the distance from the center of the CAVE to each screen is 4ft (48"). With the 72" length, the user is able to place the virtual cursor in both in negative or positive parallax regions with 7DOF travel technique while standing at the center of the CAVE.

After the parameters are set, the subject starts the trials, which are divided into four sessions. Each session contains 30 trials (5 trials × 3 box sizes × 2 parallax conditions) and uses a different offset technique. As a result, each subject should complete 120 trials in a within-subject repeated measures design. The subject is instructed to align the docking box with the target box as fast as possible, but no time limit is imposed. They can also take a short break between trials or sessions. At the end of the experiment, the subject fills out a post-questionnaire regarding subjective preferences of the offset techniques, as well as opinions on how the docking box size and parallax conditions affect the navigation task.

Table 5.1: Average completion time (CT) and standard deviation (SD) of each condition.

		NO		FO		GO		LO	
	Size	***CT***	***SD***	***CT***	***SD***	***CT***	***SD***	***CT***	***SD***
+	**25%**	50.5	25.3	38.6	19.6	46.8	20.0	25.5	8.4
	100%	28.7	10.1	18.4	7.1	31.2	14.2	14.3	5.2
	400%	33.9	21.6	23.2	8.4	27.6	8.1	18.1	5.7
−	**25%**	42.4	22.2	44.3	23.9	42.8	17.1	23.7	6.1
	100%	26.1	9.7	20.8	10.4	25.8	9.3	14.0	3.0
	400%	30.1	16.8	23.2	12.5	31.7	10.7	16.1	4.5
	Total	35.3	16.7	28.1	11.5	34.3	11.7	18.6	4.5

5.3.4 Quantitative Results

The three-way repeated measures ANOVA (analysis of variance) is performed on the per-trial mean of task completion time with three independent variables: 4 Offset Techniques, 3 Box Sizes and 2 Parallax values. The reported F tests use α=.05 for significance and use the Greenhouse-Geisser correction to protect against violation of the sphericity assumption. The post-hoc tests are conducted using Fisher's least significant differences (LSD) pairwise comparisons with α=.05 level for significance.

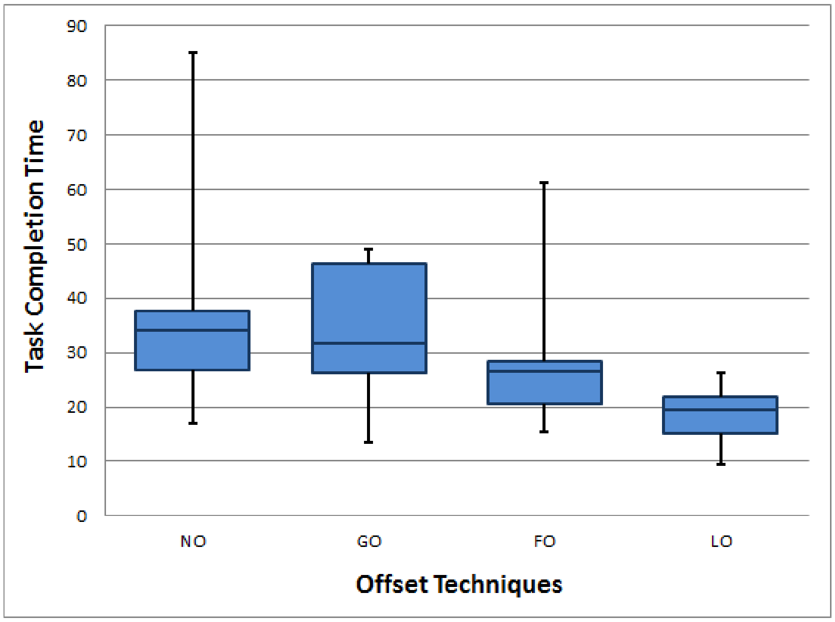


Figure 5.7: Boxplot of task completion time by offset techniques (No Offset, Go-Go Offset, Fixed-Length Offset and Linear Offset).

Table 5.1 shows the average of task completion time and standard deviation by each condition of the three factors. The plus (+) and minus (–) symbols represent the positive and negative parallax conditions respectively. The analysis reveals a significant main effect of Offset Technique on task completion time ($F(3,45)=11.18$, $p<.001$, $\eta_p^2=.427$, see Figure 5.7). Pairwise comparisons show that users perform the 3D docking task faster with LO (M=18.62, SD=4.50) than with NO (M=35.29, SD=16.69, p=.001), FO (M=28.08, SD=11.47, p=.005) and GO (M=34.31, SD=11.68, p<.001). In addition, the completion time of FO is significantly faster than NO (p=.035). However, task completion time of GO is not significantly different from NO (p=.760) or FO (p=.051). These results support our hypotheses (*H1*, *H2* and *H3*) in that the linear offset technique has the best performance of all, which indicates the linear offset technique has its advantages over other offset techniques for the navigation task. On the other hand, GO did not outperform NO by providing a quadratic offset to the virtual cursors while FO and LO did.

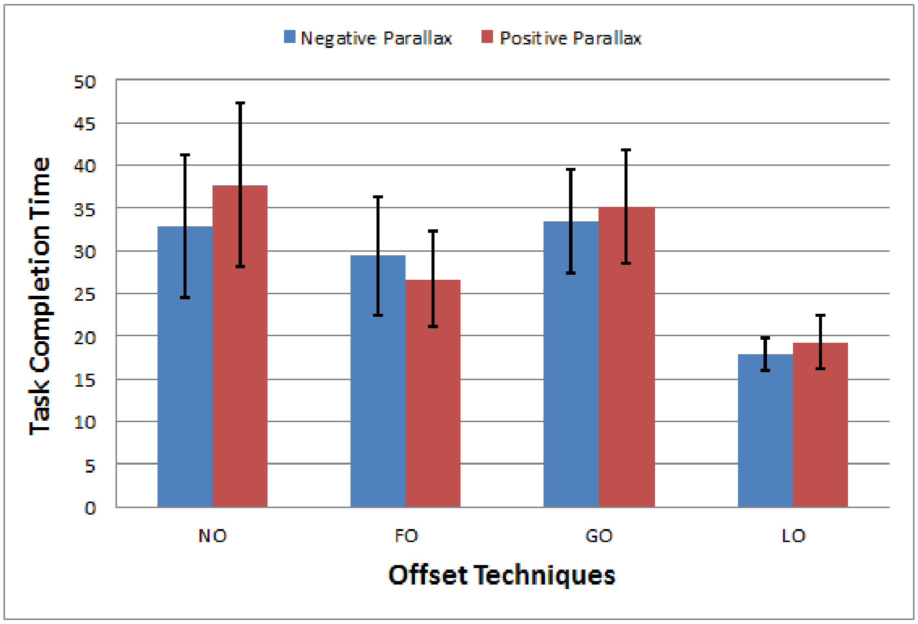


Figure 5.8: Task completion time by offset technique and parallax. The error bars represent the 95% confidence interval.

The main effect of Docking Box Size is significant ($F(2,30)=57.58$, $p<.001$, $\eta_p^2=.793$). LSD tests show that the task completion of 100% box size ($M=22.40$, $SD=6.51$) is significantly faster than 25% box size ($M=39.32$, $SD=12.85$, $p<.001$) and 400% box size ($M=25.49$, $SD=8.57$, $p=.046$). Moreover, completion time of 400% box size is faster than 25% box size ($p<.001$). This result indicates that users perform better with 6DOF tasks than with 7DOF and scaling down the virtual scene is easier than scaling the scene up, which is consistent with our prior results of one-handed experiments in Chapter 4.

There is also a main effect of Parallax on task completion time ($F(1,15)=4.54$, $p<.050$, $\eta_p^2=.232$). This result indicates that the task would take shorter time when the docking box's initial position is in the negative parallax area (*M*=28.41, *SD*=8.72) than it is in the positive parallax area (*M*=29.73, *SD*=9.18).

The analysis reveals a significant interaction effect for Offset Technique × Parallax ($F(3,45)=6.04$, $p<.002$, $\eta_p^2=.287$, see Figure 5.8). There is a simple effect of Parallax in NO condition. The completion time with a docking box that is generated at negative parallax (*M*=32.86, *SD*=15.77) is significantly faster than it is at positive parallax (*M*=37.71, *SD*=17.95, *p*=.002). For other offset techniques, the box's initial position does not make a difference on user performance.

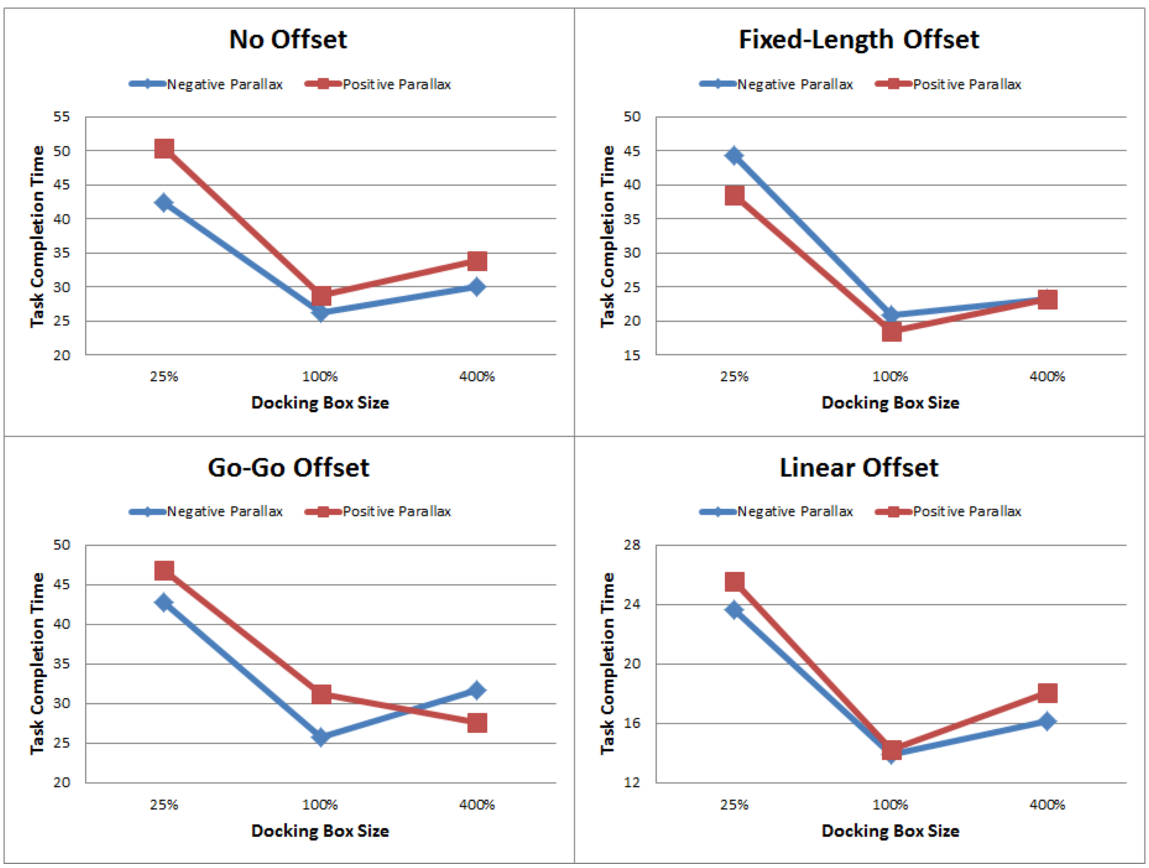


Figure 5.9: Task completion time by docking box size and parallax in different offset conditions. The difference in the shape of four graphs indicates box size × parallax interaction varies significantly across different offset techniques.

The three-way interaction effect of Offset Technique × Box Size × Parallax is also significant ($F(2.86,42.93)=3.21$, $p<.034$, $\eta_p^2=.176$). Pairwise comparisons show that the two-way interaction between Box Size × Parallax varies across different offset techniques, as shown in Figure 5.9. In NO, negative parallax (M=42.37, SD=22.23) is significantly faster than positive (M=50.48, SD=25.29, p=.018) for 25% box size while they do not differ for other sizes. In FO, positive parallax (M=38.56, SD=19.57) is faster than negative (M=44.33, SD=23.91, p=.024) for 25% box size, but no difference exists in other sizes. For GO, negative and positive parallax are different for both 100% box size (Negative: M=25.75, SD=9.28; Positive: M=31.18, SD=14.16, p=.041) and 400% (Negative: M=31.70, SD=10.74; Positive: M=27.60, SD=8.13, p=.027), but not for 25% box size. Lastly, the parallax factor does not differ significantly across box sizes in the LO condition.

5.3.5 Subjective Results

Participants rated arm fatigue level for each offset technique after finishing the corresponding session (on a 7-point Likert scale, 1=not at all to 7=very painful). The Friedman test (with a=.05 level for significance) indicates that there is a significant difference on arm fatigue ratings induced by different offset techniques (χ^2 (3)=8.375, p=.036). Median (IQR) ratings are: NO, 3.5 (2.25 to 5); FO, 3.5 (2 to 4.75); GO, 4 (3 to 5.75); and LO, 3 (2 to 4.75). However, post-hoc analysis conducted using Wilcoxon signed-rank tests with a Bonferroni adjustment for the significant value ($p<.008$) does not show any significant differences among offset techniques. (NO vs. FO: p=.207, NO vs. GO: p=.076, NO vs. LO: p=.209, FO vs. GO: p=.041, FO vs. LO: p=.603 and GO vs. LO: p=.015).

When asked which offset condition helps the most for the trials when the docking box initially appears in the positive parallax area, eleven out of sixteen chose LO, one chose GO, three chose FO and one chose NO. For negative parallax, ten selected LO, two selected FO and four selected NO. When asked to choose the easiest offset technique overall, all sixteen participants preferred the linear offset.

5.3.6 Kinematic Results

Based on our observation during the experiment, users tend to reach out more in No Offset condition due to the absence of arm extension mechanism, even though they can walk around within the space surrounded by the CAVE screens to compensate for this physical limit. Also in the Go-Go Offset condition, because of the nonlinear mapping of motion, users are most likely to spend more time in the fine adjustment phase of the task with both arm stretched out. In hope of verifying these theories, the positions (x, y, z) of each participant's head and both buttonballs are recorded per frame (15Hz) during the trials. To analyze the data, the distance from each hand to head is first calculated per frame and create a histogram that counts the number of samples occurring at fixed range of distance (bin size is 1 inch) over all trials for each offset technique. This is done separately per user per hand. Figure 5.10 shows the histograms for one particular user. The exact shapes vary with users, but this example is fairly typical being uni-modal and bell-shaped, with the left hand histogram shifting to the right on x axis compared to that of right hand because the head tracker is attached to the right side of the stereo glasses. For this user, visual inspection shows similar distribution of distance for both hands.

The bell-like shape of the histograms indicates that most operations fall in a central area and the highest bin shows the most frequent distance with which the user positions her hands relative to her body. This distance varies with the offset techniques. Note that the total area of the histogram will increase with trial duration. In Figure 5.10, NO and GO have greater area correlating with longer average trial times than FO and LO.

Mere visual inspection of the histograms of all users did not provide a strong suggestion of which offset technique might generate higher subjective arm fatigue ratings. As noted, in 5.3.5, subjective ratings were not significantly different across techniques.

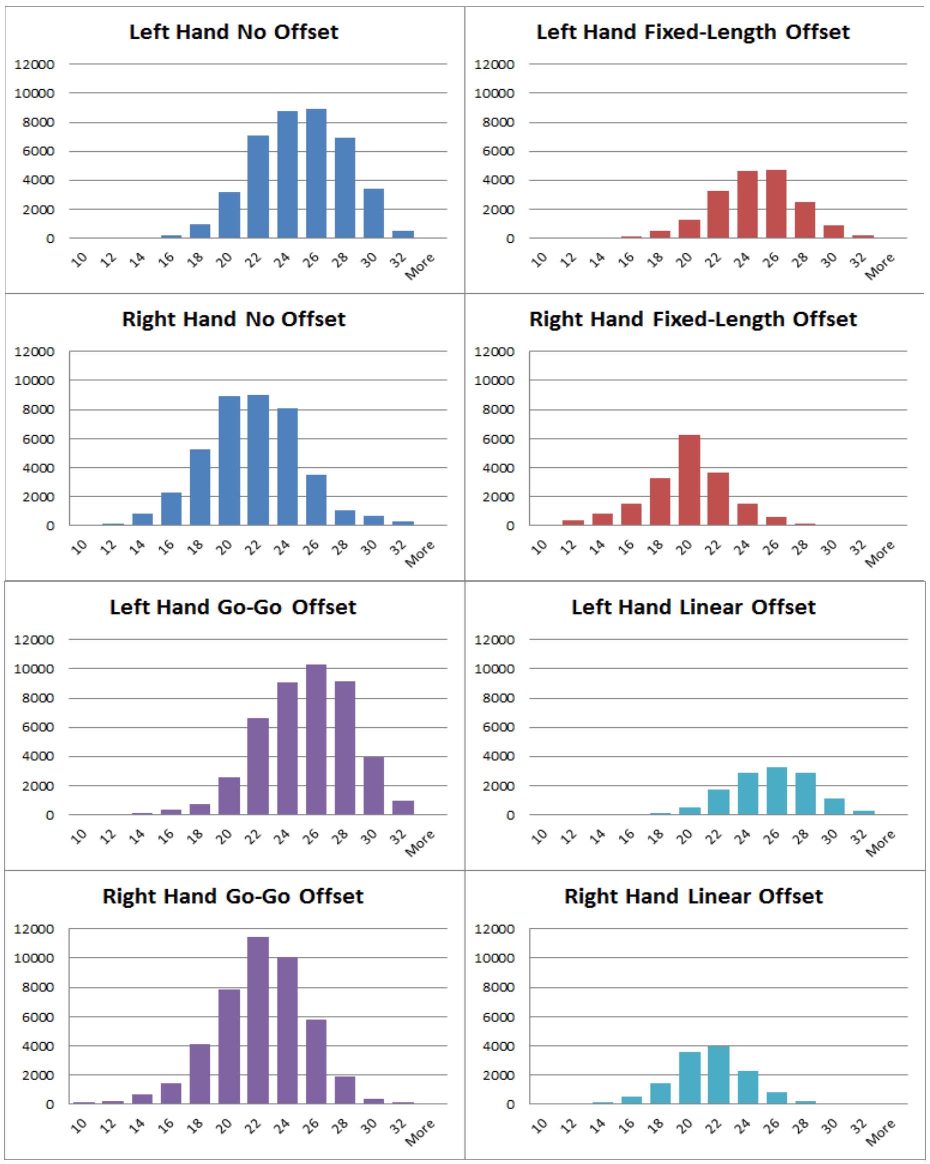


Figure 5.10: Histograms for left and right hands by offset techniques based on distance (measured by inch) from respective input device to the head tracker.

5.4 Discussion

The results of the experiment show that the linear offset technique outperforms no offset, fixed-length offset and Go-Go offset techniques in the 7DOF navigation task, which verifies our hypotheses (*H1*, *H2* and *H3*) that linear offset is more beneficial by dynamically adjusting the offset length based on the hand's relative position to the body. Fixed-length offset is faster than no offset, because it also provides an extension to the user's arm, although rigidly. Some participants claim they feel the same about linear and fixed-length offset in the final questionnaire, but the statistical analysis tells otherwise. Although Fixed-Length Offset is not significantly different from Go-Go Offset on task completion time, there is still a strong tendency (p=.051, Table 5.1). Unexpectedly, there is no difference between Go-Go and no offset, which is counter to hypotheses *H1*. It seems that the nonlinear mapping function of Go-Go has counteracted the arm extension convenience it offers by increasing the sensitivity of the control-display gain factor, despite the fact that Go-Go has the same maximum offset length for the virtual cursor as fixed-length and linear offset techniques do.

The 100% docking box size is significantly faster than 25% and 400%, which represents the difference between 6DOF and 7DOF navigation. The spatial complexity increases as an additional DOF (scale) is introduced to the interaction, resulting in the increase of the completion time. Due to the limitation of the physical space, sometimes 6DOF navigation cannot meet the user's need to explore distant areas or cannot go back to previous locations efficiently in the virtual world, which could possibly be fulfilled by using a 7DOF travel technique. Moreover, an optimal offset technique could further reduce the motion range needed to travel beyond reach.

The Offset Technique × Parallax interaction is mainly caused by different behaviors in negative and positive parallax areas of different offset techniques. Direct manipulation enables easy interaction with objects close to the body. Consequently, in No Offset condition, it is much easier to interact with a docking box that is at the negative parallax than it is at the positive one. However, with an offset presented between the hand and the virtual cursor, task performance is not affected by the parallax factor as much in other offset conditions.

The three-way interaction (Offset Technique × Size × Parallax) reveals the features of different offset technique even more by varying Size × Parallax interaction across offset conditions. Since the 25% box size requires maximum effort to complete the docking task, the parallax factor is significant only for the 25% size in both no offset and fixed-length offset conditions, but in opposite directions. In the no offset condition, the user normally needs to move forward to manipulate the view if the docking box appears far away at the positive parallax, which costs more time than it does when appearing at negative parallax. On the contrary, difficulty emerges when the docking box is close to the user at the negative parallax in fixed-length offset condition. The user could not bring the cursor all the way towards her body because of the fixed offset, which forces the user step backwards which adds to completion time. For Go-Go offset, negative parallax is significantly faster than positive parallax at 100% box size. This is because the quadratic gain factor makes it difficult to place the cursor steadily close to the docking box at positive parallax when no scale operation is required. At 400% box size however, with the help of the gain factor, it is much faster to scale down the scene when the box is at positive parallax than it is at negative parallax. Linear offset compensates for the deficiency of other offset techniques by providing an offset with different lengths based on arm pose, resulting in more consistent performance at both levels of parallax across all box sizes.

Comparing the results with those from one-handed experiment in Chapter 4, the biggest difference lies in the role parallax factor plays. In their one-handed experiment, neither the main effect of parallax nor the interaction that involves parallax is significant, but the results of our experiment indicates that navigation is faster when the docking box shows up at negative parallax than it does positive. I speculate this is caused by the kinematic difference between one-handed and two-handed techniques. Based on our observation, for one-handed interaction, in order to place the cursor within the positive parallax area, the user will need to stretch out the arm. Meanwhile, she will also rotate the torso around the axis of her spine to draw the lead shoulder slightly more forward subconsciously. The user can reach out further by increasing the rotating angle of her shoulders, without having to walk forward. But in two-handed interaction, the center of scale and rotation is determined by the midpoint of the left and right cursors. In order to place this midpoint to the distant object, the user need to stretch both arms, in which case, rotating shoulders does not help reach further. The user will have to walk around to bring the cursors to the desired area.

5.5 Conclusion

A formal user study is presented that investigates the effect of 3D virtual cursor offset for a bimanual interaction technique in a head-tracked, stereoscopic three-side CAVE. Our experiment evaluates four different cursor offset techniques and the results indicate that linear offset technique improves the user’s performance on both 6DOF and

7DOF two-handed navigation tasks in general. Future work will aim to explore the usability of cursor offset techniques in other types of VR displays, such as an HMD, as well as with other types of input devices and technologies.

CHAPTER 6: EVALUATION ON 3D OBJECT SELECTION IN A HEAD-MOUNTED DISPLAY SYSTEM

6.1 Introduction

Virtual Reality is a concept that has been existed for a couple of decades but has not gained much attention from general public until recent advancement of mass-market products, such as the Oculus Rift [93], which has renewed the interest in the design of 3D user interfaces and 3D interaction techniques in immersive virtual environments (IVEs). In such IVEs, head and hands tracking are often enabled by 6DOF tracking technologies which potentially provide natural and direct interaction with objects stereoscopically displayed in the virtual world. Consequently, the user can grab or select virtual objects that are within reach in the similar way to grabbing or selecting in the real world.

However, just like in the CAVE objects displayed behind the screen with positive parallax cannot be reached with direct interaction; it is often not possible to directly select objects that are not located within arm's reach in an HMD system. One possible solution to this problem comes from research on different indirect interaction techniques, such as the Go-Go technique [102], which adds a translational offset to the virtual cursor. These techniques make use of the entire reachable space of the user's arms during interaction with distant objects, which may become exhausting over time [1]. Therefore, it is not clear whether offset based indirect interaction will result in improved overall task performance or not.

Object selection is one of the fundamental tasks in 3DUIs. It is the basis of object manipulation and is usually followed by more complex operations. In this context, virtual hand techniques are considered to be the most natural way of directly selecting virtual objects as they map identically virtual tasks with real tasks, which stands in contrast to indirect selection [21]. However, direct selection of a virtual object in a fully immersive VE significantly differs from selecting an object in the physical world [1]. For instance, users perceive the VE stereoscopically with vergence-accommodation conflicts and often cannot see their body. For indirect selection, the decoupling of visual space from motor space during natural hand interaction may degrade performance due to the kinematics of pointing and grasping gestures in 3D space and the underlying cognitive functions [80].

The results from my previous experiments show that the linear offset techniques outperform other offset techniques for 7DOF navigation tasks in a CAVE system. In this chapter, cursor offset techniques on virtual object selection are evaluated in an HMD environment. Usually interaction techniques are designed and evaluated taking into account only one hardware configuration, due to time, availability and budget limitations. I consider comparing cursor offset techniques in another system setup mainly for three reasons. First of all, the offset techniques are defined in a user-centered reference frame, which is a better fit for an HMD, due to its physical property and 360° horizontal and vertical FOR. Secondly, despite of both being commonly used immersive displays; a CAVE and an HMD have many differences in multiple aspects, which could possibly cause quite different effects on the usability of interaction techniques. Lastly, the advent of affordable VR hardware makes it possible to set up an HMD system at a low cost.

Moreover, the VR industry has made significant improvements to 3D input devices and motion tracking systems in their products. Interacting with natural gestures in 3D space opens up new possibilities for exploiting the richness and expressiveness of the interaction [74], by allowing the user to control multiple DOFs simultaneously. However, interaction with human gestures and poses in the mid-air can be physically demanding and could possibly reduce user satisfaction and performance, which introduces challenges to the design of effective and efficient free-hand interaction techniques. The compatibility between the characteristics of input devices and the ability of the users could impact VR experience heavily. In order to increase the usability of cursor offset, I also compare fine aspects of free-hand interaction such as finger positions and gesture recognition with interaction via hand-held devices for the following experiments in this dissertation.

6.2 Methodology

6.2.1 Fitts' Law

Fitts' law [35] is probably the most frequently used theoretical framework of describing and comparing user performance for different input devices [5] via applying information theory to human behavior. As an empirical model, it has been widely used to describe the tradeoff between speed and accuracy in rapid aimed movements [154] and thus is applied to the pointing and selecting tasks in the UI design.

The model is given by:

$$MT = a + b \cdot ID \qquad \text{(Equation 6.1)}$$

MT is movement time, and *a* and *b* are empirically derived constants via linear regression for different system setups. *ID* is the index of difficulty (in *bits*). The calculation of *ID* is a logarithmic term known as the Shannon formulation [122]. It includes *D* for movement amplitude (distance between targets) and *W* for target width:

$$ID = log_2(\frac{D}{W} + 1) \quad \text{(Equation 6.2)}$$

ID represents the overall task difficulty based on the movement distance and target size. Hence, smaller farther targets are harder to select than closer larger targets. Fitts also proposed to quantify the human rate of information processing in aimed movements with a measure of performance widely known as throughput (*TP*, in *bps*), which is calculated by dividing *ID* (averaged over a block of trials) by the average of *MT* (in seconds):

$$TP = \frac{ID_{average}}{MT_{average}} \quad \text{(Equation 6.3)}$$

Fitts' law can be used as a predictive model through linear regression of measured movement times onto *ID* [133]. It can also be used to compare the effectiveness of input devices via throughput as an index of performance of the human motor system [5]. Its original model applies to rapid aimed movements in a single dimension towards a visible target. But over the years, this law has been extended for more than one dimension [77] [90] [42].

The evaluation of the kinematics and user behavior when selecting virtual objects by hand gestures or arm movements can be roughly divided into two phases [73]: a ballistic phase and a correction phase. In the ballistic phase, the user's attention is focused on the target object and the hand moves quickly to the proximity of the target using proprioceptive motor control. After that, visual feedback is used in the correction phase in order to incrementally reduce the distance between the hand and the object. Fitts' original formula for *ID* can be derived by considering the movements as a series of smaller movements with iterative corrections. MacKenzie et al. [80] showed that Fitts' law holds for the kinematics of arm movements in 3D trajectories, i.e., greater precision in the correction phase is accompanied by earlier deceleration of arm movements.

6.2.2 ISO 9241-9

ISO 9241-9 [54] is an international standard for evaluating the performance and comfort of non-keyboard input devices. When using a Fitts' model, subjects are presented with specific movement tasks to perform over a range of amplitudes and with a set of target widths [122]. The original Fitts paradigm is somewhat antiquated because angle of movement confounds pointing performance. For this reason, ISO 9241-9 recommends using the multi-directional tapping task with a circular arrangement of targets, as shown in Figure 6.1. This arrangement has the advantage of controlling for the effect of direction. The arrows indicate the path subjects follow using the pointing device, to alternating targets clockwise around the circle. Software to capture subject's movement times must graphically indicate which target the subject should proceed to next [122].

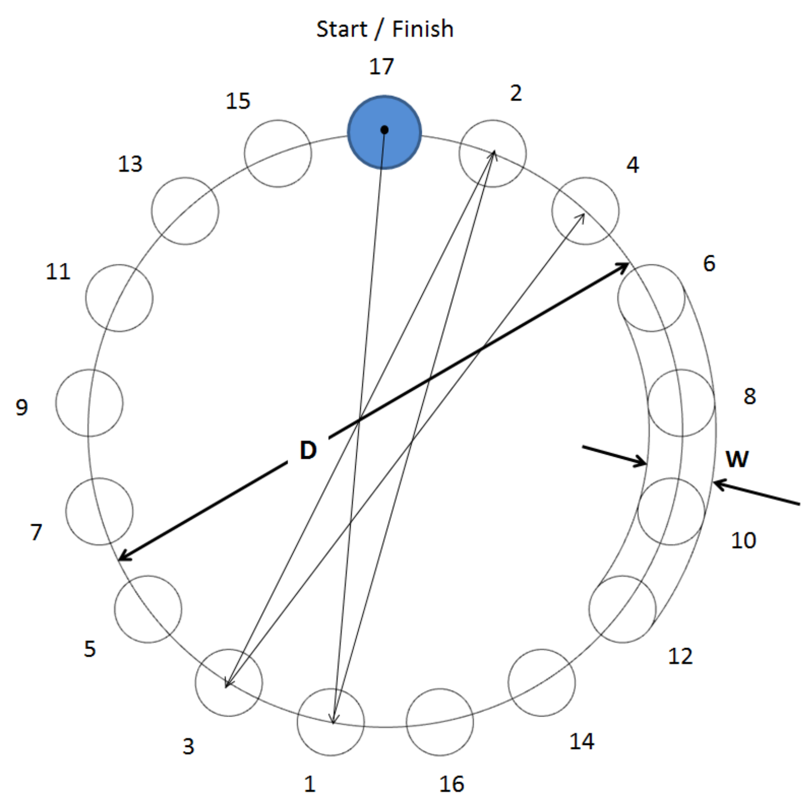


Figure 6.1: Multi-directional tapping task with seventeen targets recommended by ISO 9241-9. Selection starts at the blue target.

Performance is also measured in throughput, but instead of using the presented *ID* (Equation 6.3) in the calculation of throughput, the ISO standard introduces the use of *effective index of difficulty*, ID_e, to accommodate the spatial variability observed in responses:

$$ID_e = log_2(\frac{D_e}{W_e} + 1) \qquad \text{(Equation 6.4)}$$

Here, both movement amplitude *D* and target width *W* are adjusted to account for the task users actually performed, as opposed to the task they were presented [75]. The term D_e represents *effective distance*, which is calculated as the mean movement distance from the start of movement position to the end points for a given condition. W_e is the *effective target width*, computed from the variability of the observed endpoints:

$$W_e = 4.1333 \cdot SD_x \qquad \text{(Equation 6.5)}$$

The term *x* is the projection of the vector from the center of the selected target to the participant's click position on the task axis. The task axis is defined as the vector from the center of previous target to the center of current target. SD_x is the standard deviation of *x* over a block of trials using the same *D* and *W*. Note that *x* can be positive or negative, depending on whether selection is an overshoot or undershoot respectively [92]. This assumes that movement endpoints are normally distributed around the center of target and 4.1333 standard deviations (i.e. 96%) of clicks hit the target. W_e corrects the miss rate to 4%, allowing comparison between studies with different error rates [75].

Effective throughput incorporates speed and accuracy into a single measure and is largely unaffected by speed-accuracy tradeoff [78]. Effective measures are calculated across both hits and misses to better account for real user behavior, and thus enable more meaningful comparison [133]. They also make throughput less sensitive to device characteristics, which is desirable in device comparisons.

6.3 Experiment 4: Object Selection in HMD − Razer Hydra vs Leap Motion

Selection behavior and performance with direct or indirect input techniques have been the focus of several areas of previous research work. Most studies indicate that optimal performance may be achieved when visual and motor space are superimposed or coupled closely [88] [70]. However, to our best knowledge, most previous research that compares direct and indirect selection uses objects at arbitrary positions in the virtual space around the user and none of them follows the ISO 9241-9 task paradigm. Moreover, due to the difference in strength and endurance requirements of the arm and shoulder muscles between a hand-held device and a free-hand gesture capture device, selection performance in immersive virtual environments may be affected by various factors that are related to the ergonomics of direct and indirect interaction techniques. In particular, contributing factors may include interaction duration, hand and arm postures, frequency of movements and comfort [1].

In order to explore these uncertainties, I conduct a user study that compares four virtual cursor offset techniques on 3D object selection tasks following Fitts' model under the ISO 9241-9 standard while using two different input devices in an HMD environment. Twenty-four unpaid volunteers (twenty-two male and two female) are recruited for the experiment, with ages ranging from 23 to 39 (M=28.00). Eighteen of them are from computer science department and the rest six are non-CS major. All of the participants are right-handed while all of them have 20/20 (or corrected 20/20) eye vision and no disability using their arms and fingers. Participants have high daily computer usage (6.67

out of 7). Fourteen of them have experience with 3D UIs, such as Microsoft Kinect, Nintendo Wii Remote, Oculus Rift or HTC Vive.

6.3.1 Apparatus

The experiment uses Oculus Rift DK2, an HMD developed by Oculus VR [93] that provides first person perspective, stereoscopic viewing and integrated 6DOF head tracking and allows the seated user to translate and rotate their head within the tracking volume. Oculus Rift DK2 offers a horizontal FOV of approximately 95° and a vertical FOV of approximately 106° at a resolution of 960 × 1080 per eye with a max refresh rate at 75 Hz. The application is developed in Unity 5 – a cross-platform game development engine from Unity Technology [135], on a high performance computer equipped with Intel Xeon E5 processor (3.00 GHz), 64 GB DDR3 RAM, Nvidia GeForce GTX 1080 graphics card and a Windows 10 operating system.

(A) (B)

Figure 6.2: (A) The Razer Hydra game controller. (B) The Oculus Rift DK2 with Leap Motion VR Setup.

For hand tracking and operations, the performance of object selection is compared between two input devices: Razer Hydra and Leap Motion. Razer Hydra is a 6DOF magnetic controller developed by Sixense Entertainment [107] (Figure 6.2(A)). Leap

Motion is a new 3D motion sensing device designed by Leap Motion [67], which supports natural hand/fingers tracking and gesture recognition. The Leap Motion controller can be mounted in front of the Oculus Rift DK2 headset (Figure 6.2(B)), which creates a true 3D interface that enables users to interact with the virtual world using natural hand gestures. Figure 6.3 shows the environment of our experiment and a subject is doing the selection task using hand gestures.

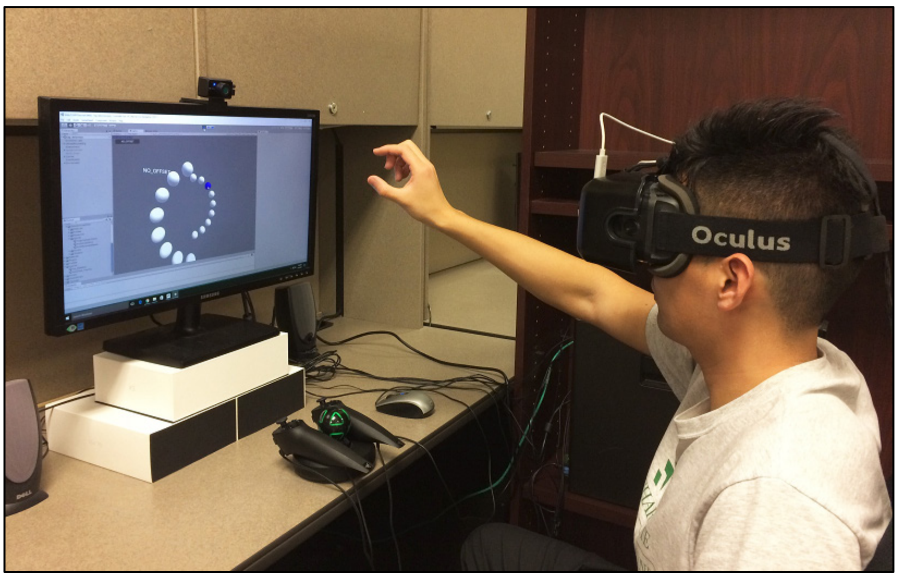


Figure 6.3: A subject doing the selection task with hand gesture.

6.3.2 Experimental Design

3D object selection is more complicated than 2D selection due to a couple of reasons such as the more DOFs added and visual cue conflicts. Although ISO 9241-9 is widely used in 2D pointing research as it allows for direct comparison between studies, there is currently no such standard for 3D interfaces. Our main goal is to determine the effects of virtual cursor offset techniques on 3D selection tasks, hence using a standard or at least close to standard method could highlight the benefits and pitfalls of the technique with consistency, which could provide useful guidelines for future 3D UI design.

Inspired by Teather and Stuerzlinger [132]'s arrangement of the target circle, I also change the depth between subsequent targets. As a result, perspective distortion affects the projection of the targets, which represents a distinctive aspect of 3D selection. The targets layout of the selection task in our experiment is shown in Figure 6.4.

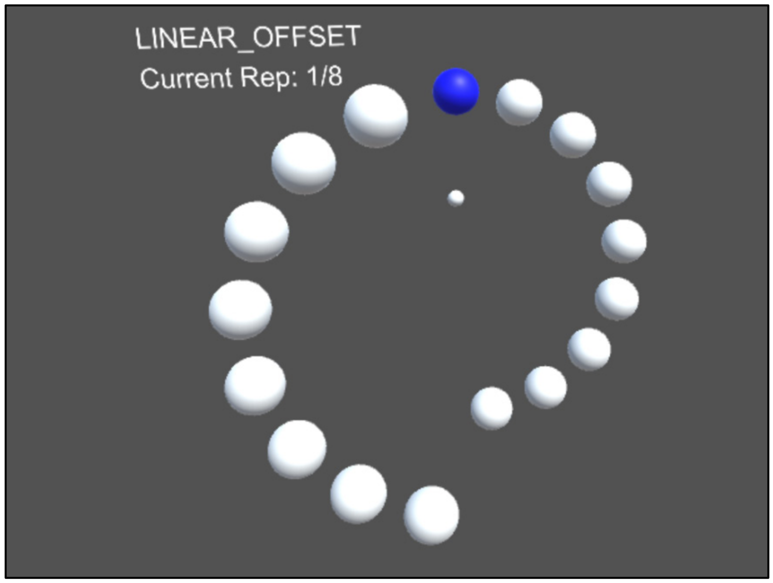


Figure 6.4: Target spheres layout in the experiment.

The targets in the experiment are represented by spheres. There are 17 spheres arranged in a circular layout with varying depths. Spheres on the right side of the circle are presented at a depth of -16cm relative to the depth of the spheres on the left side. The diameter of the layout circle is 30cm and the radius of the target sphere is 2cm. Since the purpose of this study is to test cursor offset techniques across different input devices, only one task condition is used with a nominal difficulty of 3.24 bits.

Instead of changing the index of difficulty of the selection task, I change the depth at which the circle of spheres is presented to the subject, in order to account for the varying ability of different offset techniques in placing the virtual cursor in 3D space. In the experiments in previous chapters, all four offset techniques are tested with the same depth condition− the parallax factor, specifically. That is because those experiments are conducted in a CAVE system, where the user is standing and can walk a few steps in any direction to reach farther objects, especially with No Offset condition. But for experiment using the Oculus Rift DK2, the user is sitting in front of the monitor and the tracking range of DK2 is limited. Therefore, with No Offset, the user cannot select spheres that are out of reach and with Fixed-Length Offset, it is impossible to grab nearby objects without moving backward. Go-Go Offset and Linear Offset are versatile in that they can reach to both close and distant objects.

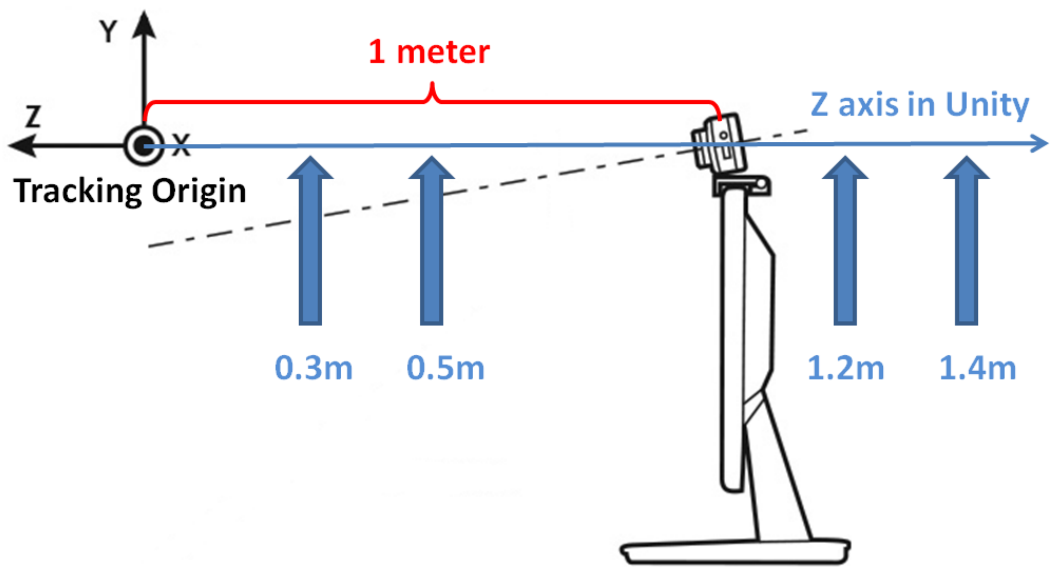


Figure 6.5: The circle of target spheres appears at certain distances to the tracking origin, indicated by the blue arrows.

Based on comfortable range of depths for a user to look at in the Oculus Rift DK2, the circle of spheres is place at four different depths to compare cursor offset techniques on object selection. They are 0.3m, 0.5m, 1.2m and 1.4m away from the tracking origin of the DK2 along z axis in the Unity coordinate system (Figure 6.5). As a result, offset techniques are tested at different depths to make object selection a reasonable task while keeping performance comparable between different offset techniques. Available Offset

Technique × Depth combinations are shown in Table 6.1. There are several reasons why I chose 1.4m as the maximum depth instead of a longer distance such as 10m. The first one is objects will look much smaller if placed that far away from the user. The second reason is that a longer offset distance will make selection difficult in the within reach condition, when using Linear Offset or even Go-Go Offset because of their offset calculation methods. Another reason is that this task is considered as the first step of a 7DOF navigation task for a future study, in which case, it is often sufficient for the offset technique to extend the cursor for a shorter amount of distance, because this will translate to a larger range by multiplying the distance by the view scale. Therefore, users normally need a much smaller motion range of the cursor through navigating in MSVEs.

Table 6.1: Test conditions for Offset Technique × Depth combination.

	30 cm	50 cm	120 cm	140 cm
No Offset (**NO**)	✓	✓	×	×
Fixed-Length Offset (**FO**)	×	×	✓	✓
Go-Go Offset (**GO**)	✓	✓	✓	✓
Linear Offset (**LO**)	✓	✓	✓	✓

Each sequence of trials begins with the participant clicking on the top blue sphere in the layout circle with the virtual cursor, which is always rendered as a small white 3D ball (Figure 6.4). With Razer Hydra, the virtual cursor appears at the controller's tracked position and selection is confirmed by clicking the trigger button. For Leap Motion, the cursor is placed at the palm position of the user's dominant hand and "clicking" action is achieved by doing the "pinch" gesture. To successfully select a target, the virtual cursor has to be inside the blue sphere when the clicking action happens. Otherwise, this attempt is classified as a miss and will be recorded. At all times, there will be only one sphere highlighted blue as the *current target*. When the participant confirms the selection, if it is a success, the current target will turn green for 0.2s and then turn right back to white. In the meantime, the next sphere to be selected will turn blue. The targets are highlighted in the order specified by the ISO 9241-9 standard. If the participant clicks outside the sphere, the current target will turn red for 0.2s to alert the participant and turn back to blue for further selection. The participant has to successfully select the current target before moving to the next one.

The experiment is a complete within-subjects design with three main independent variables: offset technique, input device and selection depth. A sequence of trials (one sphere circle) consists of 16 target selections (see Figure 6.4), and will be repeated twice at each depth. (Logging begins with the first selection; hence data are not collected for the top target.) Since at each depth, only three cursor offset techniques are available for comparison (Table 4), each participant will need to complete 768 trials (2 input devices × 4 depths × 3 offset techniques × 2 repetitions × 16 selections). During the experiment, the current offset technique and progress are displayed at the top left area of the circle, as shown in Figure 6.4. The maximum length of offset added to the virtual cursor is one meter, which is always the case for Fixed-Length Offset technique. For Go-Go Offset and Linear Offset, parameters are set accordingly to ensure the participant can reach the same position with the arm fully stretched out. The order of offset technique is counterbalanced between all subjects using a Latin square. Within each offset technique condition, the depth repetitions are entirely randomized.

The dependent variables are movement time, error rate and effective throughput. The calculation of movement time and throughput is introduced in 90. Error rate is calculated by dividing the number of missed attempts by total attempts.

Our primary hypotheses for the experiment are:

H1: When the circle of spheres are within reach (at 0.3m and 0.5m), selection is faster and error rate is lower with NO than with GO and LO, because the participant can count on proprioception to select with NO and the visual conflict is the least.

H2: When the targets are out of reach (at 1.2m and 1.4m), LO performs better than FO and GO. This is because LO adjusts offset dynamically based on the arm motion while FO has a rigid offset and GO's quadratic gain factor causes overshoot.

H3: Overall, Razer Hydra is expected to have a better throughput than Leap Motion due to the unstable detection of gesture when the hand moves out of the optimal tracking area of Leap Motion.

6.3.3 Procedure

Upon arrival, each subject is asked to sign the informed consent form followed by a short pre-questionnaire. Then the subject is briefed with the purpose of the study and is introduced to the VE and input devices. To make sure the subject is able to distinguish 3D objects within the HMD, a customized eye chart is placed at a proper distance to the camera within the VE and the subject is asked to read out loud the letters on the eye chart to the last level that is discernible with HMD's resolution; in our case, it is approximately 20/120 visual acuity when the Snellen chart is placed at the focal depth of the DK2.

The subject can proceed to the next step only if she passes the 3D vision test. Then the experimenter will demonstrate the selection task, especially with the pinch gesture. Thereafter, the subject is given time to get familiar with the hardware in a short training session. During practice, the experimenter remains in the study area to answer any of the subject's questions. The training session lasts approximately for 10 minutes.

The subject starts the actual trials by selecting the topmost sphere in the circle to trigger the timer. Half of the subjects start the selection tasks using Razer Hydra while the other half using Leap Motion. After they finish all four sessions (each session using a different offset technique) with one input device, they switch to the other device to do the same four sessions. The subjects are instructed to select the target spheres as quickly and accurately as possible. They can take a short break between circles or sessions. After each session, the subject is asked to rate arm fatigue level. At the end of the experiment, the subject fills out a post-questionnaire regarding subjective preference of the cursor offset techniques and input devices, as well as how the selection depth affects the task.

6.3.4 Quantitative Results

For each dependent variable, results are analyzed with two three-way repeated measures ANOVA tests on within reach condition (0.3m and 0.5m) and out of reach condition (1.2m and 1.4m) respectively, at the 5% significance level. Degrees of freedom are corrected using Greenhouse-Geisser adjustments to protect against violation of the sphericity assumption. The post-hoc tests are conducted using Fisher's least significant differences (LSD) pairwise comparisons with α=.05 level for significance.

6.3.4.1 Movement Time

Statistical results for movement time are reported in Table 6.2. Note that the set of offset techniques varies at two depth levels. For within reach condition, it contains NO, GO and LO while for out of reach condition, it has FO, GO and LO. This remains the same in the Movement Time, Error Rate and Throughput analysis.

Table 6.2: Significant effects on movement time for both within reach and out of reach conditions.

	Within Reach			**Out of Reach**		
Factor	***F***	***p***	η_p^2	***F***	***p***	η_p^2
Offset	$F(2,46)=27.128$	<.001	.541	$F(1.45,33.24)=17.355$	<.001	.430
Device	$F(1,23)=12.223$	=.002	.347	$F(1,23)=30.855$	<.001	.573
Offset×Device	—	—	—	$F(1.52,34.94)=9.468$	=.001	.292
Offset×Depth	$F(2,46)=7.751$	=.001	.252	$F(1.48,33.95)=9.364$	=.002	.289

In the within reach condition, the main effect of Offset Technique is significant. Pairwise comparisons show that selecting nearby targets with NO (*M*=1.80, *SD*=0.33) is significantly faster than with GO (*M*=2.59, *SD*=0.76, $p<.001$) and LO (*M*=2.56, *SD*=0.67, $p<.001$). The main effect of Device is also significant, with Razer Hydra (*M*=2.03, *SD*=0.31) faster than Leap Motion (*M*=2.61, *SD*=0.87, $p=.002$).

There is a significant interaction effect for Offset Technique × Depth (Figure 6.6). Post-hoc tests reveal simple effects of offset technique on both levels of depth. When the target circle is at 0.3m, selection with NO (*M*=1.69, *SD*=0.30) is faster than with GO (*M*=2.52, *SD*=0.82, $p<.001$) and LO (*M*=2.74, *SD*=0.74, $p<.001$). At the depth of 0.5m, not only NO (*M*=1.91, *SD*=0.44) is faster than GO (*M*=2.66, *SD*=0.76, $p<.001$) and LO (*M*=2.38, *SD*=0.85, $p<.001$), but LO is also faster than GO ($p=.030$).

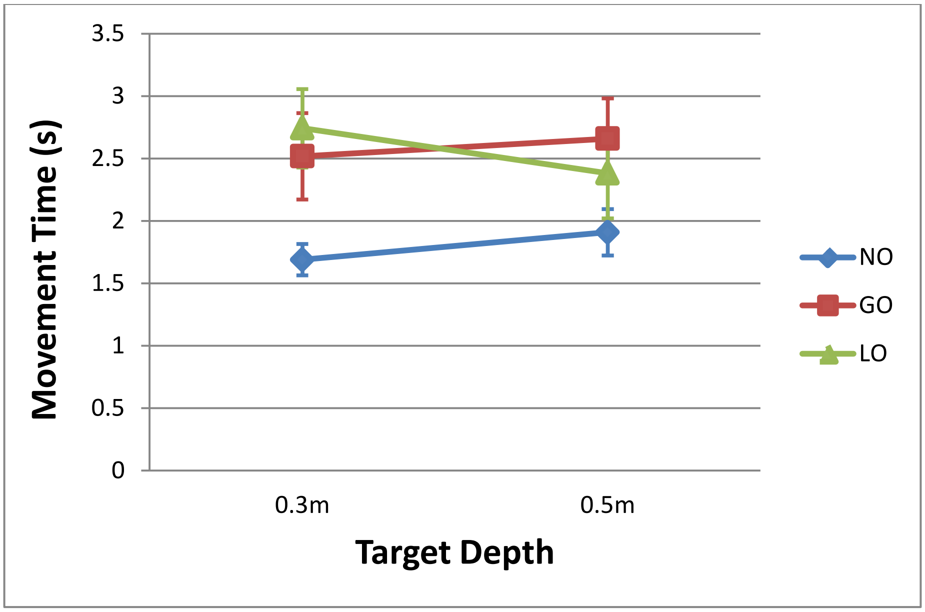


Figure 6.6: Movement time by offset technique and selection depth in the within reach condition. Error bar represents 95% confidence interval.

Similarly, in the out of reach condition, the main effects of Offset Technique and Device are both significant. For offset techniques, LO (*M*=2.92, *SD*=0.76) is faster than FO (*M*=4.78, *SD*=2.30, *p*<.001) and GO (*M*=4.97, *SD*=1.64, *p*<.001). Selection with Razer Hydra (*M*=3.36, *SD*=1.05) is significantly faster than with Leap Motion (*M*=5.09, *SD*=1.85, *p*<.001).

The interaction effect between Offset Technique and Device is significant (Figure 6.7). There are simple effects of offset on both devices. For Razer Hydra, all three offset techniques are different from each other, with LO (*M*=2.50, *SD*=0.76) faster than FO (*M*=4.20, *SD*=1.90, *p*<.001) and GO (*M*=3.39, *SD*=0.99, *p*<.001), and GO faster than FO (*p*=.033). With Leap Motion, only LO (*M*=3.35, *SD*=1.22) is faster than FO (*M*=5.37, *SD*=3.28, *p*=.002) and GO (*M*=6.55, *SD*=2.67, *p*<.001).

There is another significant interaction effect between Offset and Depth, with simple effects of offset at both 1.2m and 1.4m (Figure 6.8). Selection at 1.2m with LO

(*M*=2.80, *SD*=0.87) is faster than with FO (*M*=5.42, *SD*=3.06, *p*<.001) and GO (*M*=4.89, *SD*=1.74, *p*<.001). At 1.4m, LO (*M*=3.05, *SD*=0.84) is faster than FO (*M*=4.15, *SD*=1.77, *p*<.001) and GO (*M*=5.05, *SD*=1.68, *p*<.001), and FO faster than GO (*p*=.025).

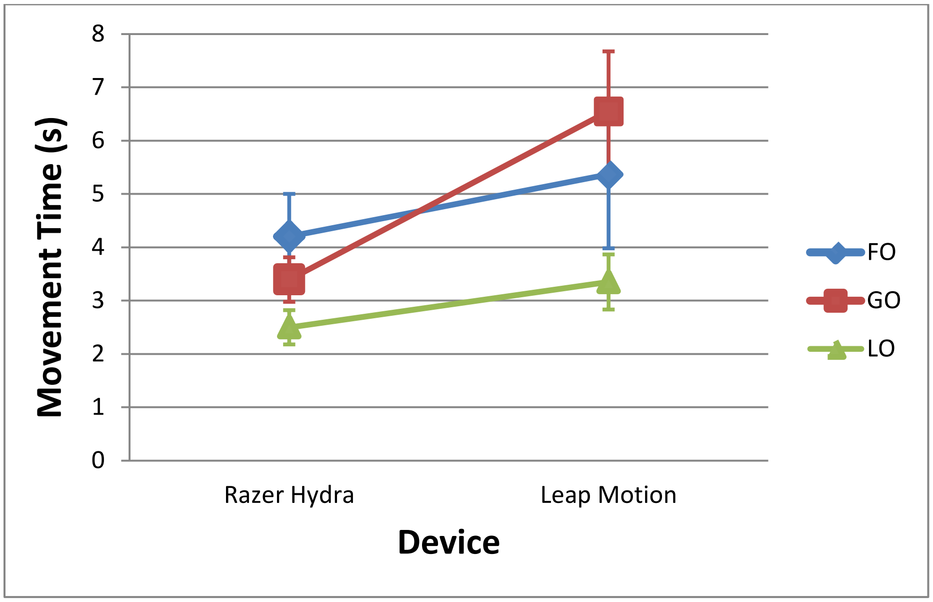


Figure 6.7: Movement time by offset technique and device in the out of reach condition. Error bar represents 95% confidence interval.

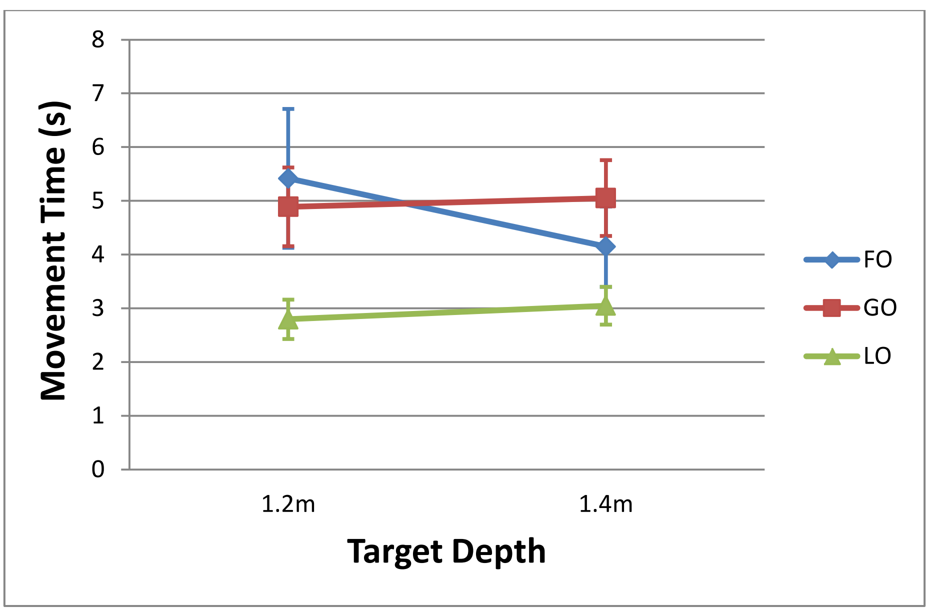


Figure 6.8: Movement time by offset technique and selection depth in the out of reach condition. Error bar represents 95% confidence interval.

6.3.4.2 Error Rate

Overall results for error rate are shown in Table 6.3. Under the within reach condition, the error rates for three offset techniques differ significantly from each other. Pairwise comparisons show that subjects miss fewer targets with NO (*M*=14.4%, *SD*=0.075) than GO (*M*=28.8%, *SD*=0.12, *p*<.001) and LO (*M*=25.0%, *SD*=0.10, *p*<.001). Additionally, using LO is more accurate than GO (*p*=.040). Two input devices also have different error rates, with Razer Hydra providing more accuracy (*M*=19.2%, *SD*=0.087) than Leap Motion (*M*=26.3%, *SD*=0.11, *p*=.001).

Table 6.3: Significant effects on error rate for both within reach and out of reach conditions.

	Within Reach			**Out of Reach**		
Factor	***F***	***p***	η_p^2	***F***	***p***	η_p^2
Offset	*F*(2,46)=36.472	<.001	.613	*F*(2,46)=82.282	<.001	.782
Device	*F*(1,23)=16.280	=.001	.414	*F*(1,23)=27.935	<.001	.548
Offset×Device	*F*(2,46)=25.827	<.001	.529	*F*(2,46)=8.466	=.001	.269
Offset×Depth	*F*(2,46)=12.965	<.001	.360	*F*(2,46)=23.566	<.001	.506
Device×Depth	*F*(1,23)=17.677	<.001	.435	—	—	—

There is a strong interaction effect between Offset Technique and Device (Figure 6.9). Post-hoc tests reveal simple effects of offset on both devices. For Razer Hydra, NO has significantly lower error rate (*M*=7.3%, *SD*=0.068) than GO (*M*=23.6%, *SD*=0.13, *p*<.001) and LO (*M*=26.8%, *SD*=0.11, *p*<.001). While for Leap Motion, GO has higher error rate (*M*=34.0%, *SD*=0.14) than NO (*M*=21.6%, *SD*=0.11, *p*<.001) and LO (*M*=23.3%, *SD*=0.13, *p*<.001).

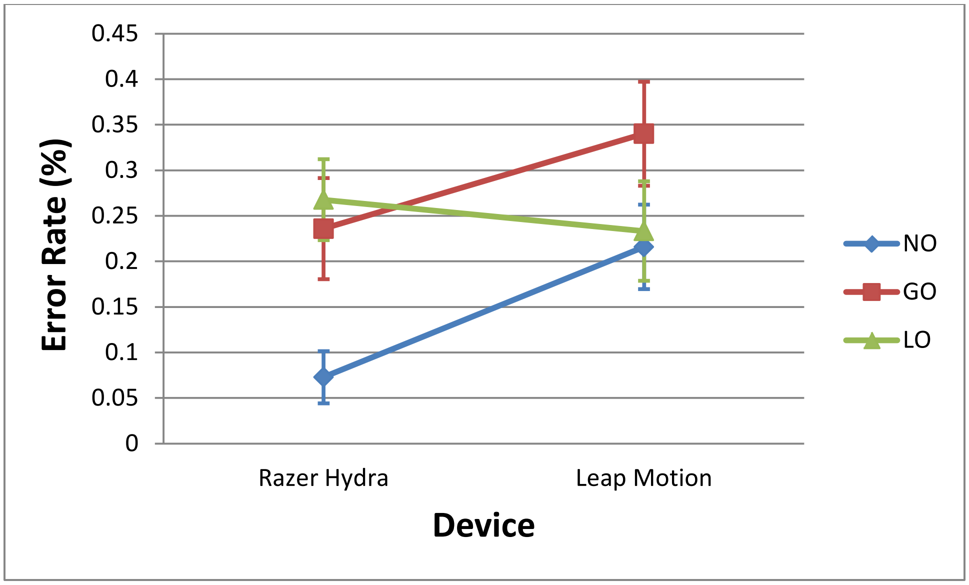


Figure 6.9: Error rate by offset technique and device in the within reach condition. Error bar represents 95% confidence interval.

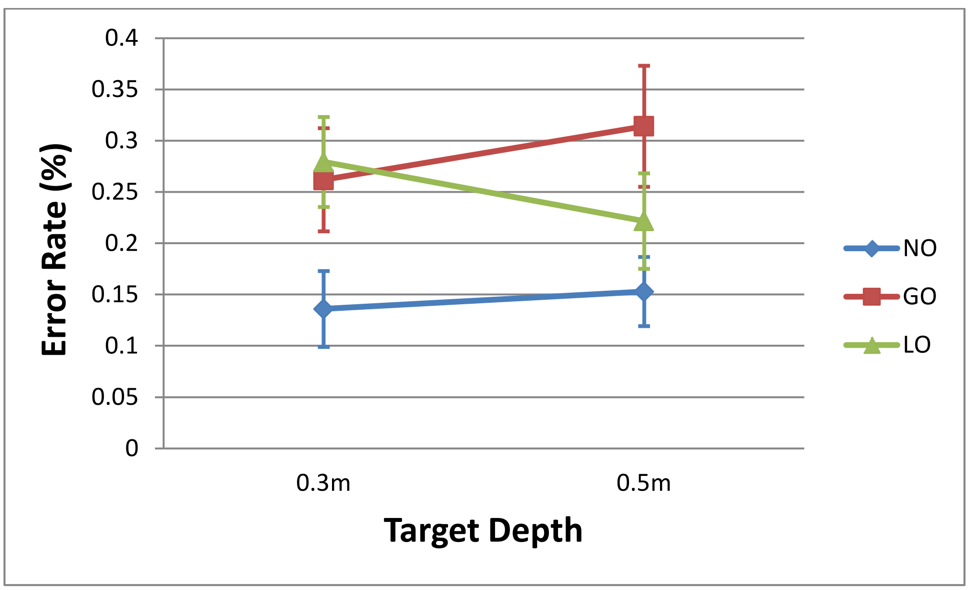


Figure 6.10: Error rate by offset technique and target depth in the within reach condition. Error bar represents 95% confidence interval.

The interaction effect of Offset × Depth is also significant, with simple effects of offset on both depths (Figure 6.10). The error rate at 0.3m is lower with NO (*M*=13.6%, *SD*=0.088) than with GO (*M*=26.2%, *SD*=0.12, $p<.001$) and LO (*M*=28.0%, *SD*=0.10,

p<.001). The error rates of offset techniques at 0.5m are significantly different from each other. NO (M=15.3%, SD=0.080) is lower than GO (M=31.4%, SD=0.14, p<.001) and LO (M=22.2%, SD=0.11, p=.001); GO is higher than LO (p<.001).

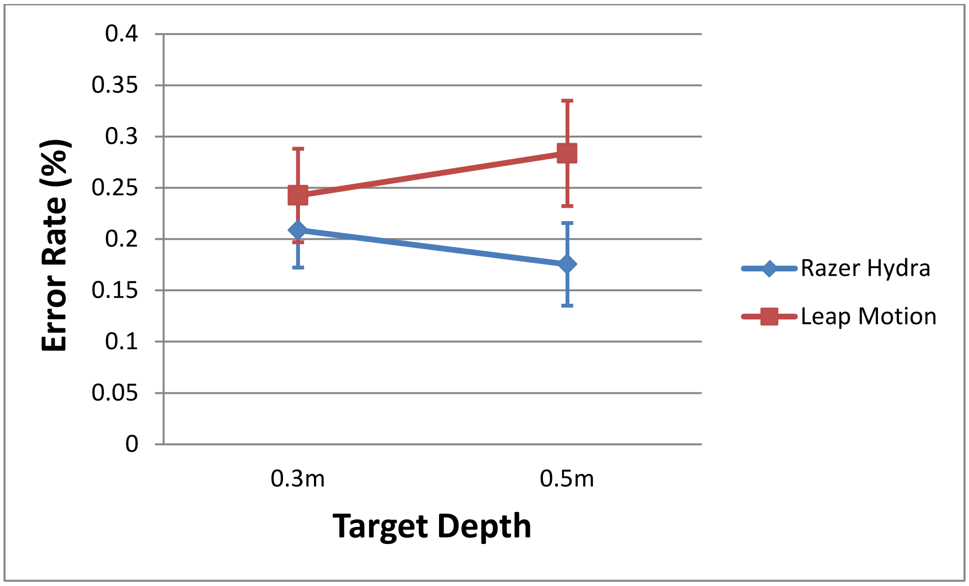


Figure 6.11: Error rate by device and target depth in the within reach condition. Error bar represents 95% confidence interval.

There is another interaction effect between Device and Target Depth (Figure 6.11). The simple effect of device is at 0.5m, where Razer Hydra has a significantly lower error rate (M=17.5%, SD=0.095) than Leap Motion (M=28.4%, SD=0.12, p<.001).

In the out of reach condition, the main effect of offset techniques is significant with all three techniques different from each other. GO has a significantly higher error rate (M=52.9%, SD=0.11) than FO (M=34.1%, SD=0.15, p<.001) and LO (M=29.1%, SD=0.13, p<.001). The error rate of LO is lower than FO (p=.012). The main effect of device is also significant. Selection with Razer Hydra (M=32.5%, SD=0.13) is more accurate than with Leap Motion (M=44.9%, SD=0.13, p<.001).

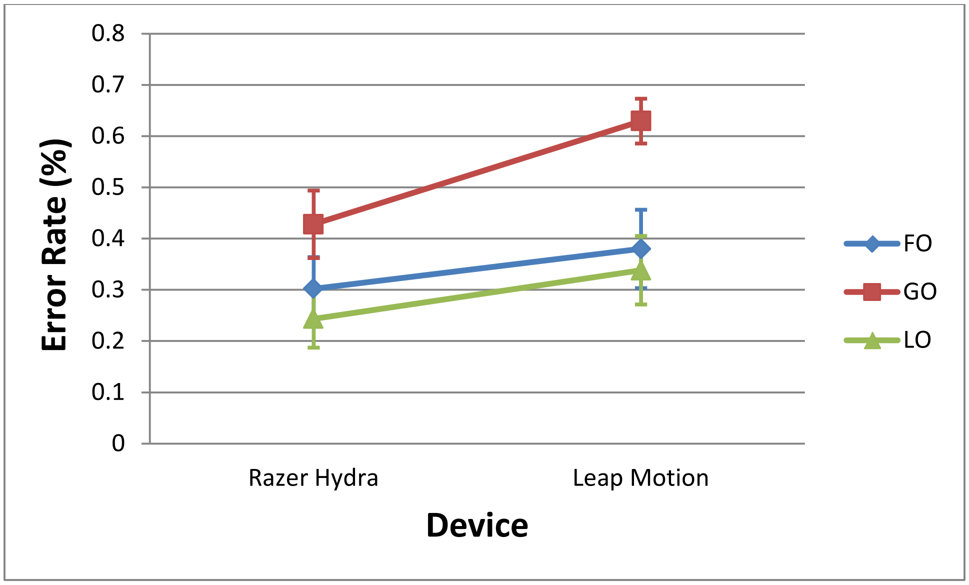


Figure 6.12: Error rate by offset technique and device in out of reach condition. Error bar represents 95% confidence interval.

The two two-way interaction effects associated with Offset Techniques are both significant. For Offset Technique × Device interaction, there are simple effects of offset on both devices (Figure 6.12). With Razer Hydra, the error rate of GO (*M*=42.8%, *SD*=0.16) is higher than FO (*M*=30.2%, *SD*=0.15, *p*<.001) and LO (*M*=24.3%, *SD*=0.13, *p*<.001). The error rate of LO is lower than FO (*p*=.005). For Leap Motion, only GO (*M*=62.9%, *SD*=0.10) is higher than FO (*M*=38.0%, *SD*=0.18, *p*<.001) and LO (*M*=33.8%, *SD*=0.16, *p*<.001); the error rates of LO and FO are not different.

In the Offset Technique × Depth interaction, offset techniques behave differently at the two depths (Figure 6.13). At 1.2m, error rates are significantly different for each offset technique. LO (*M*=26.6%, *SD*=0.12) is lower than FO (*M*=38.3%, *SD*=0.16, *p*<.001) and GO (*M*=51.3%, *SD*=0.11, *p*<.001). And FO is lower than GO (*p*<.001). GO performs the worst at 1.4m, with error rate (*M*=54.5%, *SD*=0.11) lower than both FO (*M*=29.9%, *SD*=0.15, *p*<.001) and LO (*M*=31.5%, *SD*=0.15, *p*<.001).

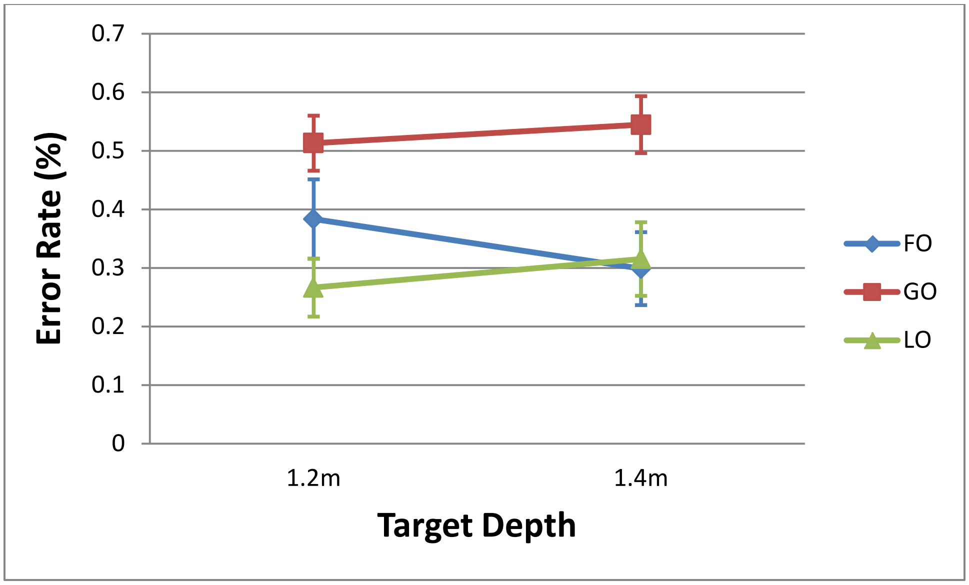


Figure 6.13: Error rate by offset technique and target depth in the out of reach condition. Error bar represents 95% confidence interval.

6.3.4.3 Effective Throughput

Table 6.4: Significant effects on effective throughput for both within reach and out of reach conditions.

	Within Reach			Out of Reach		
Factor	***F***	***p***	η_p^2	***F***	***p***	η_p^2
Offset	$F(2,46)=149.763$	<.001	.867	$F(2,46)=76.239$	<.001	.768
Device	$F(1,23)=31.243$	<.001	.576	$F(1,23)=62.391$	<.001	.731
Offset×Device	$F(2,46)=6.692$	=.003	.225	$F(2,46)=9.590$	<.001	.294
Offset×Depth	$F(2,46)=20.845$	<.001	.475	$F(2,46)=21.690$	<.001	.485
Device×Depth	$F(1,23)=4.641$	=.042	.168	—	—	—

Table 6.4 gives the overall results for effective throughput of the experiment. When the target circle is within reach, the main effect of Offset Technique is significant. NO has higher throughput (M=2.39, SD=0.36) than both GO (M=1.55, SD=0.34, p<.001)

and LO (M=1.59, SD=0.35, p<.001). There is a significant main effect of Device on throughput as well, with Razer Hydra providing higher effective throughput (M=2.06, SD=0.29) than Leap Motion (M=1.63, SD=0.43, p<.001).

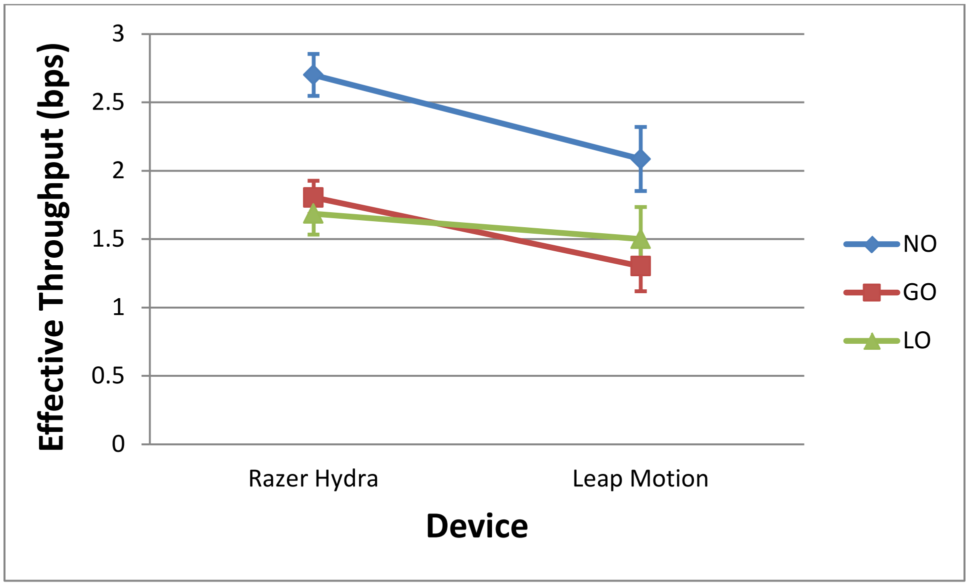


Figure 6.14: Effective throughput by offset technique and device in the within reach condition. Error bar represents 95% confidence interval.

All the two-way interactions in the within reach condition are significant on effective throughput. For the interaction between Offset Technique and Device, there are simple effects of offset on both devices (Figure 6.14). With Razer Hydra, NO has higher throughput (M=2.70, SD=0.37) than both GO (M=1.80, SD=0.29, p<.001) and LO (M=1.69, SD=0.36, p<.001). With Leap Motion, NO is higher (M=2.09, SD=0.55) than GO (M=1.30, SD=0.43, p<.001) and LO (M=1.50, SD=0.55, p<.001). LO is also higher than GO (p=.042).

For Offset Technique × Target Depth interaction, pairwise comparisons show that offset techniques provide significantly different throughput at both depths (Figure 6.15). At the depth of 0.3m, the throughput with NO (M=2.50, SD=0.37) is higher than both GO

(*M*=1.63, *SD*=0.36, *p*<.001) and LO (*M*=1.44, *SD*=0.31, *p*<.001). And GO is higher than LO (*p*=.005). Furthermore at 0.5m, the throughput with NO (*M*=2.29, *SD*=0.43) is still higher than GO (*M*=1.48, *SD*=0.37, *p*<.001) and LO (*M*=1.75, *SD*=0.44, *p*<.001). But LO has a higher throughput than GO (*p*=.001) at this depth.

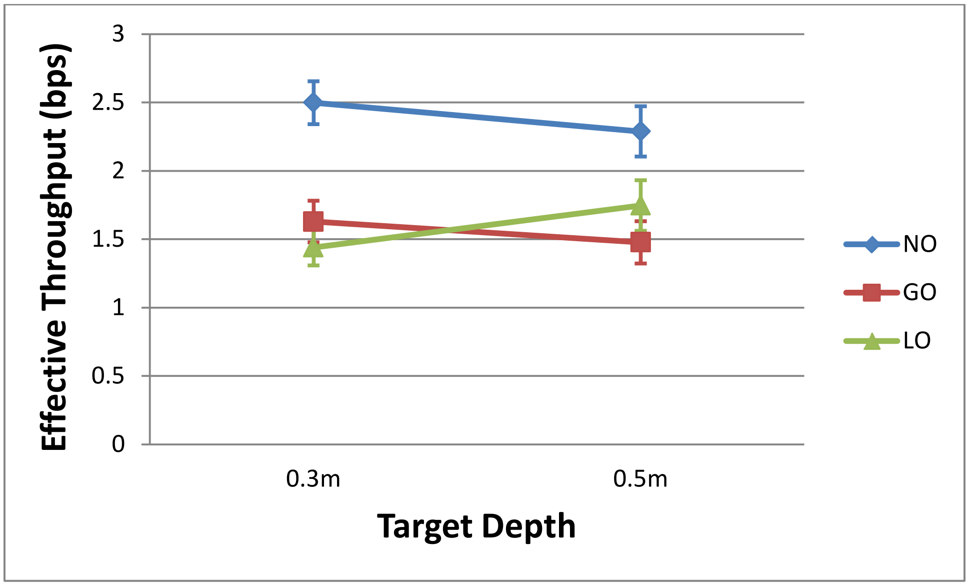


Figure 6.15: Effective throughput by offset technique and target depth in the within reach condition. Error bar represents 95% confidence interval.

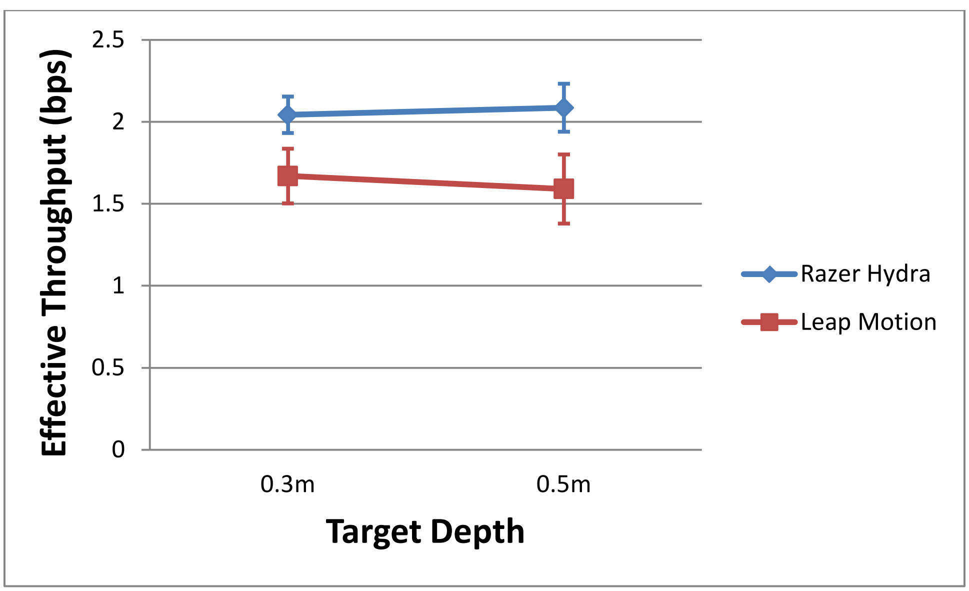


Figure 6.16: Effective throughput by device and target depth in the within reach condition. Error bar represents 95% confidence interval.

The interaction between Input Device and Target Depth is significant (Figure 6.16). Razer Hydra and Leap Motion have different throughput at both depths. Post-hoc tests indicate that Razer Hydra has higher throughput (M=2.04, SD=0.26) than Leap Motion (M=1.67, SD=0.39, p<.001) when selecting targets at 0.3m. And Razer Hydra (M=2.09, SD=0.35) also outperforms Leap Motion (M=1.59, SD=0.50, p<.001) at the depth of 0.5m.

In the out of reach condition, there is a main effect of Offset Technique on throughput. Pairwise comparisons show that selection with LO (M=1.42, SD=0.34) provides higher throughput than FO (M=0.91, SD=0.31, p<.001) and GO (M=0.86, SD=0.21, p<.001). The main effect of device is also significant, with Razer Hydra having a higher effective throughput (M=1.30, SD=0.29) than Leap Motion (M=0.83, SD=0.29, p<.001).

There is a significant interaction effect between Offset and Device (Figure 6.17). Post-hoc tests reveal simple effects of offset technique on both devices. With Razer Hydra, the throughput of LO (M=1.64, SD=0.41) is higher than FO (M=1.04, SD=0.31, p<.001) and GO (M=1.21, SD=0.31, p<.001). And GO is higher than FO (p=.025). For selection with Leap Motion, all three offset techniques differ significantly from each other, with LO (M=1.20, SD=0.47) higher than FO (M=0.78, SD=0.36, p<.001) and GO (M=0.52, SD=0.20, p<.001); FO higher than GO (p<.001).

The interaction of Offset Technique × Target Depth is also significant (Figure 6.18). Simple effects of offset technique exist at both depths. At the 1.2m depth, LO (M=1.49, SD=0.37) provides higher throughput than FO (M=0.80, SD=0.30, p<.001) and GO (M=0.90, SD=0.24, p<.001). At the depth of 1.4m, LO still has higher throughput

(*M*=1.35, *SD*=0.35) than FO (*M*=1.03, *SD*=0.34, *p*<.001) and GO (*M*=0.83, *SD*=0.20, *p*<.001). And at the same depth, the effective throughput of FO is higher than GO (*p*=.002).

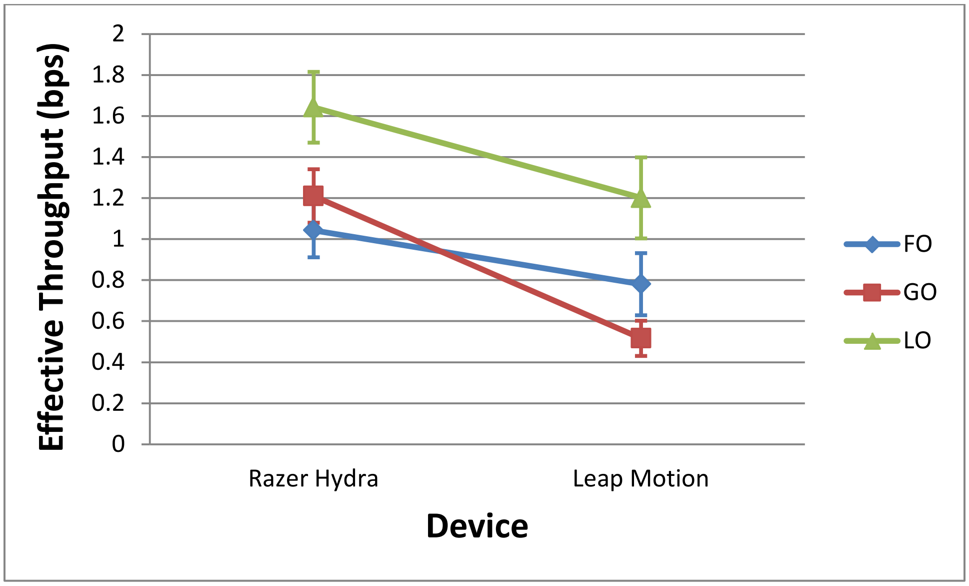


Figure 6.17: Effective throughput by offset technique and device in out of reach condition. Error bar represents 95% confidence interval.

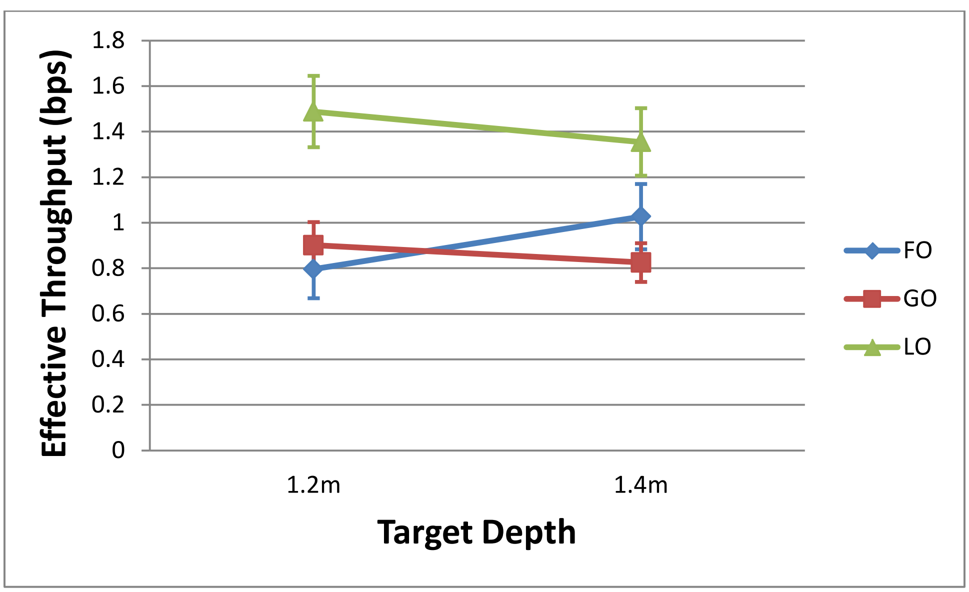


Figure 6.18: Effective throughput by offset technique and target depth in the out of reach condition. Error bar represents 95% confidence interval.

6.3.5 Subjective Preferences

Participants rate arm fatigue level after each offset technique session on a 7-point Likert scale from 1 ('Not at all') to 7 ('Very Painful') separately under each device. For Razer Hydra, the Friedman test (with α=.05 level for significance) reveals a main effect of offset technique on fatigue ratings ($\chi^2(3)=23.778$, $p<.001$). Median (IQR) ratings are: NO, 3 (2 to 5); FO, 5 (4 to 6); GO, 5 (4 to 5.75); and LO, 5 (4 to 6). Wilcoxon signed-rank tests with the Bonferroni adjustment for the level of significance (α=.0083) show that compared to NO, users feel more arm fatigue with FO ($Z=-3.667$, $p<.001$), GO ($Z=-3.417$, $p<.001$) and LO ($Z=-2.730$, $p=.006$). With Leap Motion, the Friedman test indicate that there is a significant difference on the fatigue ratings induced by different offset techniques ($\chi^2(3)=20.713$, $p<.001$). Median (IQR) ratings are: NO, 4 (3 to 6); FO, 5.5 (4 to 6); GO, 5.5 (4.25 to 7); and LO, 5 (4 to 6). Wilcoxon signed-rank tests show that selection with NO causes less arm fatigue, compared to FO ($Z=-2.693$, $p=.006$) and GO ($Z=-3.681$, $p<.001$), but not LO.

In the post-questionnaire, participants are asked to rate both offset techniques and input devices based on how easy it is to select the targets using a 7-point Likert scale from 1 ('Very easy') to 7 ('Very difficult'). There is a main effect of offset technique ($\chi^2(3)=36.014$, $p<.001$). The Median (IQR) ratings are: NO, 2 (1 to 3); FO, 4.5 (4 to 6); GO, 4 (3.25 to 5.75); and LO, 3 (3 to 4). Post-hoc tests show that users rate NO to be easier than FO ($Z=-3.938$, $p<.001$), GO ($Z=-3.961$, $p<.001$) and LO ($Z=-3.228$, $p=.001$); and LO easier than FO ($Z=-3.516$, $p<.001$). The ratings for devices are also significantly different, with Razer Hydra being rated easier than Leap Motion ($Z=-3.383$, $p<.001$). Median (IQR) rating for Razer Hydra is 2 (1.25 to 3) and for Leap Motion is 4 (3 to 5).

When asked which offset technique (NO/GO/LO) was the easiest to complete the selection task when the circle of spheres are within reach, twenty subjects chose NO, one chose GO and three chose LO. For the condition when the targets are out of reach, six answered FO, four answered GO and fourteen answered LO. As to their preferences for input devices, eighteen subjects preferred Razer Hydra while six preferred Leap Motion.

6.4 Discussion

The main effects and interactions are quite consistent across movement time, error rate and effective throughput. Overall, Razer Hydra outperforms Leap Motion for object selection at all four depths, which supports hypothesis *H3*. Users are able to select the targets with less movement time and lower error rate with Razer Hydra, which leads to a higher effective throughput than Leap Motion. Based on our observation, this is largely due to the difference between what “clicking” means for the two devices. Leap Motion requires the user to confirm the selection with an explicit pinching gesture while for Razer Hydra, the user just clicks a button. The sensors of Leap Motion controller have a field of view of about 150° and the effective range extends from approximately 0.03 to 0.6 meters above the device which makes gesture detection less accurate when users moves their hands out of the optimal tracking frustum. Even within the tracking range, as the arm reaches out, the tracking data deviates from the real gesture more often which could result in a high frequency of false negative detection. This sort of unresponsive input control makes the selection more difficult and can frustrate users a great deal. On the other hand, it’s almost impossible for the system to miss a button click which makes Razer Hydra more responsive and easier to control.

There is a significant difference in the offset technique performance as well. When the circle of spheres is at the depths of 0.3m and 0.5m, No Offset outperforms Go-Go Offset and Linear Offset techniques with less movement time, lower error rate and higher throughput, which verifies hypothesis *H1*. This conforms to Mine's conclusion [88] that offsets reduce performance compared to interaction with objects collocated with the user's hand in HMD environments. Direct manipulation remains the most natural way of interacting with objects nearby by exploiting proprioception when the physical world is not visible. In addition, the error rate with LO is lower than GO. The nonlinear mapping of hand motion of Go-Go technique can cause overshooting when the user tries to select the target, and thus increasing the error rate.

One of my contributions is to compare different indirect manipulation methods when the virtual object is out of the reaching distance. When the targets are located at the depths of 1.2m and 1.4m, Linear Offset technique provides the better performance than Fixed-Length Offset and Go-Go Offset techniques, which supports hypothesis *H2*. To successfully select the target, the user needs to place the virtual cursor inside the sphere, but the quadratic gain factor of the Go-Go technique makes it difficult for the user to engage in precise adjustment in the correction phase of object selection. Even though FO is linear mapping, the user's ability to place the virtual cursor can still be compromised if the distance between the object and the user is shorter than the length of the fixed offset. From this perspective, the Linear Offset technique can help the user precisely place the cursor when navigation in the HMD is not possible and the object is located beyond physical limits.

Interaction effects complicate the analyses of the experimental factors but also expose their characteristics at certain levels. When the targets are within reach, as the depth moves from 0.3m to 0.5m, LO performs better with reduced movement time (Figure 6.6), dropped error rate (Figure 6.10) and increased effective throughput (Figure 6.15). However, the opposite happens to NO and GO. Therefore, it is reasonable to predict that as the depth increases further, this trend will continue until the object reaches certain depth, where LO will outperform NO. But this particular depth may never be confirmed since it is quite possible that this depth is out of human arm's reaching limit. Therefore, even before hitting that point, the user would have no other option than to use offset techniques to select the object. The devices also behave differently at the two depths. As the target moving from 0.3m to 0.5m, users miss fewer spheres selecting with Razer Hydra (Figure 6.11) and perform better (Figure 6.16). The performance of Leap Motion goes the other way, which indicates that it's only optimal to use gesture tracking when the hands can be close enough to the HMD.

When the targets are completely out of reach, switching from Razer Hydra to Leap Motion causes performance to drop significantly. And GO is affected by this device change more than FO and LO (Figure 6.7, Figure 6.12, Figure 6.17). When the targets moving from 1.2m to 1.4m, it's interesting to see that the performance of FO increases as opposed to the decreasing performance of GO and LO. I suspect this results from the rigid offset added to the cursor in FO condition, in which case, it's always one meter in length. When the targets are at 1.2m, the user may need to retract their hands in order to place the cursor inside the sphere due to the length of the offset, which is counterintuitive for selecting distant objects. But this retraction would go away when selection happens at 1.4m. Users can fully extend their arm under the instinct to reach farther.

There are also some lessons learnt from this study. In the feedback gathered from the post-questionnaire, some participants mentioned when the targets are far away, it is sometimes hard to tell whether the virtual cursor is inside the sphere or not. This issue can be fixed by changing the current target's color once the cursor is inside. The reason this was not did not added to the initial experimental design is that the current target is already highlighted and the subject is expected to confirm the selection without using additional visual cues. During data processing, there were some trials with extremely high movement time and error rate in the Leap Motion group, which are caused by frequent false negative pinch detection of the device. This phenomenon can be eliminated by changing the way of "clicking". The subjects can confirm each selection by pressing a key on the keyboard or clicking a button on a separate device with their non-dominant hand to avoid any jittering caused by "clicking" with their dominant hand or to avoid the false negative caused by detection problems. In a recent study [74] that investigates direct selection in Oculus DK1 HMD and employs this design for selection confirmation, the average error rate during their experiment is 8.8% (*SD*=11.3%) and the average throughput is 1.98bps (*SD*=0.44bps). In comparison to their results, the overall error rate for Razer Hydra in our experiment is 25.8% (*SD*=13.0%) and the error rate for Leap Motion is 35.6% (*SD*=15.0%). The overall throughput for Razer Hydra is 1.68bps (*SD*=0.48bps) and the throughput for Leap Motion is 1.23bps (*SD*=0.54bps). The major difference is their much lower error rate resulting from the avoidance of using the same hand for both pointing and clicking.

6.5 Conclusion

This chapter presents an experiment that evaluates the performance of cursor offset techniques on virtual object selection using ISO 9241-9 standard within an HMD environment. Two types of input devices are also compared in the study. The results show that direct manipulation is most efficient when the target is within reach and the performance of Linear Offset has a positive correlation with selection depth. In distant areas where the target is out of reach, Linear Offset outperforms Fixed-Length Offset and Go-Go Offset on all three metrics. Overall, Razer Hydra controller provides better and more stable selection performance than Leap Motion.

CHAPTER 7: EVALUATION ON 7DOF OBJECT MANIPULATION IN HMD ENVIRONMENTS

7.1 Introduction

The previous chapter provides empirical evidence of the performance of varied cursor offset techniques on virtual object selection using the Oculus Rift DK2 headset, as well as the Razer Hydra and Leap Motion controller. Some very recent work [22] [3] [41] has also explored the usability of these new low-cost commercial devices in creating fully immersive virtual environments. However, this type of VR systems still can be restricted to relatively small space in front of the sensor due to the user's movement control. The limitations imposed on 3D user interfaces by the new generation of VR hardware need to be further studied.

The selection task only requires the user to control 3DOFs — x, y and z directions, whereas 3D object manipulation complicates the interaction by adding three rotation DOFs, which inevitably increases the necessary number of maneuvers and cognitive load as well. This, in turn, contributes to the complexity of user interface design in immersive VEs. Whether cursor offset techniques can help reduce this complexity and, if so, which kind of cursor offset helps users the most, are the research questions that drive my work in this chapter.

Direct object selection and manipulation is limited to certain area around the user within arm's reach. Without further traveling, additional virtual cursor position mapping scheme is needed to extend the control space. Nowadays, both one-handed and two-handed user interfaces are widely used in a variety of VR systems. Therefore, I conduct two experiments to evaluate the effects of cursor offset techniques on object manipulation tasks using unimanual and bimanual interfaces respectively. The results reveal interesting implications of varying cursor offset for both interfaces.

(A)

(B)

Figure 7.1: The study environment. (A) A subject doing the task with a pair of Razer Hydra controllers. (B) A subject doing the same task with hand gestures.

7.2 Evaluation

Two formal user studies are carried out with the purpose of exploring the usability of cursor offset on 3D object manipulation in an HMD environment. Different interaction techniques are employed for each experiment, with one being a one-handed interface and the other being two-handed. The rest aspects of the experimental design for the two studies are the same. The hardware used for both experiments is the same as in Section 6.3.1. Figure 7.1 shows the two types of input device being used by a subject.

7.2.1 Experimental Design

Adding a translational offset to the 3D virtual cursor increases the user's ability to interact with the VE. However, due to the inherent difference between an HMD system and a CAVE system, the frame of reference for the cursor offset is changed as well. Since the Leap Motion controller is attached to the front of the DK2, users often need to raise both hands in order for the gestures to be detected. It is natural for the user to look in the direction pointing to the object, which is also where their hands are at. Therefore, it makes more sense to place the origin of the reference frame for offset close to the user's head than it does at the user's chest, as in the CAVE experiments (Figure 3.2).

Figure 7.2 illustrates the calculation of the cursor offset. For Leap Motion, the direction of the offset vector is from the HMD's main camera position to the user's palm position; while for Razer Hydra, it is from the camera to the controller's position. The magnitude of the offset vector is determined by methods described in Chapter 3. Since navigation is disabled in our experiments, the scope of the area that the user can reach into varies based on the type of the offset technique. To accommodate each offset technique's ability of placing the virtual cursor, the space where the objects can show up is divided into two areas. In Figure 7.3, the blue rectangle represents the within reach area that compares No Offset, Go-Go Offset and Linear Offset on object manipulation; whereas the purple rectangle represents the out of reach area that compares Fixed-Length Offset, Go-Go Offset and Linear Offset. The order of magnitude for the dimension of each area is consistent with the depths of previous selection experiment.

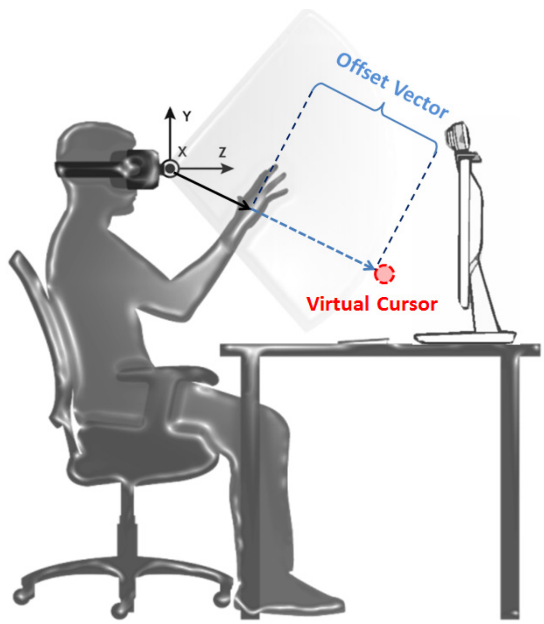


Figure 7.2: Illustration of the offset vector calculation.

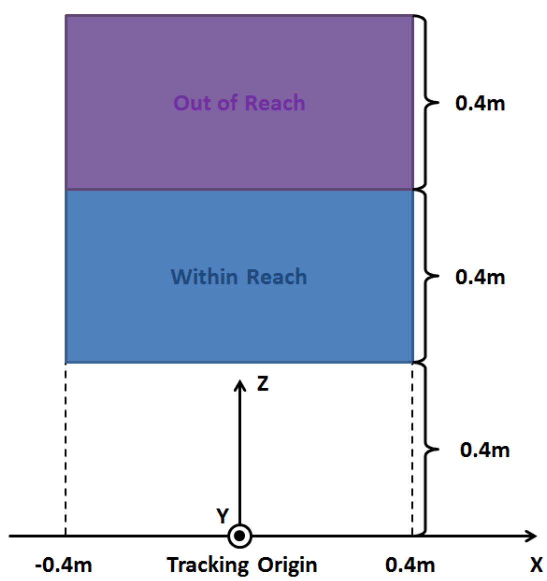
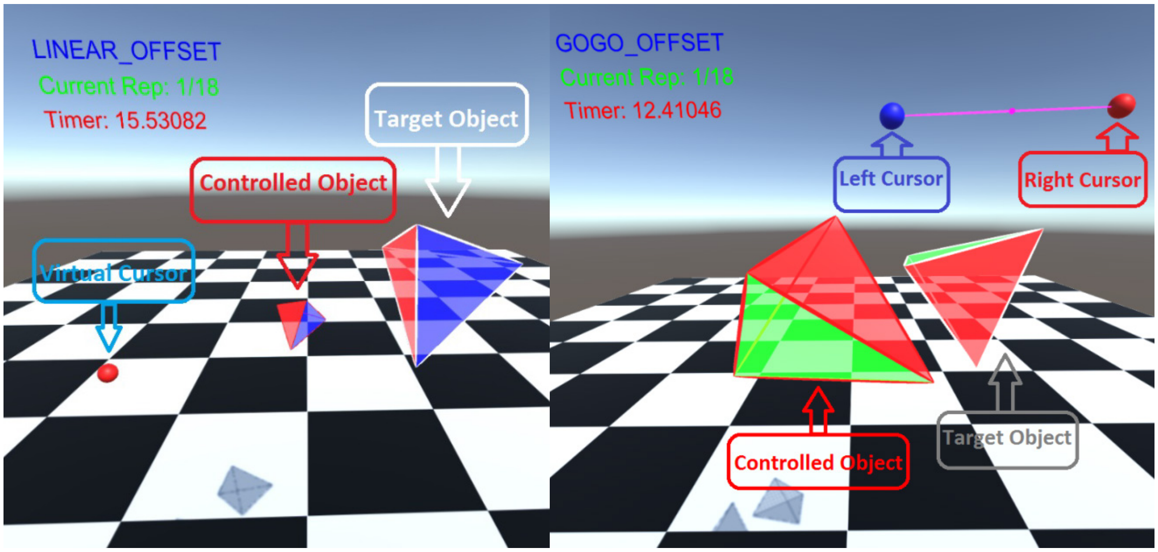


Figure 7.3: Top down view of the work space under Unity coordinate system.



(A) (B)

Figure 7.4: (A) A screen capture of the one-handed interface. (B) A screen capture of the two-handed interface.

The same docking task described in [157] is used for both studies, except that the scale of the object also varies, making it a 7DOF object manipulation task. Screenshots taken from each interface are shown in Figure 7.4. The VE consists of a checker-board ground plane and two transparent colored tetrahedra. The size of the plane is 10m × 10m. At the beginning of each trial, a target tetrahedron with a white outline will show up at a random position within certain area above the ground place with a random orientation and will remain static during the trial. The edge length of the target tetrahedron is 0.32m. The controlled tetrahedron with a red outline will appear at the center of either the blue or the purple area shown in Figure 7.3 and blink twice to remind the subject. The controlled tetrahedron comes in one of three sizes: 25%, 100% and 400% of the target tetrahedron's size and its orientation is also randomly generated. Again, for No Offset, both of the tetrahedra only show up in blue area shown in Figure 7.3 and for Fixed-Length Offset, they only show up in purple area. But for Go-Go Offset and Linear Offset, both objects can appear in either area. The tetrahedron has a different color at each face. The task is to align the two tetrahedra with the correct size and orientation.

In the one-handed interface, either a blue sphere or a red sphere is displayed in the scene as the virtual cursor, representing left palm or right palm position (Figure 7.4(A)). The user needs to select the controlled object by placing the cursor inside and then can change its position, orientation and size. With Razer Hydra controller (Figure 7.5), button 1 engages 6DOF translation and rotation and button 2 engages rate controlled scaling. These are the same button choices as used for experiments in Chapter 4. The center of rotation and scale is the virtual cursor. With Leap Motion, two distinct gestures are defined for object manipulation (Figure 7.6). With the cursor being inside the object, extending all five fingers to form a completely open hand gesture engages 6DOF transformation; pinching gesture with other three fingers fully extended engages scaling.

Figure 7.5: Button usage on the Razer Hydra Controller.

For the second experiment, the same bimanual 7DOF interaction technique — Spindle+Wheel [30] as used in Chapter 5, is used for object manipulation. In this mode, the blue sphere and red sphere are both shown for visual feedback of hand positions (Figure 7.4(B)). A thin cylinder is drawn between the virtual cursors with a small pink sphere rendered at the midpoint acting as the precise center of rotation and center of scale. This geometry also provides additional occlusion cues when the user tries to select the object. To engage control of the object, first the pink sphere needs to be inside the tetrahedron. And then with Razer Hydra, the user needs to press and hold the trigger button with either hand. For Leap Motion, the same open hand gesture as in one-handed mode is used to engage Spindle+Wheel mode. Spinning the non-controlling hand around the pink axis rotates the object around the 'spindle' axis. Figure 7.7 shows a subject doing the task with hand gestures in the bimanual interface.

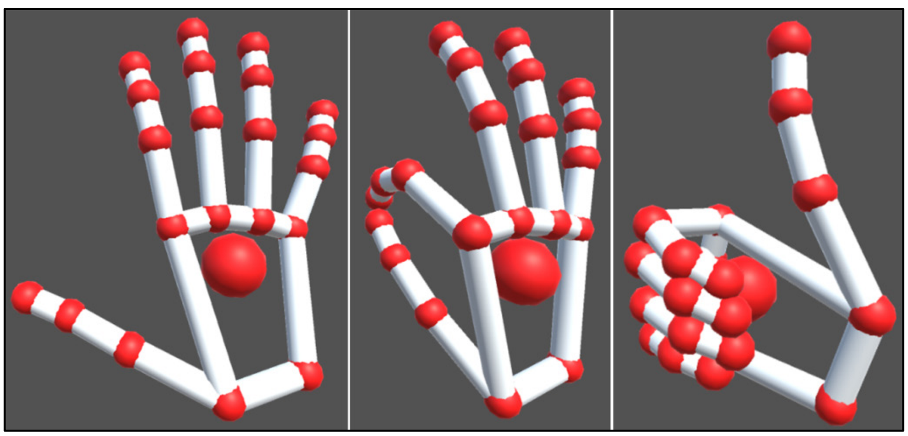


Figure 7.6: Gestures used in the experiment with the Leap Motion controller. From left to right: open hand, pinching and thumb up.

An information panel is placed at the top left area of the virtual scene, providing the user with information about current offset mode, current progress of the trials and how much time has passed since the start of the current trial (Figure 7.4). When all the distances between the corresponding vertices of the target tetrahedron and the controlled tetrahedron are within a certain tolerance, the outline of the controlled tetrahedron turns green and the timer stops. The tolerance value is calculated based on Zhai [157]'s method, which is 20% of the radius length of the tetrahedron's circumsphere. Then the user can proceed to the next trial by either clicking the 'start' button on Razer Hydra controller or performing a thumb up gesture with Leap Motion, which also triggers the timer reset.

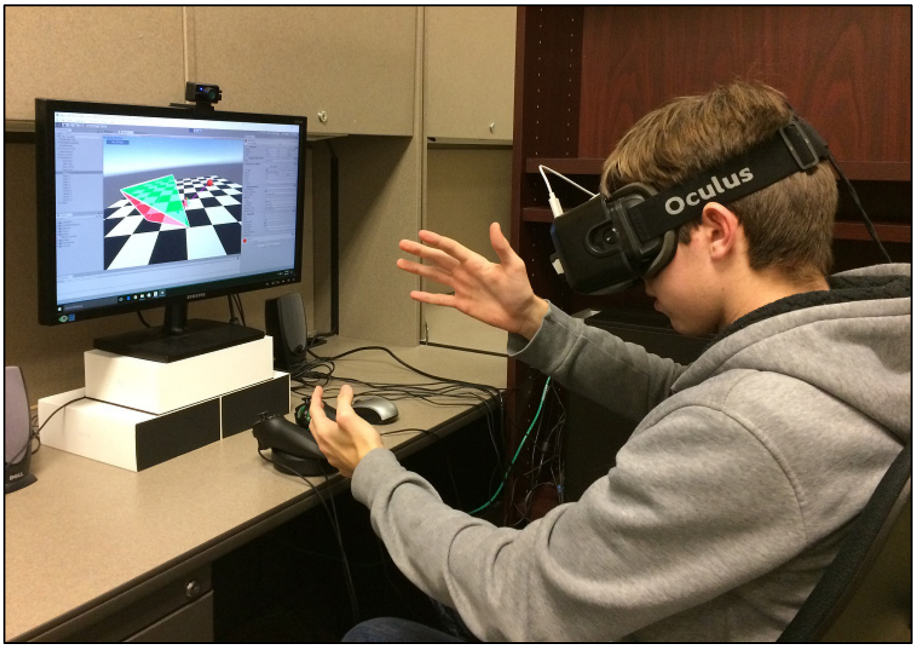


Figure 7.7: A subject is doing the task using hand gestures in bimanual mode.

7.2.2 Procedure

Upon arrival at the study room, each subject is asked to sign the informed consent form followed by a short pre-questionnaire. Then the experimenter introduces the virtual environment and input devices to the subject and demonstrates the docking process with the chosen interaction technique. Next, the subject can put on the HMD and start a short training session where she learns how to use both input devices and get familiar with the task. During the practice, the experimenter remains in the study area to act as a guide and to answer any questions from the subject. The training session lasts for approximately ten minutes.

After the subject becomes familiar with the interface and operations, she can start the actual trials. There are totally eight sessions in each experiment, resulting from the combination of four offset techniques and two input devices. The subject is instructed to align the two tetrahedra as quickly as possible, but no time limit is imposed. She can also take a short break between trials and sessions. The application records task completion time and number of clutched maneuvers for each trial. One clutched maneuver is a button click on the Razer Hydra or a hand gesture registered with the Leap Motion. At the end of each experiment, the subject fills out a post-questionnaire regarding their subjective preferences on the offset techniques and input devices, as well as opinions on how the size and position of tetrahedra affect the interaction.

7.2.3 Experiment 5: Unimanual 7DOF Object Manipulation in HMD

Four virtual cursor offset techniques are compared in this experiment: No Offset (NO), Fixed-Length Offset (FO), Go-Go Offset (GO) and Linear Offset (LO), using a one-handed hand-object manipulation technique. Sixteen unpaid volunteers are recruited (eleven male and five female) for the study, with ages ranging from 21 to 39 (M=27.44). Twelve of them are from computer science department and the rest four are non-CS major. All of the participants are right-handed and all have 20/20 (or corrected 20/20) eye vision with no disability using their arms and fingers. The daily computer usage among participants is high (6.63 out of 7). Eleven of them have experience with 3D UIs, such as Microsoft Kinect, Nintendo Wii Remote, Oculus Rift or HTC Vive.

The experiment is a complete within-subjects design with three main independent variables: offset technique, input device and object size. For each object size, the docking task will be repeated three times. Since within each area (colored rectangle in Figure 7.3) only three offset techniques can be compared, each participant will need to complete 108 trials (2 input devices × 2 areas × 3 offset techniques × 3 object sizes × 3 repetitions) in total. The maximum length of offset added to the virtual cursor is one meter, which is always the case for Fixed-Length Offset technique. For Go-Go Offset and Linear Offset, parameters are set accordingly to ensure the participant can reach the same position with the arm fully stretched out. The order of offset technique presented to each participant is counterbalanced. Under each offset technique condition, half of the subjects start with the Razer Hydra controller and then switch to Leap Motion and the other half complete the tasks using the reversed device order.

The primary hypotheses for experiment 5 are:

H1: When the object is within reach, docking performance with NO is expected to be better than GO and LO, because direct manipulation affords the least cognitive load to the user.

H2: LO will outperform FO and GO on the docking performance when the object is out of reach because it is more adaptive to the user's need.

H3: On the whole, LO is faster than GO due to the quadratic gain factor of GO.

H4: Razer Hydra is expected to have a better performance overall than Leap Motion, since hand gesture detection can be unstable.

7.2.3.1 Quantitative Results

Three-way repeated measures ANOVA tests are performed on the per trial mean of task completion time and number of clutched maneuvers for within reach condition and out of reach condition respectively, at the 5% significance level. Degrees of freedom are corrected using Greenhouse-Geisser method to protect against violation of the sphericity assumption. The post-hoc tests are conducted using Fisher's least significant differences (LSD) pairwise comparisons with α=.05 level for significance.

7.2.3.1.1 Within Reach Condition

Table 7.1: Significant effects on task completion time and number of controls in within reach condition.

Factor	Task Completion Time			Number of Clutched Maneuvers		
	F	***p***	η_p^2	***F***	***p***	η_p^2
Offset	*F*(2,30)=3.606	=.039	.194	*F*(2,30)=11.374	<.001	.431
Device	*F*(1,15)=49.615	<.001	.768	*F*(1,15)=55.120	<.001	.786
Size	*F*(2,30)=80.557	<.001	.843	*F*(2,30)=56.518	<.001	.790
Offset×Device	*F*(2,30)=10.348	<.001	.408	*F*(2,30)=11.131	<.001	.426
Offset×Size	*F*(4,60)=5.004	=.002	.250	*F*(2.73,40.92)=3.543	=.026	.191
Device×Size	*F*(2,30)=19.695	<.001	.568	*F*(2,30)=18.792	<.001	.556
Offset×Device ×Size	*F*(4,60)=4.333	=.004	.224	*F*(4,60)=3.483	=.013	.188

Significant effects of the experiment factors in the within reach condition are reported in Table 7.1. There is a significant main effect of Offset Technique on completion time, with LO (*M*=22.43, *SD*=6.43) faster than GO (*M*=26.85, *SD*=8.98, *p*=.038). There is no difference between NO and LO though. Task completion time between two devices is significantly different, with Razer Hydra (*M*=17.85, *SD*=3.48) faster than Leap Motion (*M*=31.75, *SD*=10.18, *p*<.001). The factor of Object Size is also significant. Participants finish tasks much faster with 100% of target tetrahedron size (*M*=16.28, *SD*=4.60) than 25% size (*M*=30.11, *SD*=8.13, *p*<.001) and 400% size (*M*=28.01, *SD*=7.86, *p*<.001). But there is no difference between 25% and 400%.

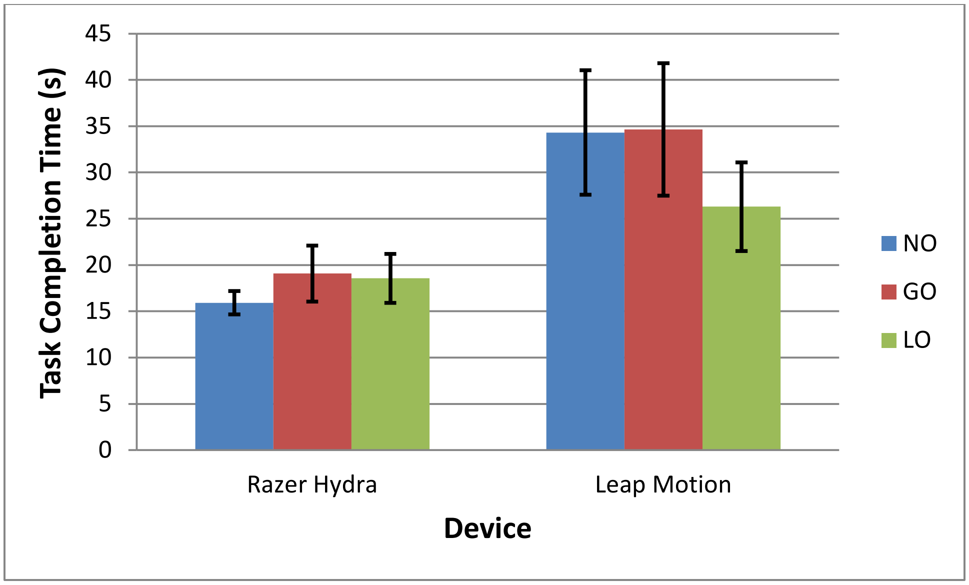


Figure 7.8: Task completion time by offset technique and device in within reach condition. Error bar represents 95% confidence interval.

There is a significant interaction effect for Offset Technique × Device (Figure 7.8). Post-hoc tests reveal that with Razer Hydra, NO (*M*=15.92, *SD*=2.36) is faster than GO (*M*=19.07, *SD*=5.69, *p*=.032). But with Leap Motion, NO is not different from GO. And LO (*M*=26.30, *SD*=8.99) is faster than both NO (*M*=34.31, *SD*=12.60, *p*=.006) and GO (*M*=34.64, *SD*=13.43, *p*=.019) using hand gestures.

The interaction between Offset Technique and Object Size is significant (Figure 7.9). Post-hoc tests show that simple effects of offset technique only exist at 25% and 400% levels, but not at 100%. In trials where the controlled tetrahedron initially appears in 25% size, docking with LO (*M*=26.43, *SD*=7.84) is faster than with GO (*M*=33.19, *SD*=12.47, *p*=.044). And when the object shows up in 400% size, LO (*M*=23.86, *SD*=6.53) is faster than both NO (*M*=30.00, *SD*=10.79, *p*=.019) and GO (*M*=30.19, *SD*=9.79, *p*=.007).

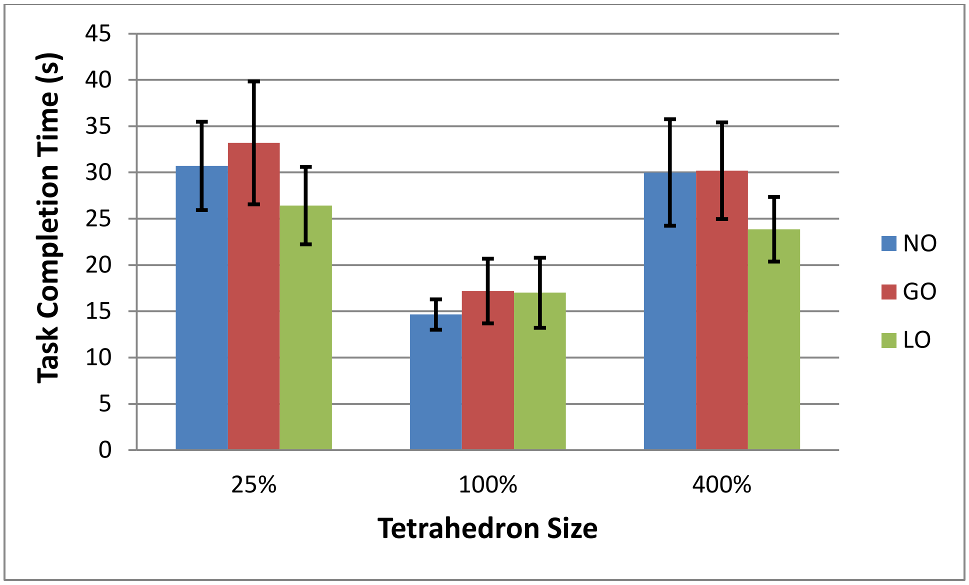


Figure 7.9: Task completion time by offset technique and object size in within reach condition. Error bar represents 95% confidence interval.

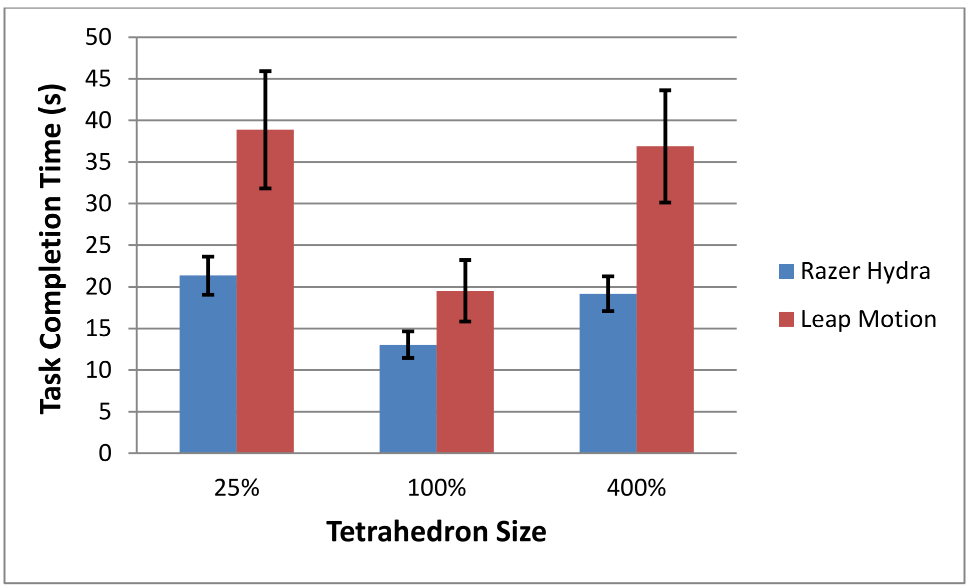


Figure 7.10: Task completion time by device and object size in within reach condition. Error bar represents 95% confidence interval.

The interaction effect for Device × Object Size is also significant, with simple effects of device across all three sizes (Figure 7.10). At 25% size level, completion time with Razer Hydra (*M*=21.35, *SD*=4.28) is faster than Leap Motion (*M*=38.86, *SD*=13.20,

p<.001). At 100% size level Razer Hydra (M=13.04, SD=3.01) is faster than Leap Motion (M=19.52, SD=6.90, p<.001). And at 400% size level Razer Hydra (M=19.16, SD=3.94) is still faster than Leap Motion (M=36.87, SD=12.65, p<.001).

The three-way interaction of Offset Technique × Device × Object Size is significant. Pairwise comparisons indicate that with Leap Motion, when the object shows up in 25% or 400% size, LO is faster than both NO and GO. (At 25%: LO (M=31.51, SD=11.80) faster than NO (M=42.07, SD=17.14, p=.009) and GO (M=43.01, SD=18.09, p=.024); At 400%: LO (M=27.65, SD=8.06) faster than NO (M=42.69, SD=19.10, p=.002) and GO (M=40.28, SD=16.26, p=.002)). This trend doesn't exist at 100% size level. At 100% level, docking with Razer Hydra is faster using NO (M=11.12, SD=2.05) than using LO (M=14.27, SD=4.39, p=.006).

For number of clutched maneuvers, the main effect of Offset Technique is significant. Pairwise comparisons show that all three offset techniques are different from each other. LO (M=9.78, SD=2.94) requires fewer number of clutched maneuvers than NO (M=14.65, SD=6.09, p=.001) and GO (M=11.69, SD=4.25, p=.025). And for the device factor, the number of button clicks on Razer Hydra (M=5.49, SD=1.02) is significantly fewer than the number of recognized gestures for Leap Motion (M=18.59, SD=7.42, p<.001). The main effect of Object Size is also significant. 100% size requires fewer clutched maneuvers (M=7.34, SD=2.70) than 25% (M=13.57, SD=4.48, p<.001) and 400% (M=15.21, SD=5.40, p<.001). But there is no difference between 25% and 400%.

There is a significant interaction effect for Offset Technique × Device (Figure 7.11). Post-hoc tests show that using Leap Motion, all three offset techniques require different numbers of recognized gestures. LO (*M*=14.19, *SD*=5.60) has fewer than NO (*M*=23.71, *SD*=11.56, *p*=.001) and GO (*M*=17.89, *SD*=8.03, *p*=.027). GO requires fewer than NO (*p*=.016).

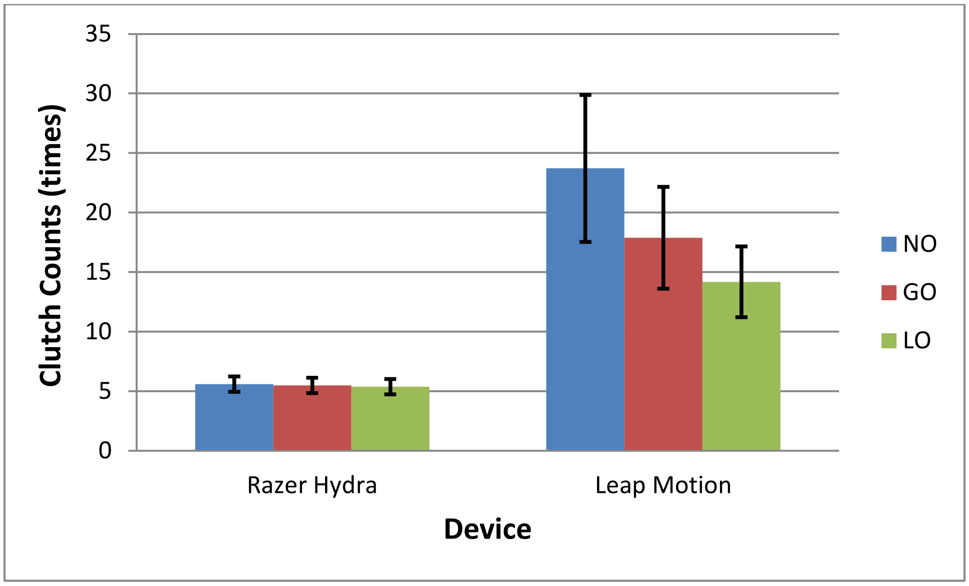


Figure 7.11: Number of clutched maneuvers by offset technique and device in within reach condition. Error bar represents 95% confidence interval.

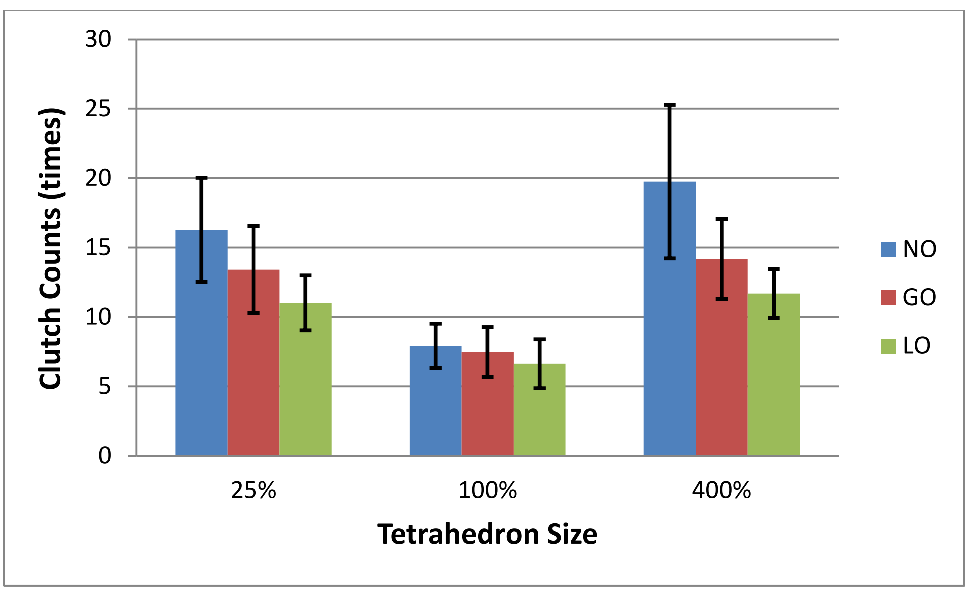


Figure 7.12: Number of clutched maneuvers by offset technique and object size in within reach condition. Error bar represents 95% confidence interval.

The interaction between Offset Technique and Object Size is significant (Figure 7.12). Pairwise comparisons reveal that at 25% size, subjects complete tasks with fewer clutched maneuvers using LO (*M*=11.01, *SD*=3.72) than using NO (*M*=16.27, *SD*=7.04, *p*=.003). While at 400% size, three offset techniques are all different from each other. The task requires fewer clutched maneuvers using LO (*M*=11.70, *SD*=3.32) than does NO (*M*=19.75, *SD*=10.39, *p*=.005) and GO (*M*=14.19, *SD*=5.40, *p*=.046); GO requires fewer than NO (*p*=.015). But at 100% size level, there is no simple effect of offset.

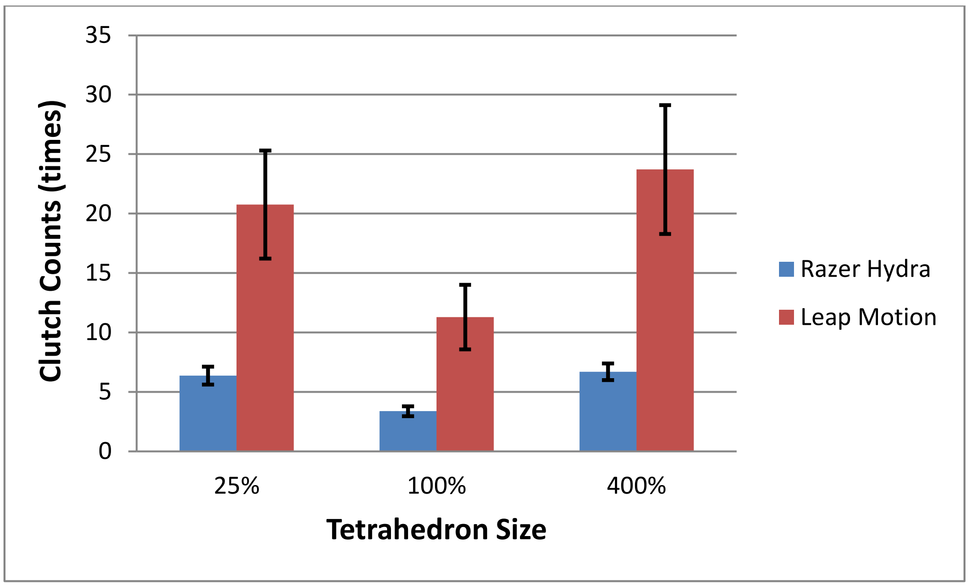


Figure 7.13: Number of clutched maneuvers by device and object size in within reach condition. Error bar represents 95% confidence interval.

The interaction effect for Device × Object Size is also significant, with simple effects of device on all three sizes (Figure 7.13). At 25% size, the number of clutched maneuvers on Razer Hydra (*M*=6.37, *SD*=1.42) is fewer than that of Leap Motion (*M*=20.76, *SD*=8.53, *p*<.001); at 100% size level, Razer Hydra (*M*=3.38, *SD*=0.77) has fewer than Leap Motion (*M*=11.31, *SD*=5.10, *p*<.001); again at 400% level, Razer Hydra (*M*=6.71, *SD*=1.31) still has fewer than Leap Motion (*M*=23.72, *SD*=10.17, *p*<.001).

The three-way interaction of Offset Technique × Device × Object size is significant. Post-hoc tests show that at 25% size level, docking using Leap Motion requires fewer clutched maneuvers with LO (*M*=15.81, *SD*=7.04) than with NO (*M*=26.00, *SD*=13.47, *p*=.002). Moreover, at 400% size level, docking using Leap Motion requires more clutched maneuvers with NO (*M*=32.60, *SD*=19.74) than GO (*M*=21.54, *SD*=10.19, *p*=.010) and LO (*M*=17.00, *SD*=5.92, *p*=.004). At 100% level, however, there is no simple effect of Offset Technique.

7.2.3.1.2 Out of Reach Condition

Table 7.2: Significant effects on task completion time and number of controls in out of reach condition.

	Task Completion Time			**Number of Clutched Maneuvers**		
Factor	***F***	***p***	η_p^2	***F***	***p***	η_p^2
Offset	*F*(1.32,19.82)=13.236	=.001	.469	*F*(2,30)=7.596	=.002	.336
Device	*F*(1,15)=63.301	<.001	.808	*F*(1,15)=55.189	<.001	.777
Size	*F*(2,30)=44.553	<.001	.748	*F*(2,30)=36.876	<.001	.711
Offset×Device	*F*(2,30)=18.345	<.001	.550	*F*(2,30)=7.226	=.003	.325
Device×Size	*F*(2,30)=7.300	=.003	.327	*F*(2,30)=8.310	=.001	.357

Significant effects of the experiment factors in out of reach condition are reported in Table 7.2. The main effect of Offset Technique is significant on task completion time. Pairwise comparisons indicate that three offset techniques are different from each other. Docking with LO (*M*=25.45, *SD*=8.84) is faster than FO (*M*=29.65, *SD*=9.54, *p*=.005) and GO (*M*=35.09, *SD*=10.74, *p*<.001). FO is faster than GO (*p*=.041). The main effect of Device is significant, with Razer Hydra (*M*=21.25, *SD*=5.27) faster than Leap Motion

(*M*=38.88, *SD*=12.78, *p*<.001). There is also a main effect of tetrahedron size on task completion time. Post-hoc tests show that 100% size (*M*=20.82, *SD*=9.54) is significantly faster than 25% (*M*=38.83, *SD*=12.28, *p*<.001) and 400% (*M*=30.54, *SD*=6.67, *p*<.001). And 400% is faster than 25% (*p*=.001).

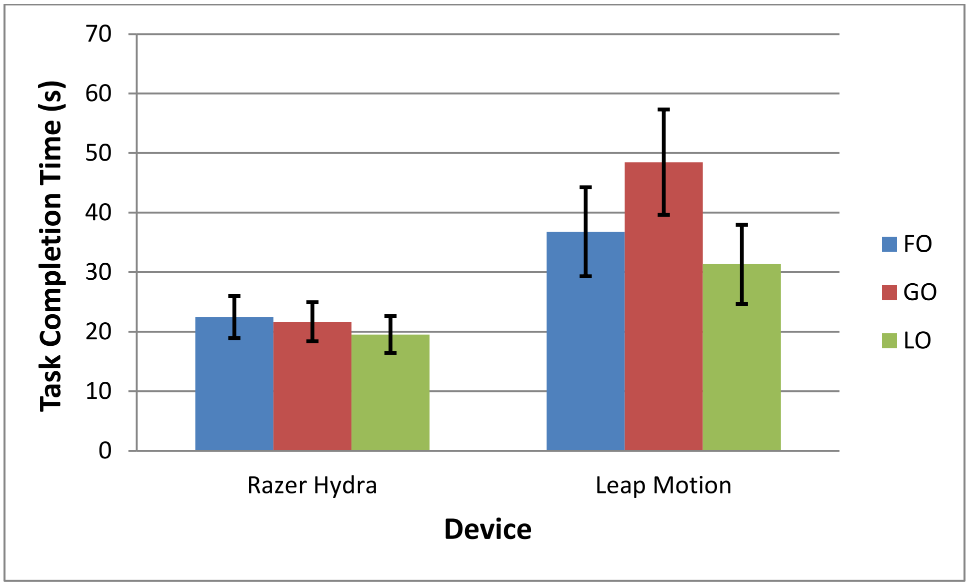


Figure 7.14: Task completion time by offset technique and device in out of reach condition. Error bar represents 95% confidence interval.

There is an interaction effect between Offset Technique and Device (Figure 7.14). Post-hoc tests reveal simple effects of offset technique on both input devices. With Razer Hydra, LO (*M*=19.54, *SD*=5.81) is faster than GO (*M*=21.70, *SD*=6.15, *p*=.034). With Leap Motion, LO (*M*=31.35, *SD*=12.47) is faster than FO (*M*=36.81, *SD*=14.02, *p*=.009) and GO (*M*=48.48, *SD*=16.58, *p*<.001). FO is also faster than GO (*p*=.007).

The interaction between Device and Object Size is significant (Figure 7.15). Post-hoc tests show that at 25% level, Razer Hydra (*M*=26.54, *SD*=7.23) is faster than Leap Motion (*M*=51.13, *SD*=18.67, *p*<.001). Razer Hydra (*M*=15.13, *SD*=4.29) is also faster than Leap Motion (*M*=26.51, *SD*=15.52, *p*=.002) at 100% size and 400% size level (Razer Hydra (*M*=22.08, *SD*=5.49), Leap Motion (*M*=39.00, *SD*=9.82, *p*<.001)).

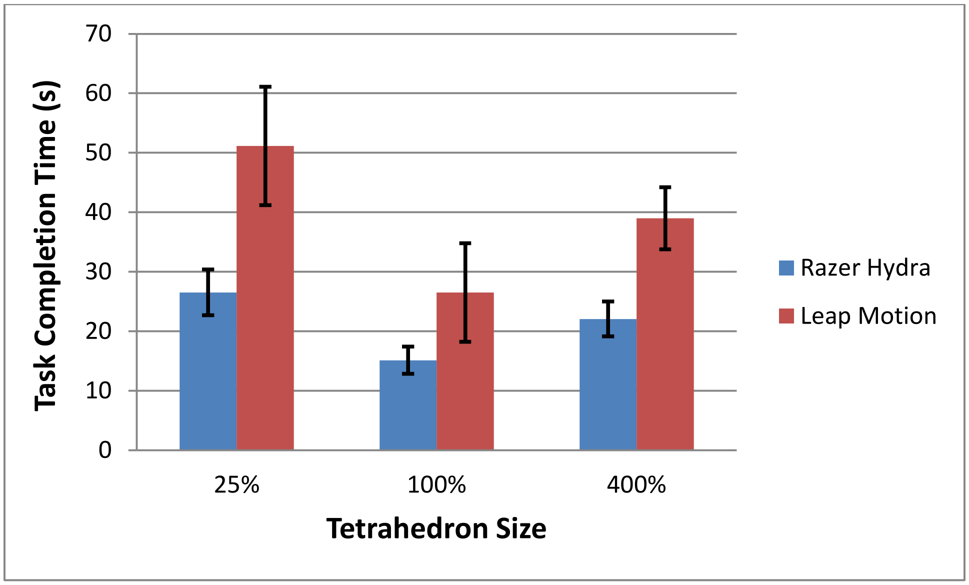


Figure 7.15: Task completion time by device and object size in out of reach condition. Error bar represents 95% confidence interval.

There is a significant main effect of Offset Techniques on the number of clutched maneuvers. Pairwise comparisons show that docking with LO requires fewer clutched maneuvers (*M*=11.07, *SD*=4.22) than with GO (*M*=15.18, *SD*=6.01, *p*=.001). The main factor of Device is significant, with fewer clutched maneuvers engaged by Razer Hydra (*M*=5.95, *SD*=1.24) than Leap Motion (*M*=19.98, *SD*=8.42, *p*<.001). There is also a significant main effect of Object Size. Pairwise comparisons show that the number of clutched maneuvers with 100% size (*M*=8.56, *SD*=4.65) is significantly fewer than that of 25% size (*M*=15.21, *SD*=6.42, *p*<.001) and 400% size (*M*=15.12, *SD*=3.62, *p*<.001).

There is an interaction effect between Offset Technique and Device (Figure 7.16). Post-hoc tests indicate that with Leap Motion, docking using LO (*M*=16.49, *SD*=8.07) requires fewer number of clutched maneuvers than using GO (*M*=24.43, *SD*=10.98,

p=.001). With Razer Hydra, however, the numbers of clutched maneuvers are not different among three offset techniques.

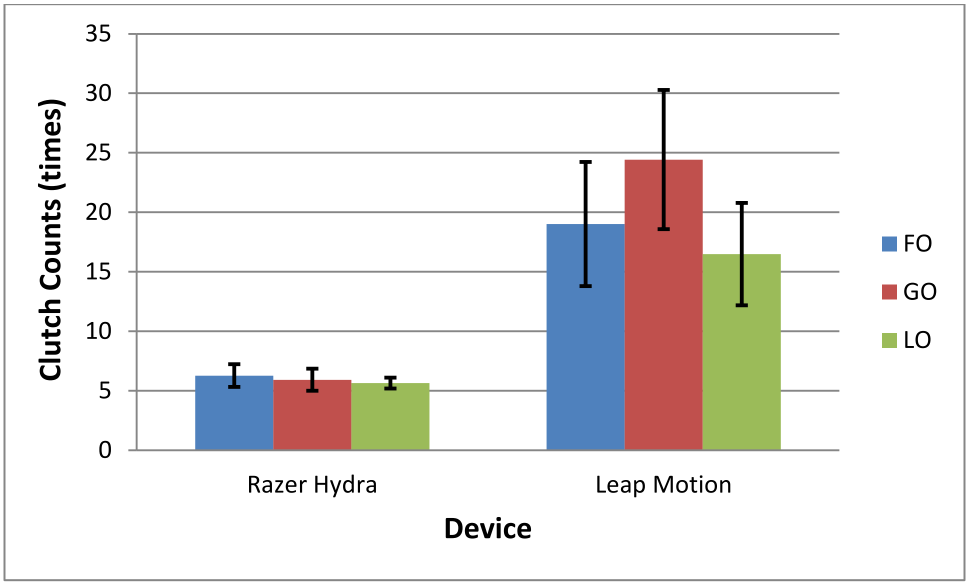


Figure 7.16: Number of clutched maneuvers by offset technique and device in out of reach condition. Error bar represents 95% confidence interval.

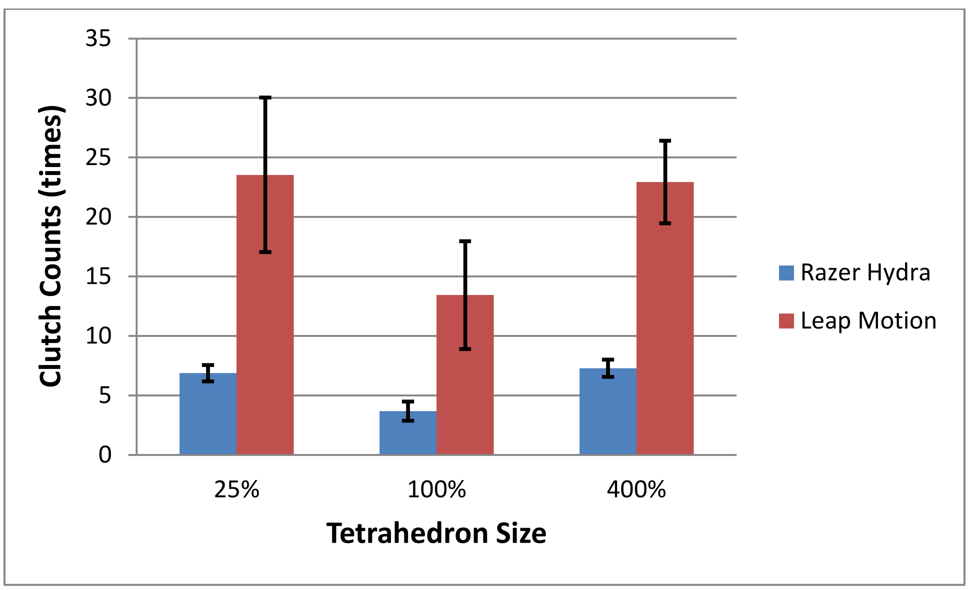


Figure 7.17: Number of clutched maneuvers by device and object size in out of reach condition. Error bar represents 95% confidence interval.

The interaction between Device and Object Size is significant (Figure 7.17). Post-hoc tests reveal simple effects of device on all three object sizes. At 25% size, the number of clutched maneuvers using Razer Hydra (*M*=6.88, *SD*=1.30) is fewer than using Leap Motion (*M*=23.54, *SD*=12.18, *p*<.001). At 100% size level, Razer Hydra requires fewer clutched maneuvers to complete the task (*M*=3.69, *SD*=1.51) than Leap Motion does (*M*=13.44, *SD*=8.52, *p*<.001). And so does 400% size level (Razer Hydra (*M*=7.29, *SD*=1.39), Leap Motion (*M*=22.95, *SD*=6.51, *p*<.001)).

7.2.3.2 Subjective Preferences

Participants give their ratings for arm fatigue level after each session on the 7-point Likert scale from 1 (‘Not at all’) to 7 (‘Very Painful’). For Razer Hydra, Friedman test (with α=.05 level for significance) did not show a main effect of offset technique on arm fatigue ratings (*p*=.304). Median (IQR) ratings using Razer Hydra are: NO, 3 (2.25 to 4.75); FO, 5 (3 to 6); GO, 4 (3.25 to 5); and LO, 4 (3 to 5). There is no significant main effect (*p*=.406) for Leap Motion either. Median (IQR) ratings using Leap Motion are: NO, 4 (3.25 to 5.75); FO, 5 (3.25 to 6); GO, 5 (4 to 6); and LO, 4 (3.25 to 6).

In the post-questionnaire, participants are asked to rate both the offset techniques and input devices based on how easy it is to finish the docking task using a 7-point Likert scale from 1 (‘Very easy’) to 7 (‘Very difficult’). The Friedman test shows a significant main effect of offset technique on the user ratings ($\chi^2(3)$=20.878, *p*<.001). Median (IQR) ratings are: NO, 3.5 (2 to 4.75); FO, 4 (3 to 5.75); GO, 4 (4 to 5); and LO, 3 (2 to 3). Wilcoxon signed-rank tests with Bonferroni correction for level of significance (α=.0083) show that users rate LO to be easier than FO (*Z*=−3.211, *p*=.001) and GO (*Z*=−2.984, *p*=.001), but not than NO. The ratings between two devices are also significantly different, with Razer Hydra being rated easier than Leap Motion (*Z*=−3.562, *p*<.001). The Median (IQR) rating for Razer Hydra is 2 (2 to 2.75) and for Leap Motion is 5 (5 to 5).

When asked which offset technique (NO/GO/LO) was the easiest to complete the docking task when the target tetrahedron is within reach, three subjects chose NO, two chose GO and eleven chose LO. For the condition when the target tetrahedron is out of reach, all sixteen subjects chose LO unanimously. For the input device, fifteen subjects preferred Razer Hydra and one preferred Leap Motion.

7.2.3.3 Discussion

In the within reach condition, Linear Offset performs better than Go-Go Offset. To our surprise, No Offset did not outperform Linear Offset. The trend even indicates the opposite with less overall mean of task complete time (M=22.43) for LO than that of NO (M=25.12). This contradicts hypothesis *H1*, which is originally built based on previous research results. NO did not show any advantage over GO and LO. Instead, it requires more clutched maneuvers to complete the docking task.

The interaction between offset technique and input device (Figure 7.8) indicates that Linear Offset is more robust than NO and GO, because it is affected the least when switching to hand gestures. NO is affected the most, because with no offset added to the cursor, the user will have to reach out more. This could result in the Leap Motion's intermittent loss of hand tracking, which also explains the increased number of clutched maneuvers with NO, since the user needs to retract the hand closer to the Leap Motion controller more often, to regain again.

Linear Offset also performs better compared to No Offset and Go-Go Offset when scaling is necessary, according to the interaction effect between Offset Technique and Object Size (Figure 7.9). When scaling is not required, three offset techniques are not different from each other. But when the object's size is different from the target's size, the performance of NO and GO drops more drastically than LO does. Similar pattern exists in the Offset Technique × Object Size interaction on the number of clutched maneuvers (Figure 7.12). When the controlled object shows up with the same size as the target, offset techniques do not differ whereas different sizes of the object increase the variance in the performance of offset techniques.

When the object is out of reach, Linear Offset outperforms Fix-Length Offset and Go-Go Offset. This is consistent with the results of our previous experiments in one-handed 3D UIs (Chapter 4) and supports hypothesis *H2*. Compared to FO's rigid offset length and GO's nonlinear gain factor, LO provides a more flexible offset calculation mechanism based on user movement. FO being faster than GO could be caused by the overshooting issue of GO.

The interaction effect between Offset Technique and Input Device (Figure 7.14) in out of reach condition shows that Go-Go Offset is affected the most when switching the input from Razer Hydra to Leap Motion. Both nonlinear mapping of hand's movement and instability of gesture recognition add up to the decline in GO's performance. The same trend can also be found in the interaction of Offset × Device on the number of clutched maneuvers (Figure 7.16).

Table 7.3: The marginal means of task completion time and number of clutched maneuvers for Razer Hydra and Leap Motion in within reach and out of reach conditions.

	Task Completion Time (s)		**Number of Clutched Maneuvers**	
Condition	**Razer Hydra**	**Leap Motion**	**Razer Hydra**	**Leap Motion**
Within Reach	17.85	31.75	5.49	18.59
Out of Reach	21.25	38.88	5.95	19.98

Overall, Linear Offset performs better than Go-Go Offset in both within reach and out of reach conditions, which supports hypothesis *H3*. And Razer Hydra shows a better performance than Leap Motion (*H4*). In particular, the number of gestures maneuvers with Leap Motion is three times more than the number of button clicks with Razer Hydra in both conditions (Table 7.3). Based on our observation, this is because Leap Motion can easily lose hand gesture tracking and the user has to adjust their hands to regain the control of the object multiple times whereas for Razer Hydra, the same operation can be completed with a single button click in most cases.

Moreover, the Device × Object Size interaction in both conditions (Figure 7.10, Figure 7.15) reveals that Leap Motion performs worse when changing scale is necessary compared to when no scale change is required. This is also caused by unstable tracking of hand since scaling with Leap Motion requires the user to move the hand away from the HMD or towards the HMD within the sagittal plane, while keeping the pinch gesture, which can be difficult for Leap Motion to detect if the hand is getting too close or too far. Specifically, pinching with Leap Motion comes with two issues. The first one is the false negative detection of the initial pinch gesture, which forces the user to spend more time trying to engage the scaling control. The other one is the false release of the pinching during hand movement when the user is still trying to change the size of the object, which results in increased number of maneuvers and thus leads to longer task completion time.

For number of clutched maneuvers, the interaction between Offset Technique and Input Device in within reach condition (Figure 7.11) and out of reach condition (Figure 7.16) indicates that the performance of Razer Hydra is consistent across different offset techniques in terms of button click, while with Leap Motion, the performance of offset techniques varies significantly depending on the motion range of the user's hand for each particular offset technique.

7.2.4 Experiment 6: Bimanual 7DOF Object Manipulation in HMD

Same offset techniques are compared in this experiment: No Offset, Fixed-Length Offset, Go-Go Offset and Linear Offset, using the Spindle+Wheel interaction technique. Sixteen volunteers are recruited (fifteen male and one female) for the study, with ages ranging from 18 to 32 (M=26.38). Eleven of them are from computer science department and five are non-CS major. All of the participants are right-handed and all have 20/20 (or corrected 20/20) eye vision with no disability using their arms and fingers. The daily computer usage among participants is high (6.75 out of 7). Ten of them have experience with 3D UIs, such as Microsoft Kinect, Nintendo Wii Remote and Oculus Rift.

This experiment has the same within-subjects design as Experiment 5, except that two separate offset vectors are added to the virtual cursors respectively for each hand to ensure offset technique's compatibility with the bimanual interaction (Figure 5.1(B)). For each cursor, the maximum length of offset is one meter, which is exactly the length for Fixed-Length Offset technique. For Go-Go Offset and Linear Offset, parameters are set accordingly to ensure the participant can reach the same position with both arms fully stretched. The order of offset technique for each participant is counterbalanced. Under each offset technique condition, half of the subjects start with Razer Hydra and then switch to Leap Motion and the other half finish the tasks with the reversed device order.

The primary hypotheses for experiment 6 are:

H1: When the object is within reach, NO is expected to be faster than GO and LO, because direct manipulation is better than indirect manipulation.

H2: LO will outperform FO and GO on the docking performance when the object is out of reach because it increments the offset gradually.

H3: Razer Hydra will perform better than Leap Motion, because bimanual interaction requires both hands to be tracked, which increases the chance of losing tracking.

7.2.4.1 Quantitative Results

Three-way repeated measures ANOVA tests are performed on the per trial mean of task completion time and effective number of controls for within reach condition and out of reach condition respectively, at the 5% significance level. Degrees of freedom are adjusted using Greenhouse-Geisser correction to protect against violation of sphericity assumption. Post-hoc tests are conducted using LSD pairwise comparisons with α=.05 level for significance.

7.2.4.1.1 Within Reach Condition

Table 7.4: Significant effects on task completion time and number of effective controls in within reach condition.

	Task Completion Time			**Number of Clutched Maneuvers**		
Factor	$\boldsymbol{F}$	$\boldsymbol{p}$	$\boldsymbol{\eta_p^2}$	$\boldsymbol{F}$	$\boldsymbol{p}$	$\boldsymbol{\eta_p^2}$
Offset	—	—	—	$F(1.31,19.68)=10.156$	=.003	.404
Device	$F(1,15)=25.467$	<.001	.629	$F(1,15)=28.806$	<.001	.658
Size	$F(2,30)=11.498$	<.001	.434	$F(2,30)=7.768$	=.002	.341
Offset×Device	$F(2,30)=15.711$	<.001	.512	$F(2,30)=10.499$	<.001	.412
Offset×Size	—	—	—	$F(4,60)=3.079$	=.023	.170

Significant effects of the experiment factors in the within reach condition are reported in Table 7.3. There is a significant main effect of Device on task completion time. Docking with Razer Hydra (*M*=20.79, *SD*=5.28) is faster than Leap Motion (*M*=30.61, *SD*=11.45, *p*<.001). The main effect of object size is significant. Pairwise comparisons show that tasks with 100% size (*M*=22.52, *SD*=7.80) has less completion time than those with 25% size (*M*=28.90, *SD*=9.66, *p*<.001) and 400% size (*M*=25.68, *SD*=8.20, *p*=.025). And 400% is faster than 25% (*p*=.028).

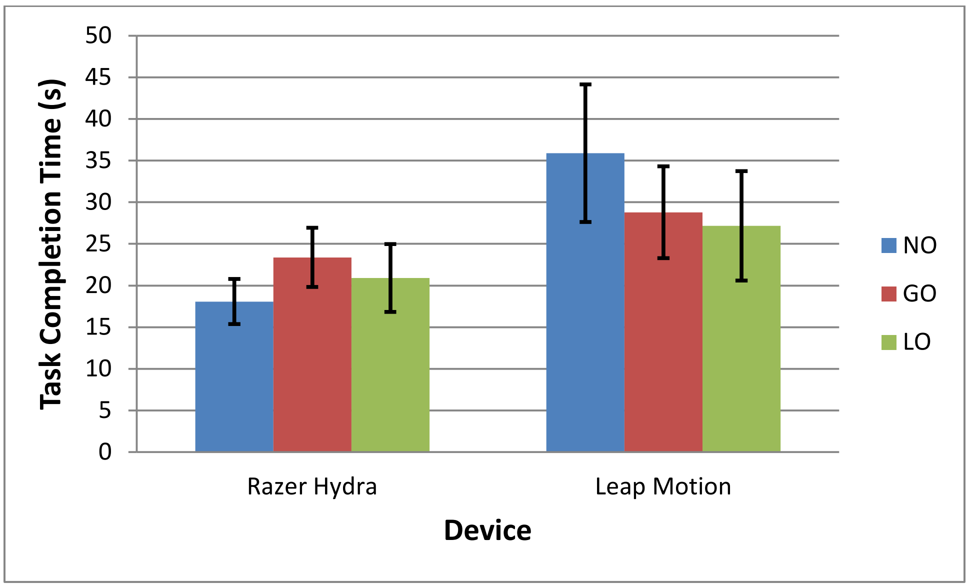


Figure 7.18: Task completion time by offset technique and device in within reach condition. Error bar represents 95% confidence interval.

There is a significant interaction effect between Offset Technique and Device (Figure 7.18). Post-hoc tests show that with Razer Hydra, docking using NO (*M*=18.07, *SD*=5.09) is faster than using GO (*M*=23.38, *SD*=6.68, *p*<.001). While with Leap Motion, NO (*M*=35.90, *SD*=15.49) is significantly slower than GO (*M*=28.79, *SD*=10.36, *p*=.007) and LO (*M*=27.16, *SD*=12.33, *p*=.013).

For clutch counts, the main effect of Offset Technique is significant. Pairwise comparisons show that it requires more clutched maneuvers docking with NO (*M*=8.22, *SD*=2.55) than with GO (*M*=6.34, *SD*=1.58, *p*=.003) and with LO (*M*=5.82, *SD*=2.26, *p*=.005). But there is no difference between GO and LO. There is a significant main effect of Device. The number of button clicks on Razer Hydra (*M*=4.76, *SD*=1.68) is significantly fewer than the number of gesture maneuvers with Leap Motion (*M*=8.82, *SD*=2.80, *p*<.001). The main effect of Object Size is also significant. 100% size requires fewer clutched maneuvers (*M*=6.06, *SD*=1.75) than 25% (*M*=6.85, *SD*=2.10, *p*=.036) and 400% (*M*=7.47, *SD*=1.93, *p*=.002). There is no difference between 25% and 400%.

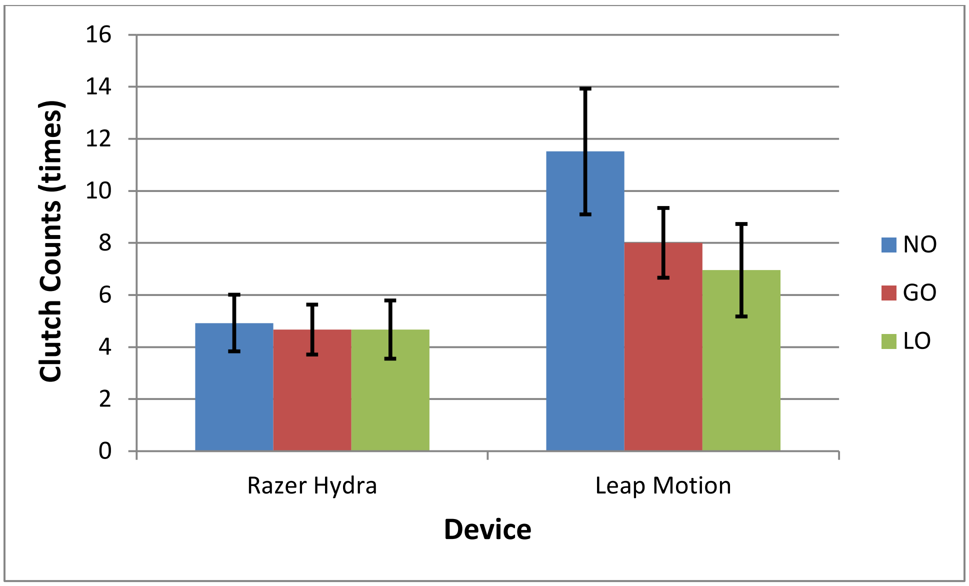


Figure 7.19: Number of clutched maneuvers by offset technique and device in within reach condition. Error bar represents 95% confidence interval.

There is an interaction effect between Offset Technique and Device (Figure 7.19). Post-hoc tests indicate that using Leap Motion, docking with NO requires significantly more gesture maneuvers (*M*=11.51, *SD*=4.53) than with GO (*M*=8.00, *SD*=2.52, *p*=.001) and LO (*M*=6.96, *SD*=3.33, *p*=.002). Using Razer Hydra, the numbers of button clicks are not different from each other among three offset techniques.

The interaction between Offset Technique and Object Size is also significant (Figure 7.20). Post-hoc tests reveal simple effects of offset technique on all three sizes.

At 25% level, number of clutched maneuvers with NO (*M*=8.82, *SD*=3.10) is more than that of GO (*M*=5.55, *SD*=1.80, *p*<.001) and LO (*M*=6.18, *SD*=2.64, *p*=.005). At 100% level, LO has fewer number of clutched maneuvers (*M*=4.98, *SD*=1.94) than NO (*M*=6.74, *SD*=2.32, *p*=.003) and GO (*M*=6.45, *SD*=2.23, *p*=.025). And lastly at 400% level, the number of clutched maneuvers of NO (*M*=9.09, *SD*=3.77) is more than GO (*M*=7.01, *SD*=2.00, *p*=.040) and LO (*M*=6.29, *SD*=3.04, *p*=.044).

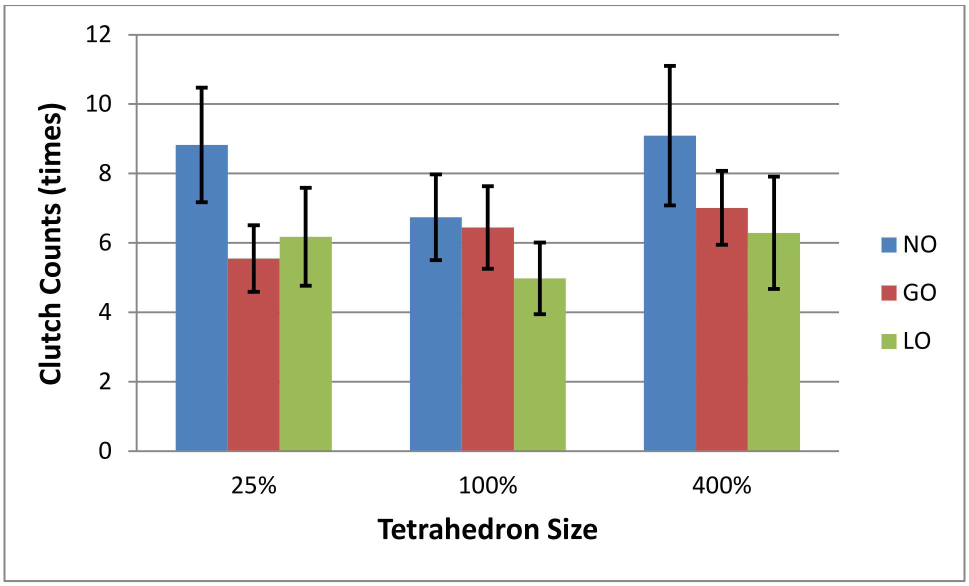


Figure 7.20: Number of clutched maneuvers by offset technique and object size in within reach condition. Error bar represents 95% confidence interval.

7.2.4.1.2 Out of Reach Condition

Significant effects in the out of reach condition are reported in Table 7.4. The main effect of Offset Technique is significant. Pairwise comparisons indicate that LO (*M*=26.17, *SD*=8.73) is faster than FO (*M*=35.62, *SD*=11.55, *p*=.010) and GO (*M*=32.31, *SD*=11.68, *p*=.026). The task completion time of FO is not different from GO. There is a significant main effect of Device, with Razer Hydra (*M*=26.12, *SD*=7.20) faster than Leap Motion (*M*=36.61, *SD*=10.46, *p*<.001). The main effect of Object Size is also significant. Docking with 25% size (*M*=36.73, *SD*=9.29) is slower than both 100% size (*M*=28.98, *SD*=10.97, *p*=.001) and 400% size (*M*=28.39, *SD*=6.98, *p*<.001). However, the completion time of 100% is not different from that of 400%.

Table 7.5: Significant effects on task completion time and number of effective controls in out of reach condition.

	Task Completion Time			**Number of Clutched Maneuvers**		
Factor	***F***	***p***	η_p^2	***F***	***p***	η_p^2
Offset	*F*(2,30)=5.356	=.010	.263	—	—	—
Device	*F*(1,15)=40.071	<.001	.728	*F*(1,15)=50.825	<.001	.772
Size	*F*(2,30)=14.952	<.001	.499	—	—	—
Offset×Size	*F*(2.18,32.66)=3.459	=.040	.187	—	—	—

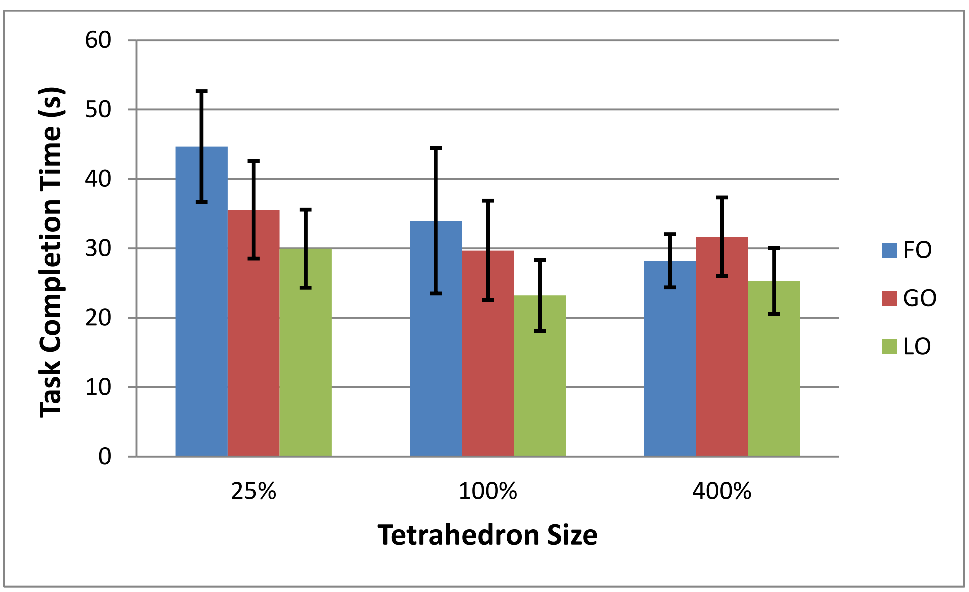


Figure 7.21: Task completion time by offset technique and object size in out of reach condition. Error bar represents 95% confidence interval.

There is a significant interaction effect between Offset Technique and Object Size (Figure 7.21). Post-hoc tests show that at 25% size level, docking with LO is significantly faster (*M*=29.97, *SD*=10.53) than FO (*M*=44.67, *SD*=15.00, *p*=.004) and GO (*M*=35.56, *SD*=13.18, *p*=.048). At 400% level, docking with LO (*M*=25.29, *SD*=8.91) is faster than GO (*M*=31.67, *SD*=10.67, *p*=.033). Whereas at 100% level, the three offset techniques are not different from each other on task completion times.

For number of clutched maneuvers, only the main effect of Device is significant. The number of button clicks on Razer Hydra (*M*=5.57, *SD*=2.23) is fewer than the number of gestures maneuvers used with Leap Motion (*M*=10.44, *SD*=2.92, *p*<.001).

7.2.4.2 Subjective Preferences

Participants give arm fatigue ratings after using each device within each offset technique condition. The Friedman tests (with α=.05 level for significance) do not reveal any main effect of offset technique on arm fatigue ratings, for either Razer Hydra (*p*=.067) or Leap Motion (*p*<.434). Median (IQR) fatigue ratings using Razer Hydra are: NO, 4 (3 to 5.75); FO, 4.5 (4 to 5.75); GO, 4 (3 to 5.75); and LO, 3.5 (3 to 4.75). Median (IQR) fatigue ratings using Leap Motion are: NO, 4 (4 to 5); FO, 5 (4 to 5.75); GO, 4.5 (3 to 6); and LO, 4.5 (3 to 5).

In the post-questionnaire, participants are asked to rate both offset techniques and input devices based on how easy it makes the docking task on a 7-point Likert scale from 1 ('Very easy') to 7 ('Very difficult'). The Friedman test shows a main effect of offset on user ratings (χ^2(3)=21.596, *p*<.001). Median (IQR) ratings are: NO, 3 (2 to 3.75); FO, 4 (4 to 5); GO, 3.5 (3 to 4); and LO, 3 (2 to 3). The Wilcoxon signed-rank tests with Bonferroni adjustment for the level of significance (α=.0083) show that users rate NO to be easier than FO (Z=−2.961, p=.002). LO is rated to be easier than both FO (Z=−3.482, p<.001) and GO (Z=−2.696, p=.008). The ratings for two devices are also significantly different, with Razer Hydra being rated easier than Leap Motion (Z=−3.163, p=.001). The Median (IQR) rating for Razer Hydra is 2 (1 to 2.75) and for Leap Motion is 5 (3 to 5).

When asked which offset technique (NO/GO/LO) was the easiest to complete the docking task when the target tetrahedron is within reach, eight subjects chose NO, two chose GO and six chose LO. For out of reach condition (FO/GO/LO), one answered FO, six answered GO and nine answered LO. As to the preferences for input devices, thirteen subjects preferred Razer Hydra while three preferred Leap Motion.

7.2.4.3 Discussion

The analysis indicates that when the target tetrahedron is within reach, docking with No Offset, Go-Go Offset and Linear Offset techniques does not differ in terms of task completion time, which counters hypothesis *H1*. I suspect that the absence of significance is caused by the complexity of bimanual interaction imposed. Since different offset vectors are added to the left and right hand, the global effect of both vectors can be unpredictable. On the other hand, this demonstrates that direct manipulation does not necessarily perform better than indirect manipulation.

The interaction effect between Offset Technique and Input Device (Figure 7.18) indicates that No Offset is affected more by Leap Motion, compared to GO and LO. This is probably because in NO mode, the user has to reach out further more frequently than she does in GO and LO mode, resulting in hands moving out of the tracking space more often. This also causes the interaction effect of Offset Technique × Device on the number of controls (Figure 7.19), since the user has to regain the control of the object more often after the tracking is recovered.

GO is affected the least by different object sizes on the number of clutched maneuvers based on the interaction Offset Technique × Object Size (Figure 7.20). As the size increases, the number of clutched maneuvers with GO also increments in contrast to the trend shown by NO and LO that 100% size requires the smallest number of clutched maneuvers. Since with the Spindle+Wheel technique, moving two hands closer or farther apart scales the object, GO has the advantage of changing the size of the object faster than NO and LO by amplifying the hand motion through its quadratic gain factor.

When the object is out of reach, LO performs better than FO and GO as expected (*H2*), because it dynamically adjusts offset length based on the hand's relative position to the HMD. FO and GO can be helpful, but both have their drawbacks. With FO, users often need to retract their hands or even move their body backwards in order to put the virtual cursor inside the object. And GO constantly causes overshooting, especially when the controlled object is at 25% size and in the far end of the workspace.

The interaction effect between offset technique and size (Figure 7.21) shows that docking with FO benefits more from the 400% target size. Since the object is far away and FO provides with a rigid offset vector, it is easier for the user to grab the object and make it smaller from 400% size.

Overall, Razer Hydra outperforms Leap Motion controller in both task completion time and number of clutched maneuvers (Table 7.6), which confirms hypothesis *H3*. The Spindle+Wheel interaction technique enables the possibility of simultaneously changing all 7DOF. Therefore, users sometimes can finish the docking task with only one button click on Razer Hydra. The limited tracking frustum and unstable tracking of Leap Motion greatly reduces the chance of finishing the task with only few gestures. But compared to the results of one-handed manipulation (Table 7.3), the difference between the two controllers becomes much smaller. Especially with Leap Motion, the clutched maneuver counts reduced 50% on average. This is mainly because for bimanual interaction, only one gesture is needed to engage control of the object, which is the open palm gesture. Whereas in the one-handed experiment, a second pinch gesture is required to change the size of the object. Based on our observation and experience, the open palm gesture has a higher recognition rate than the pinch gesture. Therefore, the performance of one-handed manipulation is affected more by the accuracy of the gesture detection and hand tracking than the two-handed mode.

Table 7.6: The marginal means of task completion time and number of clutched maneuvers for Razer Hydra and Leap Motion in within reach and out of reach conditions.

	Task Completion Time (s)		**Number of Clutched Maneuvers**	
Condition	**Razer Hydra**	**Leap Motion**	**Razer Hydra**	**Leap Motion**
Within Reach	20.79	30.61	4.76	8.82
Out of Reach	26.12	36.61	5.57	10.43

7.3 Overall Discussion

Although performance of offset techniques on object manipulation varies between one-handed and two-handed interaction, and between input devices, linear offset stands out in general. When the object is within reach distance, direct manipulation (NO) is not shown to be faster than manipulation with a linear offset. In addition, LO requires fewer number of clutched maneuvers than NO does for the docking task making LO more efficient than NO. When the object is out of reach, LO proves to be superior than NO and GO in both experiments.

Docking with Razer Hydra controller shows overwhelming advantages over Leap Motion controller, but that does not necessarily mean interaction with hand gestures is not as good as interaction with hand-held devices. The Leap Motion controller is known to provide relatively stable and accurate tracking of individual fingers with minimal jitter, which makes using bare hand gestures as input convenient, but the frequent loss of hand tracking given the range of arm motion users typically used in these experiments can easily frustrate users and thus limiting its usability in a more natural 3D user interface. For example, the initial design used grabbing the object via a fist gesture to engage control of the object, in order to make the gesture analogous to the interaction in the real world. But when held in a fist, the hand tracking become too unstable to be usable for unimanual 6DOF manipulation. Several informal experiments demonstrated that the more extended the fingers are, the more stable the hand orientation tracking can be. Therefore the open hand gesture is eventually adopted.

One speculation about why fewer significant main effects and interaction effects are found in the bimanual experiment than those found in the unimanual experiment is that it takes longer to learn how to properly use the Spindle+Wheel technique. Because this technique provides the ability to simultaneously change 7DOF using both hands, it imposes more cognitive load to the user, which in turn requires higher training effort. In Schultheis et al. [116]'s study, they attribute the success of their two-handed interface to the proper training and sufficient practice with an average of 47 minutes, which is almost three times longer than our training session. And their conclusion is that the two-handed interface may only be appropriate for professionals who use it on a daily basis.

7.4 Conclusion

Two formal user studies are conducted to investigate the extensive influence of virtual cursor offset techniques on object manipulation in an HMD system using one-handed and two-handed interaction techniques respectively. The results of Experiment 5 show that for manipulation of objects within reach, Linear Offset is better than Go-Go offset in task completion time and required number of controls. No significant difference is found between No Offset and Linear Offset. When the objects are out of reach, docking is faster with LO than with FO or GO. LO is also more efficient than GO on the number of operations. In the bimanual experiment, three offset techniques are not significantly different on completion time when the object is close by, although direct manipulation requires more number of operations. For objects that are at a distance, manipulation is faster with Linear Offset than with FO and GO. Across all conditions, the Razer Hydra controller performs better than the Leap Motion controller. And the culprit seems to be the loss of hand tracking during the range of motion users expected to use in this experiment.

CHAPTER 8: CONCLUSION AND FUTURE WORK

8.1 Primary Contributions

For nearly five decades, VR researchers have been running experiments and gathering evidence for the fundamental effects of control-display mapping on different VR systems. But not much work went into validating the effects and benefits of virtual cursor offset, and more importantly, understanding what features of direct and indirect interaction are important to immersive VR and 3D user interfaces. Throughout the six experiments presented in this dissertation, I add new scientific evidence suggesting that carefully designed cursor offset technique could significantly improve task performance for 3D interaction in immersive virtual environments. Our empirical results provide additional supporting evidence to existing literature as well. The primary contributions of the work presented in this dissertation are as follows:

Chapter 4 presents two formal user studies to examine the effect of cursor offset techniques on one-handed 7DOF navigation in a head-tracked, stereoscopic three-side CAVE system. The first study compares four different virtual cursor offset techniques and the results indicate that Linear Offset technique outperforms other three techniques on both 6DOF and 7DOF travel tasks. Moreover, Fixed-Length Offset enhances user performance more than No Offset does, but Go-Go Offset does not prove to be better than No Offset after all with a nonlinear mapping. The majority of the subjects (12 out of 16) chose Linear Offset to be the most helpful condition for the task. The second study explores how different offset length of Linear Offset affects user performance on the same task. Statistical analysis reveals that the length of 96” could help the user complete the task more than 24” on all three box sizes, although 48” is not different from either 24” or 96”. And no offset with 0” is the slowest. Subjects report more arm fatigue with 0” than with 24” and 96”. Most subjects (12 out of 16) preferred 96” for the offset setting.

Chapter 5 evaluates the same four offset techniques on a bimanual navigation task in the surround-screen CAVE system. I first extend cursor offset mechanism by adding a separate offset vector to left and right cursor respectively, based on the motor behavior of both hands, thus enabling smooth and intuitive addition of offset without introducing inconsistency to the graphical feedback. Then an experiment is carried out to examine the effects of cursor offset further on bimanual interaction. The results show that the Linear Offset technique helps the user perform better than No Offset, Fixed-Length Offset and Go-Go Offset and FO is better than NO, but GO is not different from NO or FO due to its sensitive cursor motion. Linear Offset technique is rated the best unanimous in subjective preference. Further examination of the kinematic characteristics of four offset techniques indicate that with NO and GO, the users tend to reach out and hold the arm longer than FO and LO, although no significant effect is found for arm fatigue ratings between offset techniques.

In Chapter 6, cursor offset techniques are re-evaluated on object selection tasks following Fitts law model under ISO 9241-9 standard using two different input devices in an HMD environment. The analysis of the experiment results are divided into two parts based on the selection distance to the user. When the targets are within reach, selection performance is better with No Offset than with Go-Go Offset and Linear Offset, which is in line with the conclusion from previous study [88] that manipulation of objects that are collocated with the user's hand performs better than manipulation of objects at a fixed or varied offsets in HMD environments. Direct manipulation remains the most natural way of interacting with nearby objects. The performance of Linear Offset reveals a tendency of positive correlation with the target depth. When the targets are located at a distance out of reach, Linear Offset outperforms Fixed-Length Offset and Go-Go Offset on movement time, error rate and effective throughput by enabling the user to precisely place the virtual cursor inside the 3D object. In the meantime, our preliminary comparison indicates that the Razer Hydra controller is more reliable for selection tasks than Leap Motion. The main disadvantage of the Leap Motion controller is its intermittent loss of hand tracking during the range of motion users expected to use in our experiment, which is likely to be improved considerably in the future implementations using multiple cameras with wider FOV.

Chapter 7 presents two experiments that investigate the influence of virtual cursor offset techniques on object manipulation in an HMD system using one-handed and two-handed interaction techniques respectively. The results of the unimanual study indicate that Linear Offset is more advantageous than Go-Go Offset at manipulation of objects within reach. Direct manipulation with No Offset is not faster than GO or LO, which is in contradiction to the previous research. Furthermore, users even need more clutched maneuvers with NO to complete the docking task than with GO and LO. When the objects are out of reach, docking is faster with LO than with FO or GO. LO is also more efficient than GO on the number of clutched maneuvers. In the bimanual experiment, three offset techniques are not significantly different from each other on task completion time when the object is close by, although NO requires more clutched maneuvers than GO and LO. For objects at a distance, manipulation is faster with Linear Offset than with FO and GO. Across all conditions, the Razer Hydra controller performs better than the Leap Motion controller.

8.2 Limitations

I tried to be meticulous and comprehensive as much as possible in experimental designs and procedures, but I acknowledge the possible limitations of our approach and findings. Our empirical studies are guided by VR experts, but the participants are always novices to this domain, which is prone to introduce possible discrepancies between the expected results and the actual performance. This is primarily caused by the differences between expert and novice users in comprehending the task and strategies for employing the input device. To partially compensate for this inconsistency, I recommend expert strategies to the novice participants during the training session. I also closely monitor the operations of the participants to ensure the efficiency of the interaction and the clarity of their understanding, by using layman terms in the definitions and providing further clarifications whenever necessary during the trials.

Fitts' law model is mostly used in predicting the movement time or comparing the performance of different input devices on object selecting or pointing tasks through rapid arm movements. In the selection experiment, I observed that participant's consecutive selection movement is often interrupted by various factors and thus affecting the data distribution for statistical analysis. Therefore, I summarize two improvements that can be made to possibly eliminate the interruption. First, provide more visual cues in the virtual scene by either making the targets semi-transparent or changing the target's color whenever the cursor is inside, so that the user can have a better perception of the cursor position relative to the target. Second, use the non-dominant hand to confirm the selection once the cursor is inside the target to avoid any jittering caused by any movement of their dominant hand or to avoid the false negative detection problem with camera based finger tracking.

In all three experiments conducted with the HMD, Razer Hydra shows superior performance over Leap Motion due to the frequent loss of hand tracking provided by Leap Motion given the range of arm motion users typically used in these experiments. But that does not necessarily mean that mid-air bare hand interaction is not as effective as interaction with hand-held devices. More advanced motion capture cameras can be used to provide more accurate hand tracking data, possibly with markers attached to fingers or palm. The Leap Motion controller is chosen for our studies because of its low cost and compatibility with the Oculus Rift HMD.

8.3 Future Work

This dissertation can be extended in several important directions. Our experiments show that different offset technique and VR system combinations affect different types of tasks to different degrees. To explore more possibilities, additional evaluation of cursor offset techniques on 7DOF navigation can be carried out in an HMD system. In a navigation task, virtual cursor does not need to be placed as precisely within a specific object, so the user will have more freedom of “grabbing” the entire scene, which could expose more characteristics of the offset techniques in the HMD environment.

With cursor offset techniques, it can be difficult and frustrating for users to move their hands to a precise location and hold still in 3D work space, because essentially these techniques scale up the hand movement to increase the range of control space, which inevitably results in the imprecision of user interaction. To compensate for this drawback, some researchers [37] propose a new technique that adjusts the movement scale based on the user's behavior to increase precision. Therefore, I suggest a hybrid offset technique that combines the advantages of linear offset and their PRISM technique can be evaluated in the further study.

Control-Display mapping is a concept involving multiple factors, among which, translation is the most studied one. However, rotation also plays an important role in 3D user interaction. Most studies employ isomorphic rotation mapping for interaction in the VE and few [106] [66] has explored the performance characteristics of non-isomorphic rotation techniques. Another possible direction is to combine non-isomorphic translation and rotation techniques to provide a more comprehensive investigation of the influence of Control-Display mapping on the speed and accuracy of 3D user interaction in immersive virtual environments.

REFERENCES

[1] Argelaguet, F. and Andujar, C., "A Survey of 3D Object Selection Techniques for Virtual Environments," Computers & Graphics, vol. 37, no. 3, 121-136 (2013).

[2] Argelaguet, F., Andujar, C., and Trueba, R., "Overcoming Eye-hand Visibility Mismatch in 3D Pointing Selection," Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology, New York, NY, USA, pp. 43-46 (2008).

[3] Argelaguet, F., Hoyet, L., Trico, M., and Lecuyer, A., "The Role of Interaction in Virtual Embodiment: Effects of the Virtual Hand Representation," 2016 IEEE Virtual Reality (VR), Greenville, SC, USA, pp. 3-10 (2016).

[4] Auteri, C., Guerra, M., and Frees, S., "Increasing Precision for Extended Reach 3D Manipulation," The International Journal of Virtual Reality, vol. 12, no. 1, 66-73 (2013).

[5] Bachmann, D., Weichert, F., and Rinkenauer, G., "Evaluation of the Leap Motion Controller as a New Contact-Free Pointing Device," Sensors, vol. 15, no. 1, 214-233 (2015).

[6] Balakrishnan, R. and Hinckley, K., "Symmetric Bimanual Interaction," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 33-40 (2000).

[7] Balakrishnan, R. and Hinckley, K., "The Role of Kinesthetic Reference Frames in Two-handed Input Performance," Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 171-178 (1999).

[8] Balakrishnan, R. and Kurtenbach, G., "Exploring Bimanual Camera Control and Object Manipulation in 3D Graphics Interfaces," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 56-62 (1999).

[9] Balakrishnan, R. and MacKenzie, I. S., "Performance Differences in the Fingers, Wrist, and Forearm in Computer Input Control," Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 303-310 (1997).

[10] Baudel, T. and Beaudouin-Lafon, M., "Charade: Remote Control of Objects Using Free-hand Gestures," Commun. ACM, vol. 36, no. 7, 28-35 (1993).

[11] Benko, H. and Feiner, S., "Balloon Selection: A Multi-Finger Technique for Accurate Low-Fatigue 3D Selection," 3D User Interfaces, 2007. 3DUI '07. IEEE

Symposium on, pp. 79-86 (2007).

[12] Benko, H. and Wilson, A. D., "Multi-point Interactions with Immersive Omnidirectional Visualizations in a Dome," ACM International Conference on Interactive Tabletops and Surfaces, New York, NY, USA, pp. 19-28 (2010).

[13] Best, S., "Perceptual and Oculomotor Implications of Interpupillary Distance Settings on a Head-Mounted Virtual Display," Aerospace and Electronics Conference, 1996. NAECON 1996., Proceedings of the IEEE 1996 National, vol. 1, pp. 429-434 (1996).

[14] Blanch, R., Guiard, Y., and Beaudouin-Lafon, M., "Semantic Pointing: Improving Target Acquisition with Control-display Ratio Adaptation," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 519-526 (2004).

[15] Bogdan, N., Grossman, T., and Fitzmaurice, G., "HybridSpace: Integrating 3D Freehand Input and Stereo Viewing into Traditional Desktop Applications," 3D User Interfaces (3DUI), 2014 IEEE Symposium on, pp. 51-58 (2014).

[16] Boritz, J. and Booth, K. S., "A Study of Interactive 6 DOF Docking in a Computerized Virtual Environment," Virtual Reality Annual International Symposium, 1998. Proceedings of the IEEE 1998, Washington, DC, USA, pp. 139-146 (1998).

[17] Bowman, D. A., Datey, A., Ryu, Y. S., Farooq, U., and Vasnaik, O., "Empirical Comparison of Human Behavior and Performance with Different Display Devices for Virtual Environments," Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 46, pp. 2134-2138 (2002).

[18] Bowman, D. A., Gabbard, J. L., and Hix, D., "A Survey of Usability Evaluation in Virtual Environments: Classification and Comparison of Methods," Presence: Teleoper. Virtual Environ., vol. 11, no. 4, 404-424 (2002).

[19] Bowman, D. A. and Hodges, L. F., "An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments," Proceedings of the 1997 Symposium on Interactive 3D Graphics, New York, NY, USA, pp. 35-38 (1997).

[20] Bowman, D. A., Koller, D., and Hodges, L. F., "Travel in Immersive Virtual Environments: An Evaluation of Viewpoint Motion Control Techniques," Proceedings of the 1997 IEEE Virtual Reality Annual International Symposium, Washington, DC, USA, pp. 45-52 (1997).

[21] Bowman, D. A., Kruijff, E., LaViola, J. J., and Poupyrev, I., 3D User Interfaces:

Theory and Practice, Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc. (2004).

[22] Bruder, G., Lubas, P., and Steinicke, F., "Cognitive Resource Demands of Redirected Walking," Visualization and Computer Graphics, IEEE Transactions on, vol. 21, no. 4, 539-544 (2015).

[23] Bruder, G., Steinicke, F., and Stürzlinger, W., "Effects of Visual Conflicts on 3D Selection Task Performance in Stereoscopic Display Environments," 3D User Interfaces (3DUI), 2013 IEEE Symposium on, pp. 115-118 (2013).

[24] Buxton, W. and Myers, B., "A Study in Two-handed Input," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 321-326 (1986).

[25] Cashion, J., Wingrave, C., and Jr., J. J. L., "Dense and Dynamic 3D Selection for Game-Based Virtual Environments," IEEE Transactions on Visualization and Computer Graphics, vol. 18, no. 4, 634-642 (2012).

[26] Cashion, J., Wingrave, C., and Jr., J. J. L., "Optimal 3D Selection Technique Assignment Using Real-Time Contextual Analysis," 3D User Interfaces (3DUI), 2013 IEEE Symposium on, pp. 107-110 (2013).

[27] Casiez, G., Vogel, D., Balakrishnan, R., and Cockburn, A., "The Impact of Control-Display Gain on User Performance in Pointing Tasks," Human-Computer Interaction, vol. 23, no. 3, 215-250 (2008).

[28] Chen, J., Pyla, P. S., and Bowman, D. A., "Testbed Evaluation of Navigation and Text Display Techniques in an Information-Rich Virtual Environment," Virtual Reality, 2004. Proceedings. IEEE, pp. 181-188 (2004).

[29] Cho, I., Li, J., and Wartell, Z., "Evaluating Dynamic-Adjustment of Stereo View Parameters in a Multi-Scale Virtual Environment," 3D User Interfaces (3DUI), 2014 IEEE Symposium on, pp. 91-98 (2014).

[30] Cho, I. and Wartell, Z., "Evaluation of a Bimanual Simultaneous 7DOF Interaction Technique in Virtual Environments," 3D User Interfaces (3DUI), 2015 IEEE Symposium on, pp. 133-136 (2015).

[31] Cruz-Neira, C., Sandin, D. J., and DeFanti, T. A., "Surround-screen Projection-based Virtual Reality: The Design and Implementation of the CAVE," Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA, pp. 135-142 (1993).

[32] Cutler, L. D., Fröhlich, B., and Hanrahan, P., "Two-handed Direct Manipulation on the Responsive Workbench," Proceedings of the 1997 Symposium on Interactive

3D Graphics, New York, NY, USA, pp. 107-114 (1997).

[33] Dixon, M. W., Wraga, M., Proffitt, D. R., and Williams, G. C., "Eye Height Scaling of Absolute Size in Immersive and Nonimmersive Displays.," Journal of Experimental Psychology: Human Perception and Performance, vol. 26, no. 2, 582-593 (2000).

[34] Ellis, S. R., "Nature and origins of virtual environments: a bibliographical essay," Computing Systems in Engineering, vol. 2, no. 4, 321-347 (1991).

[35] Fitts, P. M., "The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement," Journal of Experimental Psychology, vol. 47, no. 6, 381-391 (1954).

[36] Forsberg, A., Herndon, K., and Zeleznik, R., "Aperture Based Selection for Immersive Virtual Environments," Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 95-96 (1996).

[37] Frees, S. and Kessler, G. D., "Precise and Rapid Interaction Through Scaled Manipulation in Immersive Virtual Environments," Proceedings of the IEEE Conference on Virtual Reality, 2005. VR '05., Washington, DC, USA, pp. 99-106 (2005).

[38] Frees, S., Kessler, G. D., and Kay, E., "PRISM Interaction for Enhancing Control in Immersive Virtual Environments," ACM Trans. Comput.-Hum. Interact., vol. 14, no. 1, 1-31 (2007).

[39] Geuss, M. N., Stefanucci, J. K., Creem-Regehr, S. H., Thompson, W. B., and Mohler, B. J., "Effect of Display Technology on Perceived Scale of Space," Human Factors: The Journal of the Human Factors and Ergonomics Society, vol. 57, no. 7, 1235-1247 (2015).

[40] Gibbs, C. B., "Controller design: Interactions of controlling limbs, time-lags, and gains in positional and velocity systems," Ergonomics, vol. 5, no. 2, 385-402 (1962).

[41] Greunke, L. and Sadagic, A., "Taking Immersive VR Leap in Training of Landing Signal Officers," IEEE Transactions on Visualization and Computer Graphics, vol. 22, no. 4, 1482-1491 (2016).

[42] Grossman, T. and Balakrishnan, R., "Pointing at Trivariate Targets in 3D Environments," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 447-454 (2004).

[43] Grossman, T. and Balakrishnan, R., "The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 281-290 (2005).

[44] Grossman, T. and Balakrishnan, R., "The Design and Evaluation of Selection Techniques for 3D Volumetric Displays," Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 3-12 (2006).

[45] Grossman, T., Wigdor, D., and Balakrishnan, R., "Multi-finger Gestural Interaction with 3D Volumetric Displays," Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 61-70 (2004).

[46] Guiard, Y., "Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model," Journal of Motor Behavior, vol. 19, no. 4, 486-517 (1987).

[47] Hackenberg, G., McCall, R., and Broll, W., "Lightweight Palm and Finger Tracking for Real-time 3D Gesture Control," Proceedings of the 2011 IEEE Virtual Reality Conference, Washington, DC, USA, pp. 19-26 (2011).

[48] Hauptmann, A. G., "Speech and Gestures for Graphic Image Manipulation," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 241-245 (1989).

[49] Hilliges, O. et al., "Interactions in the Air: Adding Further Depth to Interactive Tabletops," Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 139-148 (2009).

[50] Hinckley, K., Pausch, R., Goble, J. C., and Kassell, N. F., "A Survey of Design Issues in Spatial Input," Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 213-222 (1994).

[51] Hinckley, K., Pausch, R., and Proffitt, D., "Attention and Visual Feedback: The Bimanual Frame of Reference," Proceedings of the 1997 Symposium on Interactive 3D Graphics, New York, NY, USA, pp. 121-126 (1997).

[52] Hinckley, K., Pausch, R., Proffitt, D., and Kassell, N. F., "Two-handed Virtual Manipulation," ACM Trans. Comput.-Hum. Interact., vol. 5, no. 3, 260-302 (1998).

[53] Hinckley, K., Pausch, R., Proffitt, D., Patten, J., and Kassell, N., "Cooperative Bimanual Action," Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 27-34 (1997).

[54] ISO 9241-9, "Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 9: Requirements for non-keyboard input devices," International Organization for Standardization, (2000).

[55] Jang, Y., Noh, S.-T., Chang, H. J., Kim, T.-K., and Woo, W., "3D Finger CAPE: Clicking Action and Position Estimation under Self-Occlusions in Egocentric Viewpoint," Visualization and Computer Graphics, IEEE Transactions on, vol. 21, no. 4, 501-510 (2015).

[56] Kabbash, P., Buxton, W., and Sellen, A., "Two-handed Input in a Compound Task," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 417-423 (1994).

[57] Keijser, J., Carpendale, S., Hancock, M., and Isenberg, T., "Exploring 3D Interaction in Alternate Control-Display Space Mappings," 3D User Interfaces, 2007. IEEE Symposium on, pp. 17-24 (2007).

[58] Kenyon, R. V., Sandin, D., Smith, R. C., Pawlicki, R., and Defanti, T., "Size-Constancy in the CAVE," Presence: Teleoper. Virtual Environ., vol. 16, no. 2, 172-187 (2007).

[59] Kim, H., Albuquerque, G., Havemann, S., and Fellner, D. W., "Tangible 3D: Hand Gesture Interaction for Immersive 3D Modeling," Proceedings of the 11th Eurographics Conference on Virtual Environments, Aire-la-Ville, Switzerland, Switzerland, pp. 191-199 (2005).

[60] König, W. A., Gerken, J., Dierdorf, S., and Reiterer, H., "Adaptive Pointing — Design and Evaluation of a Precision Enhancing Technique for Absolute Pointing Devices," Proceedings of the 12th IFIP TC 13 International Conference on Human-Computer Interaction: Part I, Berlin, Heidelberg, pp. 658-671 (2009).

[61] Kopper, R., Bacim, F., and Bowman, D. A., "Rapid and Accurate 3D Selection by Progressive Refinement," 3D User Interfaces (3DUI), 2011 IEEE Symposium on, pp. 67-74 (2011).

[62] Kopper, R., Bowman, D. A., Silva, M. G., and McMahan, R. P., "A Human Motor Behavior Model for Distal Pointing Tasks," Int. J. Hum.-Comput. Stud., vol. 68, no. 10, 603-615 (2010).

[63] Kopper, R., Ni, T., Bowman, D. A., and Pinho, M., "Design and Evaluation of Navigation Techniques for Multiscale Virtual Environments," Virtual Reality Conference, 2006. Proceedings. IEEE, pp. 175-182 (2006).

[64] Kulik, A. et al., "C1x6: A Stereoscopic Six-user Display for Co-located Collaboration in Shared Virtual Environments," Proceedings of the 2011

SIGGRAPH Asia Conference, Hong Kong, China, pp. 188:1-188:12 (2011).

[65] Latta, J. N. and Oberg, D. J., "A Conceptual Virtual Reality Model," IEEE Computer Graphics and Applications, vol. 14, no. 1, 23-29 (1994).

[66] LaViola Jr., J. J. and Katzourin, M., "An Exploration of Non-Isomorphic 3D Rotation in Surround Screen Virtual Environments," 2007 IEEE Symposium on 3D User Interfaces, Charlotte NC, USA, pp. 49-54 (2007).

[67] Leap Motion | Mac & PC Motion Controller for Games, Design, Virtual Reality & More. https://www.leapmotion.com/

[68] Leganchuk, A., Zhai, S., and Buxton, W., "Manual and Cognitive Benefits of Two-handed Input: An Experimental Study," ACM Trans. Comput.-Hum. Interact., vol. 5, no. 4, 326-359 (1998).

[69] Leigh, J., Johnson, A. E., Vasilakis, C. A., and DeFanti, T. A., "Multi-Perspective Collaborative Design in Persistent Networked Virtual Environments," Virtual Reality Annual International Symposium, 1996., Proceedings of the IEEE 1996, pp. 253-260 (1996).

[70] Lemmerman, D. K. and LaViola Jr., J. J., "An Exploration of Interaction-Display Offset in Surround Screen Virtual Environments," 3D User Interfaces, 2007. 3DUI '07. IEEE Symposium on, pp. 9-15 (2007).

[71] Liang, J. and Green, M., "JDCAD: A highly interactive 3D modeling system," Computers & Graphics, vol. 18, no. 4, 499-506 (1994).

[72] Li, J., Cho, I., and Wartell, Z., "Evaluation of 3D Virtual Cursor Offset Techniques for Navigation tasks in a Multi-Display Virtual Environment," 3D User Interfaces (3DUI), 2015 IEEE Symposium on, pp. 59-66 (2015).

[73] Liu, G., Chua, R., and Enns, J. T., "Attention for perception and action: task interference for action planning, but not for online control," Experimental Brain Research, vol. 185, no. 4, 709-717 (2007).

[74] Lubos, P., Bruder, G., and Steinicke, F., "Analysis of Direct Selection in Head-Mounted Display Environments," 3D User Interfaces (3DUI), 2014 IEEE Symposium on, pp. 11-18 (2014).

[75] MacKenzie, I. S., "Fitts' Law As a Research and Design Tool in Human-computer Interaction," Hum.-Comput. Interact., vol. 7, no. 1, 91-139 (1992).

[76] MacKenzie, I. S., "Input Devices and Interaction Techniques for Advanced Computing," in Virtual Environments and Advanced Interface Design, Woodrow Barfield and Thomas A. Furness III, Eds., Oxford University Press, Inc., 437-470

(1995).

[77] MacKenzie, S. I. and Buxton, W., "Extending Fitts' Law to Two-dimensional Tasks," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Monterey, California, USA, pp. 219-226 (1992).

[78] MacKenzie, I. S. and Isokoski, P., "Fitts' Throughput and the Speed-accuracy Tradeoff," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 1633-1636 (2008).

[79] MacKenzie, I. S., Kauppinen, T., and Silfverberg, M., "Accuracy Measures for Evaluating Computer Pointing Devices," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 9-16 (2001).

[80] MacKenzie, C. L., Marteniuk, R. G., Dugas, C., Liske, D., and Eickmeier, B., "Three-dimensional movement trajectories in Fitts' task: Implications for control," The Quarterly Journal of Experimental Psychology Section A: Human Experimental Psychology, vol. 39, no. 4, 629-647 (1987).

[81] MacKenzie, I. S. and Riddersma, S., "Effects of output display and control—display gain on human performance in interactive systems," Behaviour & Information Technology, vol. 13, no. 5, 328-337 (1994).

[82] Mackinlay, J. D., Card, S. K., and Robertson, G. G., "Rapid Controlled Movement Through a Virtual 3D Workspace," Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA, pp. 171-176 (1990).

[83] Malik, S. and Laszlo, J., "Visual Touchpad: A Two-handed Gestural Input Device," Proceedings of the 6th International Conference on Multimodal Interfaces, New York, NY, USA, pp. 289-296 (2004).

[84] Mapes, D. P. and Moshell, J. M., "A Two-handed Interface for Object Manipulation in Virtual Environments," Presence: Teleoper. Virtual Environ., vol. 4, no. 4, 403-416 (1995).

[85] McMahan, R. P., Gorton, D., Gresock, J., McConnell, W., and Bowman, D. A., "Separating the Effects of Level of Immersion and 3D Interaction Techniques," Proceedings of the ACM Symposium on Virtual Reality Software and Technology, New York, NY, USA, pp. 108-111 (2006).

[86] Mendes, D., Fonseca, F., Araujo, B., Ferreira, A., and Jorge, J., "Mid-Air Interactions Above Stereoscopic Interactive Tables," 3D User Interfaces (3DUI), 2014 IEEE Symposium on, pp. 3-10 (2014).

[87] Mine, M. R., "Virtual Environment Interaction Techniques," University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, Tech. rep. (1995).

[88] Mine, M. R., Brooks Jr., F. P., and Sequin, C. H., "Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction," Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA, pp. 19-26 (1997).

[89] Mlyniec, P. et al., "iMedic: A Two-Handed Immersive Medical Environment for Distributed Interactive Consultation," Studies in Health Technology and Informatics: Medicine Meets Virtual Reality 18, vol. 163, 372-378 (2011).

[90] Murata, A. and Iwase, H., "Extending Fitts' law to a three-dimensional pointing task ," Human Movement Science , vol. 20, no. 6, 791-805 (2001).

[91] Naceri, A. and Chellali, R., "Depth Perception Within Peripersonal Space Using Head-mounted Display," Presence: Teleoperators and Virtual Environments, vol. 20, no. 3, 254-272 (2011).

[92] Natapov, D., Castellucci, S. J., and MacKenzie, I. S., "ISO 9241-9 Evaluation of Video Game Controllers," Proceedings of Graphics Interface 2009, Kelowna, British Columbia, Canada, pp. 223-230 (2009).

[93] Oculus Rift | Oculus. https://www.oculus.com/en-us/rift/

[94] OpenSceneGraph. www.openscenegraph.org

[95] Osawa, N., "Automatic Adjustments for Efficient and Precise Positioning and Release of Virtual Objects," Proceedings of the 2006 ACM International Conference on Virtual Reality Continuum and Its Applications, New York, NY, USA, pp. 121-128 (2006).

[96] Owen, R., Kurtenbach, G., Fitzmaurice, G., Baudel, T., and Buxton, B., "When It Gets More Difficult, Use Both Hands: Exploring Bimanual Curve Manipulation," Proceedings of Graphics Interface 2005, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, pp. 17-24 (2005).

[97] Paljic, A., Coquillart, S., Burkhardt, J.-M., and Richard, P., "A Study of Distance of Manipulation on the Responsive Workbench," Proc. of the 7th Annual Immersive Projection Technology Symposium, pp. 251-258 (2002).

[98] Pausch, R., Burnette, T., Brockway, D., and Weiblen, M. E., "Navigation and Locomotion in Virtual Worlds via Flight into Hand-held Miniatures," Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA, pp. 399-400 (1995).

[99] Pierce, J. S. et al., "Image Plane Interaction Techniques in 3D Immersive Environments," Proceedings of the 1997 Symposium on Interactive 3D Graphics, New York, NY, USA, pp. 39-43 (1997).

[100] Pierce, J. S. and Pausch, R., "Comparing Voodoo Dolls and HOMER: Exploring the Importance of Feedback in Virtual Environments," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 105-112 (2002).

[101] Pierce, J. S., Stearns, B. C., and Pausch, R., "Voodoo Dolls: Seamless Interaction at Multiple Scales in Virtual Environments," Proceedings of the 1999 Symposium on Interactive 3D Graphics, New York, NY, USA, pp. 141-145 (1999).

[102] Poupyrev, I., Billinghurst, M., Weghorst, S., and Ichikawa, T., "The Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR," Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 79-80 (1996).

[103] Poupyrev, I. and Ichikawa, T., "Manipulating Objects in Virtual Worlds: Categorization and Empirical Evaluation of Interaction Techniques," Journal of Visual Languages & Computing, vol. 10, no. 1, 19-35 (1999).

[104] Poupyrev, I., Ichikawa, T., Weghorst, S., and Billinghurst, M., "Egocentric Object Manipulation in Virtual Environments: Empirical Evaluation of Interaction Techniques," Computer Graphics Forum, vol. 17, no. 3, 41-52 (1998).

[105] Poupyrev, I., Weghorst, S., Billinghurst, M., and Ichikawa, T., "A Framework and Testbed for Studying Manipulation Techniques for Immersive VR," Proceedings of the ACM Symposium on Virtual Reality Software and Technology, New York, NY, USA, pp. 21-28 (1997).

[106] Poupyrev, I., Weghorst, S., and Fels, S., "Non-Isomorphic 3D Rotational Techniques," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, The Hague, The Netherlands, pp. 540-547 (2000).

[107] Razer Hydra. http://sixense.com/razerhydra

[108] Rheingold, H., Virtual Reality, New York: Secker & Warburg (1991).

[109] Riyal, R., Patel, A. D., Murugesan, T., Turini, G., and Young, J. G., "Effect of control-display transfer function on pointing performance for a hand/finger based touchless gestural controls: a preliminary investigation," Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 59, no. 1, 1085-1089 (2015).

[110] Robinett, W. and Holloway, R., "Implementation of Flying, Scaling and Grabbing in Virtual Worlds," Proceedings of the 1992 Symposium on Interactive 3D

Graphics, New York, NY, USA, pp. 189-192 (1992).

[111] Robinett, W. and Holloway, R., "The Visual Display Transformation for Virtual Reality," Presence: Teleoper. Virtual Environ., vol. 4, no. 1, 1-23 (1995).

[112] Sato, Y., Saito, M., and Koike, H., "Real-Time Input of 3D Pose and Gestures of a User's Hand and Its Applications for HCI," Proceedings of the 2001 IEEE Virtual Reality Conference, pp. 79-86 (2001).

[113] Schlattmann, M. and Klein, R., "Efficient Bimanual Symmetric 3D Manipulation for Markerless Hand-Tracking," Proceedings of the Virtual Reality International Conference, VRIC 2009. (2009).

[114] Schlattmann, M. and Klein, R., "Simultaneous 4 Gestures 6 DOF Real-time Two-hand Tracking Without Any Markers," Proceedings of the 2007 ACM Symposium on Virtual Reality Software and Technology, New York, NY, USA, pp. 39-42 (2007).

[115] Schmandt, C., "Spatial Input/Display Correspondence in a Stereoscopic Computer Graphic Work Station," SIGGRAPH Comput. Graph., vol. 17, no. 3, 253-261 (1983).

[116] Schultheis, U., Jerald, J., Toledo, F., Yoganandan, A., and Mlyniec, P., "Comparison of a Two-Handed Interface to a Wand Interface and a Mouse Interface for Fundamental 3D Tasks," 3D User Interfaces (3DUI), 2012 IEEE Symposium on, pp. 117-124 (2012).

[117] Shaw, C. D., "Pain and Fatigue in Desktop VR: Initial Results," Proceedings of the Graphics Interface 1998 Conference, June 18-20, 1998, Vancouver, BC, Canada, pp. 185-192 (1998).

[118] Shaw, C. and Green, M., "Two-handed Polygonal Surface Design," Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 205-212 (1994).

[119] Slater, M., Usoh, M., and Steed, A., "Taking Steps: The Influence of a Walking Technique on Presence in Virtual Reality," ACM Trans. Comput.-Hum. Interact., vol. 2, no. 3, 201-219 (1995).

[120] Song, P., Goh, W. B., Hutama, W., Fu, C.-W., and Liu, X., "A Handle Bar Metaphor for Virtual Object Manipulation with Mid-air Interaction," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 1297-1306 (2012).

[121] Song, D. and Norman, M., "Nonlinear Interactive Motion Control Techniques for Virtual Space Navigation," Virtual Reality Annual International Symposium, 1993

IEEE, pp. 111-117 (1993).

[122] Soukoreff, R. W. and MacKenzie, I. S., "Towards a Standard for Pointing Device Evaluation, Perspectives on 27 Years of Fitts' Law Research in HCI," Int. J. Hum.-Comput. Stud., vol. 61, no. 6, 751-789 (2004).

[123] Southard, D. A., "Viewing model for virtual environment displays," Journal of Electronic Imaging, vol. 4, no. 4, 413-420 (1995).

[124] Stanney, K., "Realizing the Full Potential of Virtual Reality: Human Factors Issues That Could Stand in the Way," Virtual Reality Annual International Symposium, 1995. Proceedings., Washington, DC, USA, pp. 28-34 (1995).

[125] Steinicke, F., Ropinski, T., and Hinrichs, K., "Object Selection in Virtual Environments with an Improved Virtual Pointer Metaphor," in Computer Vision and Graphics: International Conference, ICCVG 2004, Warsaw, Poland, September 2004, Proceedings, K. Wojciechowski et al., Eds., Springer Netherlands, vol. 32, 320-326 (2006).

[126] Stoakley, R., Conway, M. J., and Pausch, R., "Virtual Reality on a WIM: Interactive Worlds in Miniature," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 265-272 (1995).

[127] Suma, E. A., Finkelstein, S. L., Clark, S., Goolkasian, P., and Hodges, L. F., "Effects of Travel Technique and Gender on a Divided Attention Task in a Virtual Environment," 3D User Interfaces (3DUI), 2010 IEEE Symposium on, pp. 27-34 (2010).

[128] Sutherland, I. E., "A Head-mounted Three Dimensional Display," Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, Part I, New York, NY, USA, pp. 757-764 (1968).

[129] Tan, D. S., Robertson, G. G., and Czerwinski, M., "Exploring 3D Navigation: Combining Speed-coupled Flying with Orbiting," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 418-425 (2001).

[130] Taylor, F. V., "Psychology and the design of machines," The American Psychologist, vol. 12, no. 5, 249-258 (1957).

[131] Teather, R. J. and Stuerzlinger, W., "Factors Affecting Mouse-Based 3D Selection in Desktop VR Systems," Proceedings of the 3rd ACM Symposium on Spatial User Interaction, New York, NY, USA, pp. 10-19 (2015).

[132] Teather, R. J. and Stuerzlinger, W., "Pointing at 3D Target Projections with One-eyed and Stereo Cursors," Proceedings of the SIGCHI Conference on Human

Factors in Computing Systems, New York, NY, USA, pp. 159-168 (2013).

[133] Teather, R. J. and Stuerzlinger, W., "Pointing at 3D Targets in a Stereo Head-Tracked Virtual Environment ," 3D User Interfaces (3DUI), 2011 IEEE Symposium on, pp. 87-94 (2011).

[134] Teather, R. J. and Stuerzlinger, W., "Visual Aids in 3D Point Selection Experiments," Proceedings of the 2nd ACM Symposium on Spatial User Interaction, New York, NY, USA, pp. 127-136 (2014).

[135] Unity - Game Engine. https://unity3d.com/

[136] Vanacken, L., Grossman, T., and Coninx, K., "Exploring the Effects of Environment Density and Target Visibility on Object Selection in 3D Virtual Environments," 3D User Interfaces, 2007. 3DUI '07. IEEE Symposium on, pp. 115-122 (2007).

[137] Vogel, D. and Balakrishnan, R., "Distant Freehand Pointing and Clicking on Very Large, High Resolution Displays," Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 33-42 (2005).

[138] Wang, Y. and MacKenzie, C. L., "Effects of Orientation Disparity Between Haptic and Graphic Displays of Objects in Virtual Environments," INTERACT '99: Proceedings of Seventh IFIP Conference on Human-Computer Interaction, pp. 391-398 (1999).

[139] Wang, R., Paris, S., and Popović, J., "6D Hands: Markerless Hand-tracking for Computer Aided Design," Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 549-558 (2011).

[140] Wang, R. Y. and Popović, J., "Real-time Hand-tracking with a Color Glove," ACM SIGGRAPH 2009 Papers, New York, NY, USA, pp. 63:1--63:8 (2009).

[141] Ware, C., "Dynamic Stereo Displays," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 310-316 (1995).

[142] Ware, C., "Using Hand Position for Virtual Object Placement," Vis. Comput., vol. 6, no. 5, 245-253 (1990).

[143] Ware, C., Arthur, K., and Booth, K. S., "Fish Tank Virtual Reality," Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 37-42 (1993).

[144] Ware, C. and Balakrishnan, R., "Reaching for Objects in VR Displays: Lag and Frame Rate," ACM Trans. Comput.-Hum. Interact., vol. 1, no. 4, 331-356 (1994).

[145] Ware, C., Gobrecht, C., and Paton, M. A., "Dynamic Adjustment of Stereo Display Parameters," IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 28, no. 1, 56-65 (1998).

[146] Ware, C. and Osborne, S., "Exploration and Virtual Camera Control in Virtual Three Dimensional Environments," Proceedings of the 1990 Symposium on Interactive 3D Graphics, New York, NY, USA, pp. 175-183 (1990).

[147] Wartell, Z. et al., "Interaction Volume Management in a Multi-scale Virtual Environment," in Advances in Information and Intelligent Systems, Zbigniew W. Ras and William Ribarsky, Eds., Springer Berlin Heidelberg, vol. 251, 327-349 (2009).

[148] Wartell, Z., Ribarsky, W., and Hodges, L., "Third-Person Navigation of Whole-Planet Terrain in a Head-tracked Stereoscopic Environment," Virtual Reality, 1999. Proceedings., IEEE, pp. 141-148 (1999).

[149] Weimer, D. and Ganapathy, S. K., "A Synthetic Visual Environment with Hand Gesturing and Voice Input," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 235-240 (1989).

[150] Wilkes, C. and Bowman, D. A., "Advantages of Velocity-based Scaling for Distant 3D Manipulation," Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology, New York, NY, USA, pp. 23-29 (2008).

[151] Wilson, A. D., "Robust Computer Vision-based Detection of Pinching for One and Two-handed Gesture Input," Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, pp. 255-258 (2006).

[152] Zeleznik, R. C., Forsberg, A. S., and Strauss, P. S., "Two Pointer Input for 3D Interaction," Proceedings of the 1997 Symposium on Interactive 3D Graphics, New York, NY, USA, pp. 115-120 (1997).

[153] Zhai, S., Buxton, W., and Milgram, P., "The "Silk Cursor": Investigating Transparency for 3D Target Acquisition," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 459-464 (1994).

[154] Zhai, S., Kong, J., and Ren, X., "Speed-accuracy Tradeoff in Fitts' Law Tasks: On the Equivalency of Actual and Nominal Pointing Precision," International Journal of Human-Computer Studies , vol. 61, no. 6, 823-856 (2004), Fitts' law 50 years

later: applications and contributions from human-computer interaction.

[155] Zhai, S. and Milgram, P., "Human Performance Evaluation of Manipulation Schemes in Virtual Environments," Virtual Reality Annual International Symposium, 1993 IEEE, pp. 155-161 (1993).

[156] Zhai, S. and Milgram, P., "Quantifying Coordination in Multiple DOF Movement and Its Application to Evaluating 6 DOF Input Devices," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 320-327 (1998).

[157] Zhai, S., Milgram, P., and Buxton, W., "The Influence of Muscle Groups on Performance of Multiple Degree-of-freedom Input," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, pp. 308-315 (1996).

APPENDIX A: EXPERIMENT DOCUMENTS FOR CHAPTER 4

UNC CHARLOTTE

Research and Economic Development
Office of Research Compliance
9201 University City Boulevard, Charlotte, NC 28223-0001
t/ 704.687.1876 f/ 704.687.0980 http://research.uncc.edu/compliance-ethics

Institutional Review Board (IRB) for Research with Human Subjects

Certificate of Approval

Protocol #	**13-10-52**		
Protocol Type:	**Expedited**	7	
Title:	**Evaluating Usability of a View Direction Cursor Offset Technique for a Navigation Task in a Multi-Display Virtual Environment**		
Initial Approval:	**11/26/2013**		
Responsible Faculty	**Dr. Zachary**	**Wartell**	**Computer Science**
Investigator	**Mr. Isaac**	**Cho**	**Computer Science**
Co-investigator	**Mr. Jialei**	**Li**	**Computer Science**

After careful review, the protocol listed above was approved by the Institutional Review Board (IRB) for Research with Human Subjects under 45 CFR 46.111. This approval will expire one year from the date of this letter. In order to continue conducting research under this protocol after one year, the "Annual Protocol Renewal Form" must be submitted to the IRB. This form can be obtained from the Office of Research Compliance web page http://research.uncc.edu/compliance-ethics/human-subjects.

Please note that it is the investigator's responsibility to promptly inform the committee of any changes in the proposed research prior to implementing the changes, and of any adverse events or unanticipated risks to subjects or others.

Amendment and Event Reporting forms are available on our web page at: http://research.uncc.edu/compliance-ethics/human-subjects/amending-your-protocol.

M. L. [signature] 12/6/13
Dr. M. Lyn Exum, IRB Chair Date

UNC CHARLOTTE
The University of North Carolina at Charlotte

Informed Consent for
<u>Evaluating Usability of View Direction Cursor Offset Techniques for a Navigation Task in a Multi-Display Virtual Environment</u>

Project Purpose

In this study we will determine usability of cursor offset techniques for a one-handed navigation technique in a CAVE that consists of three large, back-projected displays.

Investigators

Isaac Cho, Computer Science
Jialei Li, Computer Science
Zachary Wartell, Computer Science

Eligibility

You may participate in this study if you are above 18 and if you have 20/20 vision or corrected vision to 20/20 (i.e. you can clearly read text on a computer monitor), can comfortably use your arms and fingers and communicate in spoken English.

Overall Description of Participation

In the first step, we will demonstrate how to use a CAVE system. This system uses three large stereoscopic displays and head-tracking to display 3D computer graphics. Stereopsis is an important depth cue generated by the differences between the two views of a scene seen from a person's two eyes. Head-tracking is a tracking system which tracks the position and orientation of person's head to generate an optimal perspective 3D image. We will demonstrate how these technologies can be used with a 3D application in the CAVE. We will train you on how these technologies work by showing you how to view the 3D application and how to perform a navigation task (rotating and translating) with either one-handed or two-handed navigation technique using 3D input devices, called 6DOF ("degree-of-freedom") buttonball. The instructor will let you know which navigation technique you will use for the experiment. We will also survey your past experience with 3D computer programs and 3D media (e.g. 3D games and 3D movies) and your familiarity with using a computer and various user interfaces.

Next, you will perform a docking task in the virtual environment using 4 different variations of the user interface. A pair of the 3D cursors will appear in the virtual environment representing your hands positions. The four variations differ in their software mechanisms used to relate your hands physical position to the 3D cursor positions. In particular they differ on how much of a translational offset there is between your hands and the 3D cursors.

The target-finding task requires finding a colored box (i.e. target box) on a gridded ground plane and placing the box so that it contains a second docking box. The docking box is located in the center of a randomly chosen display (one of the three). You need to move the target box to the docking box by adjusting its location and orientation. You, also, need to match the colors of the faces of the target and docking boxes. When complete, you click a button on the buttonball to finish the task. You will repeat this task 30 times using each of the 3 offset methods. After each block of 30 trials, we will ask you questions using a questionnaire about your experience during that block of trials.

After you finish the experiment, you will take a final, post-experiment questionnaire.

Length of Participation

Participation should take approximately 50-60 minutes.

Risks and Benefits of Participation

When using virtual reality systems, some people may experience symptoms of disorientation, nausea, or dizziness. These can be similar to "motion sickness" or the feeling experienced in wide-screen movies and theme park rides. We do not expect these to be strong or to last after participants leave the laboratory. If you feel uncomfortable and wish to stop the experiment, you are free to do so at any time without penalty by simply announcing their desire to stop.

The primary scientific benefit of the study is to compare the usability of cursor offset techniques for one-handed navigationt technique in a CAVE.

The benefit for participants who are students in ITCS 4120 Computer Graphics is to get a chance to particpant in a graduate research project that uses real-time 3D computer graphics in an advanced virtual reality system. 4120 students will receive extra credit for participating in this experiment, and have the option of having their data not collected for analysis. This meets University IRB protocol requirements for course extra credit. The informed consent form below has a special section for ITCS 4120 students informing them of this option.

Volunteer Statement

You are a volunteer. The decision to participate in this study is completely up to you. If you decide to participate, you may stop at any time. You will not be treated any differently if you decide not to participate in the study or if you stop once you have started. You can request to withdraw your segment after the testing is complete.

If you are enrolled in ITCS 4120, your participation in this study will count as an extra credit assignment. Further, having your data recorded and used for this research study is voluntary.

Confidentiality Statement

Any information about your participation, including your identity, is completely confidential. The following efforts will be taken to protect confidentiality:

1) The consent form will be kept in a locked filing cabinet, separate from the rest of data.
2) All participants will be assigned a random ID consisting two randomly-generated initials (initials will not correspont to participants' name). The participants will only be referred by assigned alphanumeric codes both in internal communication between researchers or in the form of written reports.
3) The investigator and co-investigators will ask the participants not to mention their name or identify themselves during the recordings. The recording is only for internal use such

as transcription and will not be made available to the public. Screenshots from the video recording might be published without disclosing the identify of any participants.

4) All digitally recorded files during the study will be kept in the Charlotte Visualization Center (room 437 in Woodward Hall) on password-protected computers. The files will be destroyed after two years by investigators under the guidance of the responsible faculty.

Statement of Fair Treatment and Respect

UNC Charlotte wants to make sure that you are treated in a fair and respectful manner. Contact the university's Research Compliance Office (704-687-3309) if you have questions about how you are treated as a study participant. If you have any questions about the actual project or study, please contact Isaac Cho (icho1@uncc.edu), Jialei Li (jli42@uncc.edu) or Dr. Zachary Wartell (zwartell@uncc.edu) at 704-687-8442.

Approval Date

This form was approved on Nov XX, 2013 for use for one year.

Participant Consent

I have read the contents in this consent form. I have had the chance to ask questions about this study, and those questions have been answered to my satisfaction. I am at least 18 years of age, and I agree to participate in this research project. I understand that I will receive a copy of this form after it has been signed by me and the principal investigator of this study.

If you are a student in ITCS 4120, please also complete the following box:

I am a student in ITCS 4120. I understand that participating in the experiment is an extra credit assignment. Further, the collection of my data for use in the research study is also voluntary and my choice on whether to have the data collected will not affect my extra credit grade.

Please circle one option:

As an ITCS 4120 student, I [do] | [do not] consent to the collection of my data for use in the research study.

________________________________ ____________________

Participant Name (PRINT) DATE

____

Participant Signature

________________________________ ____________________

Investigator Signature DATE

Pre-questionnaire

1. Your Given ID (Instructor only):

2. Your age:

3. Your gender:

4. Occupational Status:

    Undergraduate student ___

    Master Student ___

    PhD Student ___

    Research Assistant/Fellow ___

    Staff-systems, technical ___

    Faculty ___

    Administrative Staff ___

    Other: __________________

5. Your major:

6. Are you colorblind? Yes / No

7. Do you have 20/20 eyesight (or corrected 20/20)? Yes / No

8. Do you have any disabilities or injuries that might limit your ability to use either your left or right arm, hand and/or fingers in every tasks such as writing, painting, using a computer mouse or advanced game controller? Yes / No

9. How much arm or hand fatigue did you feel right now?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

10. Are you familiar with using a mouse and keyboard? Yes / No

11. Have you ever felt motion sick (dizziness or nausea) while playing a computer game or viewing a large, screen movie before? Yes / No

1. How often do you use a computer in your daily activities?

(Never) (A Great Deal)

1 2 3 4 5 6 7

2. How often do you play 2D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

3. How often do you play 3D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

4. How often do you play computer games (of any kind) on a computer/PC?

(Never) (A Great Deal)

1 2 3 4 5 6 7

5. How often do you play computer games using a game console, such as Nintendo®, Xbox®, Sony PlayStation® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

6. How often do play computer games using a game console with a motion capture device, such as Xbox Kinect®, Sony PlayStation Move® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

Definition: Stereoscopic 3D

Stereoscopic 3D refers to a display that creates a true 3D image that appears to pop-out in front of and behind the screen. These displays are used in some movie theaters, television sets and computer monitors. Most stereoscopic 3D display technologies known to consumers require they wear special glasses.

7. How often do you watch stereoscopic 3D movies in the theater?

(Never)						(A Great Deal)
1	2	3	4	5	6	7

8. How often do you play computer games or watch movies on an in-home television using stereoscopic 3D?

(Never)						(A Great Deal)
1	2	3	4	5	6	7

Definition: 3D User Interface

A “3D user interface” is a human-computer interface where the user views 3D computer graphics and interacts with those graphics by traveling through the 3D environment, and/or manipulating and changing the 3d environment. 3D user interfaces may or may not use stereoscopic 3D displays. Also 3D user interfaces may or may not use advanced 3D input devices such as the Microsoft Kinect, PlayStation Move, Nintendo Wii, etc.

9. If you have used any 3D user interfaces before, then describe what 3D user interfaces you have used and mention what type of display and input device technology you used with them.

____

____

____

____

Arm Fatigue Ratings (For Experiment 1)

Your Given ID (Instructor only):

1. How much arm fatigue did you feel with the **No offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

2. How much arm fatigue did you feel with the **Fixed-length offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

3. How much arm fatigue did you feel with the **Go-Go technique offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

4. How much arm fatigue did you feel with the **Torso position offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

Post-questionnaire (For Experiment 1)

Your Given ID (Instructor only):

Use the list below for reference when answering the following questions. You may write just the letter (A-D) that labels the offset technique you want to refer to in your answers.

Conditions:

A. No offset
B. Fixed-Length offset
C. Go-Go technique offset
D. Torso position offset

1. Which condition (A to D) was the easiest to accomplish the overall **trial**? Why?

____
____
____

2. Which condition (A to D) was the easiest to accomplish the navigation task when the target box is **behind the docking box**? Why?

____
____
____

3. Which condition (A to D) was the easiest to accomplish the navigation task when the target box is **in front of the docking box**? Why?

____
____
____

4. Does the **target box size** make any difference in completing the task? Why?

____
____
____

If you have any comments for this study, please give us feedback.

____
____
____
____

Arm Fatigue Ratings (For Experiment 2)

Your Given ID (Instructor only):

1. How much arm fatigue did you feel with the **no offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

2. How much arm fatigue did you feel with the **short offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

3. How much arm fatigue did you feel with the **medium offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

4. How much arm fatigue did you feel with the **long offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

Post-questionnaire (For Experiment 2)

Your Given ID (Instructor only):

Use the list below for reference when answering the following questions. You may write just the letter (A-D) that labels the offset technique you want to refer to in your answers.

Conditions:

A. No offset
B. Torso Position with short offset
C. Torso Position with medium offset
D. Torso Position with long offset

1. Which condition (A to D) was the easiest to accomplish the overall **trial**? Why?

2. Which condition (A to D) was the easiest to accomplish the navigation task when the target box is **behind the docking box**? Why?

3. Which condition (A to D) was the easiest to accomplish the navigation task when the target box is **in front of the docking box**? Why?

4. Does the **target box size** make any difference in completing the task? Why?

If you have any comments for this study, please give us feedback.

APPENDIX B: EXPERIMENT DOCUMENTS FOR CHAPTER 5

UNC CHARLOTTE

Research and Economic Development
Office of Research Compliance
9201 University City Blvd, Charlotte, NC 28223-0001
t/ 704.687.1876 f/ 704.687.0980 http://research.uncc.edu/compliance-ethics

Institutional Review Board (IRB) for Research with Human Subjects

Certificate of Approval

Protocol #	**15-04-33**		
Protocol Type:	**Expedited**	7	
Title:	**Evaluating Usability of Virtual Cursor Offset Techniques for a Navigation Task in a Multi-Display Virtual Environment**		
Initial Approval:	**6/8/2015**		
Student Investigator	**Mr. Jialei**	**Li**	**Computer Science**
Responsible Faculty	**Dr. Zachary**	**Wartell**	**Computer Science**
Investigator	**Dr. Isaac**	**Cho**	**Computer Science**

After careful review, the protocol listed above was approved by the Institutional Review Board (IRB) for Research with Human Subjects under 45 CFR 46.111. This approval will expire one year from the date of this letter. In order to continue conducting research under this protocol after one year, the "Annual Protocol Renewal Form" must be submitted to the IRB. This form can be obtained from the Office of Research Compliance web page http://research.uncc.edu/compliance-ethics/human-subjects.

Please note that it is the investigator's responsibility to promptly inform the committee of any changes in the proposed research prior to implementing the changes, and of any adverse events or unanticipated risks to subjects or others.

Amendment and Event Reporting forms are available on our web page at: http://research.uncc.edu/compliance-ethics/human-subjects/amending-your-protocol.

M. Ex 6/11/15

Dr. M. Lyn Exum, IRB Chair Date

UNC CHARLOTTE
The University of North Carolina at Charlotte

Informed Consent for
Evaluating Usability of Virtual Cursor Offset Techniques for a Navigation Task in a Multi-Display Virtual Environment

Project Purpose

In this study we will investigate usability of cursor offset techniques for two-handed navigation techniques in a CAVE (Cave Automatic Virtual Environment) system that consists of three large, back-projected displays.

Investigators

Jialei Li, Computer Science
Isaac Cho, Computer Science
Zachary Wartell, Computer Science

Eligibility

You may participate in this study if you are 18 years of age or older and if you have 20/20 vision or corrected vision to 20/20 (i.e. you can clearly read text on a computer workstation monitor), and can comfortably use your arms and fingers and communicate in spoken English.

Overall Description of Participation

In the first step, you will need to complete a pre-questionnaire which is about demographics and your past experience with 3D computer applications and 3D media (e.g. 3D games and 3D movies), and your familiarity with using a computer and various user interfaces. Then you will take a stereo vision test using the Nvidia 3D Vision application. You will need to wear 3D shutter glasses during the test. If you fail the stereo vision test, unfortunately you will not be able to continue the study.

Once you pass the test, we will demonstrate how to use a CAVE system. This system consists of three large stereoscopic displays and head-tracking to display 3D computer graphics. Stereopsis is an important depth cue generated by the differences between the two views of a scene seen from a person's two eyes. Head-tracking is a tracking system which tracks the position and orientation of a person's head to generate an optimal perspective 3D image. We will demonstrate how these technologies can be used with a 3D application in the CAVE. We will train you on how these technologies work by showing you how to view the 3D application and how to perform a navigation task (rotating and translating) with two-handed navigation technique using 3D input devices, called 6DOF ("degree-of-freedom") buttonball.

In the next step, you will perform a docking task using four different variations of the user interface. A pair of the 3D cursors will appear in the virtual environment that represents your hands positions. The four variations differ in their software mechanisms used to relate the physical hands positions to the 3D cursor positions. In particular they differ on how much of a translational offset there is between your hands and the 3D cursors.

The docking task requires finding a colored target box on a gridded ground plane and placing the box so that it aligns with a second docking box. The docking box is located at the center of a randomly chosen display (one of the three). You need to move the target box to the docking box by adjusting its location and orientation. You also need to match the colors of the faces of the two boxes. When completed, you should click a button on the buttonball to finish the task. You will repeat this task 30 times using each of the four offset techniques. After each session of 30 trials, we will ask you questions using a questionnaire about your experience during that block of trials.

After you finish the entire experiment, you will take a final, post-experiment questionnaire.

Length of Participation

Participation should take approximately 50-60 minutes.

Risks and Benefits of Participation

While using virtual reality display systems, some people may experience slight symptoms of disorientation, nausea or dizziness. This is similar to the feeling experienced in wide-screen movies and theme park rides. We do not expect these to be strong or to last after participants leave the laboratory. If you feel uncomfortable and wish to stop the experiment, you are free to do so at any time without penalty by simply announcing your desire to stop.

The primary scientific benefit of the study is to compare the usability of cursor offset techniques for two-handed navigation techniques in a CAVE.

Volunteer Statement

You are a volunteer. The decision to participate in this study is completely up to you. If you decide to participate in the study, you may stop at any time. You will not be treated any differently if you decide not to participate in the study or if you stop once you have started. You can request to withdraw your data after the experiment is complete.

Confidentiality Statement

Information about your participation, including your identity, will be kept as confidential as possible. The following efforts will be taken to protect confidentiality and privacy:

1) The informed consent form will be kept in a locked filing cabinet, separate from the rest of the data.
2) All participants will be assigned a random ID consisting of two randomly-generated initials (initials will not correspond to participants' names). The participants will only be referred to by the assigned alphanumeric codes both in internal communication between researchers and in the form of written reports.
3) All questionnaries during the study will be kept in the Charlotte Visualization Center (room 412 in Woodward Hall) in a locked filing cabinet.
4) After one year, the files will be destroyed by investigator under the guidance of the responsible faculty. Paper documents will be shredded and electronic data will be deleted.

Statement of Fair Treatment and Respect

UNC Charlotte wants to make sure that you are treated in a fair and respectful manner. Contact the university's Research Compliance Office (704-687-3309) if you have questions about how you are treated as a study participant. If you have any questions about the actual project or study, please contact Jialei Li (jli42@uncc.edu), Dr. Isaac Cho (icho1@uncc.edu) or Dr. Zachary Wartell (zwartell@uncc.edu) at 704-687-8442.

Approval Date

This form was approved for use on *June XX, 2015* for one year.

Participant Consent

I have read the information in this consent form. I have had the chance to ask questions about this study, and those questions have been answered to my satisfaction. I am at least 18 years old and I agree to participate in this research project. I understand that I will receive a copy of this form after it has been signed by me and the principal investigator of this research study.

_______________________________________ _______________________

Participant Name (PRINT) DATE

____

Participant Signature

_______________________________________ _______________________

Investigator Signature DATE

Pre-questionnaire

1. Your Given ID (Instructor only):

2. Your age:

3. Your gender:

4. Occupational Status:

   Undergraduate student ___

   Master Student ___

   PhD Student ___

   Research Assistant/Fellow ___

   Staff-systems, technical ___

   Faculty ___

   Administrative Staff ___

   Other: ___________________

5. Your major:

6. Are you colorblind? Yes / No

7. Do you have 20/20 eyesight (or corrected 20/20)? Yes / No

8. Do you have any disabilities or injuries that might limit your ability to use either your left or right arm, hand and/or fingers in every tasks such as writing, painting, using a computer mouse or advanced game controller? Yes / No

9. How much arm or hand fatigue did you feel right now?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

10. Are you familiar with using a mouse and keyboard? Yes / No

11. Have you ever felt motion sick (dizziness or nausea) while playing a computer game or viewing a large, screen movie before? Yes / No

1. How often do you use a computer in your daily activities?

(Never) (A Great Deal)

1 2 3 4 5 6 7

2. How often do you play 2D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

3. How often do you play 3D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

4. How often do you play computer games (of any kind) on a computer/PC?

(Never) (A Great Deal)

1 2 3 4 5 6 7

5. How often do you play computer games using a game console, such as Nintendo®, Xbox®, Sony PlayStation® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

6. How often do play computer games using a game console with a motion capture device, such as Xbox Kinect®, Sony PlayStation Move® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

Definition: Stereoscopic 3D

Stereoscopic 3D refers to a display that creates a true 3D image that appears to pop-out in front of and behind the screen. These displays are used in some movie theaters, television sets and computer monitors. Most stereoscopic 3D display technologies known to consumers require they wear special glasses.

7. How often do you watch stereoscopic 3D movies in the theater?

(Never) (A Great Deal)

1 2 3 4 5 6 7

8. How often do you play computer games or watch movies on an in-home television using stereoscopic 3D?

(Never) (A Great Deal)

1 2 3 4 5 6 7

Definition: 3D User Interface

A "3D user interface" is a human-computer interface where the user views 3D computer graphics and interacts with those graphics by traveling through the 3D environment, and/or manipulating and changing the 3d environment. 3D user interfaces may or may not use stereoscopic 3D displays. Also 3D user interfaces may or may not use advanced 3D input devices such as the Microsoft Kinect, PlayStation Move, Nintendo Wii, etc.

9. If you have used any 3D user interfaces before, then describe what 3D user interfaces you have used and mention what type of display and input device technology you used with them.

____

____

____

____

Arm Fatigue Ratings

Your Given ID (Instructor only):

1. How much arm fatigue did you feel with the **No offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

2. How much arm fatigue did you feel with the **Fixed-length offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

3. How much arm fatigue did you feel with the **Go-Go technique offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

4. How much arm fatigue did you feel with the **Torso position offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

Post-questionnaire

Your Given ID (Instructor only):

Use the list below for reference when answering the following questions. You may write just the letter (A-D) that labels the offset technique you want to refer to in your answers.

Conditions:

A. No offset
B. Fixed-Length offset
C. Go-Go technique offset
D. Torso Position offset

1. Which condition (A to D) was the easiest to accomplish the overall **trial**? Why?

2. Which condition (A to D) was the easiest to accomplish the task when the random box is **behind the target box**? Why?

3. Which condition (A to D) was the easiest to accomplish the task when the random box is **in front of the target box**? Why?

4. Does the **random box size** make any difference in completing the task? Why?

If you have any comments for this study, please give us feedback.

APPENDIX C: EXPERIMENT DOCUMENTS FOR CHAPTER 6

UNC CHARLOTTE

Research and Economic Development

Office of Research Compliance

9201 University City Blvd, Charlotte, NC 28223-0001

t/ 704.687.1876 f/ 704.687.0980 http://research.uncc.edu/compliance-ethics

Institutional Review Board (IRB) for Research with Human Subjects

Certificate of Continuing Approval

~ for Year 2 of Study ~

Protocol #	**15-04-33**		
Protocol Type:	**Expedited 7**		
Title:	**Evaluating Usability of Virtual Cursor Offset Techniques for a Navigation Task in a Multi-Display Virtual Environment**		
Date:	**6/7/2016**		
Student Investigator	**Mr. Jialei**	**Li**	**Computer Science**
Responsible Faculty	**Dr. Zachary**	**Wartell**	**Computer Science**
Investigator	**Dr. Isaac**	**Cho**	**Computer Science**

After careful review, the protocol listed above was approved by the Institutional Review Board (IRB) for Research with Human Subjects per 45 CFR 46.111. This approval will expire one year from the date of this letter. In order to continue conducting research under this protocol after one year, the "Annual Renewal" form must be submitted to the IRB. The renewal form can be obtained from the Office of Research Compliance web page http://research.uncc.edu/compliance-ethics/human-subjects.

Please note that it is the investigator's responsibility to promptly inform the committee of any changes in the proposed research prior to implementing the changes, and of any adverse events or unanticipated risks to subjects or others. Amendment and Event Reporting forms are available on our web page at http://research.uncc.edu/compliance-ethics/human-subjects/amending-your-protocol.

M. Lyn Exum 5/19/16

Dr. M. Lyn Exum, IRB Chair Date

UNC CHARLOTTE

Research and Economic Development

Office of Research Compliance

9201 University City Blvd, Charlotte, NC 28223-0001
t/ 704.687.1876 f/ 704.687.0980 http://research.uncc.edu/compliance-ethics

Institutional Review Board (IRB) for Research with Human Subjects

University of North Carolina at Charlotte

Approval of Amendment

Protocol #	**15-04-33**		
Title:	**Evaluating Usability of Virtual Cursor Offset Techniques in Immersive Virtual Environments**		
Date:	**6/15/2016**		
Student Investigator	**Mr. Jialei**	**Li**	**Computer Science**
Responsible Faculty	**Dr. Zachary**	**Wartell**	**Computer Science**
Investigator	**Dr. Isaac**	**Cho**	**Computer Science**

The Institutional Review Board (IRB) has approved the amendment of the protocol listed above for Research with Human Subjects.

Please note that it is the investigator's responsibility to promptly inform the committee of any changes in the proposed research, as well as any unanticipated problems that may arise involving risks to subjects.

Amendment Details: Study procedure has been modified to add Head-Mounted Display (HMD) that the participants will use in performing the selection and navigation tasks using four different cursor offset techniques. The HMD system consists of a helmet with two small displays and an adjustable lens system which produces stereoscopic viewing by presenting separate images to each eye. When using the HMD, the participant either holds a pair of Razer Hydra game controllers or controls the virtual scene with bare hand gestures using Leap Motion controller attached to the front of HMD. The currently approved protocol only uses Cave Automatic Virtual Environment (CAVE) in performing the tasks. All other study procedures remain the same. Furthermore, expected number of participants is increased from 48 to 64. Lastly, project title has been changed from "Evaluating Usability of Virtual Cursor Offset Techniques in Immersive Virtual Environments" to "Evaluating Usability of Virtual Cursor Offset Techniques in Immersive Virtual Environments." Informed consent has been updated accordingly.

M. Lyn Exum 6/20/16

Dr. M. Lyn Exum, IRB Chair   Date

UNCCHARLOTTE
The University of North Carolina at Charlotte

Informed Consent for
Evaluating Usability of Virtual Cursor Offset Techniques in Immersive Virtual Environments

Project Purpose

In this study we will investigate usability of cursor offset techniques for selection tasks in an HMD (Head-Mounted Display) system.

Investigators

Jialei Li, Computer Science
Isaac Cho, Computer Science
Zachary Wartell, Computer Science

Eligibility

You may participate in this study if you are 18 years of age or older and if you have 20/20 vision or corrected vision to 20/20 (i.e. you can clearly read text on a computer workstation monitor), and can comfortably use your arms and fingers and communicate in spoken English.

Overall Description of Participation

In the first step, you will need to complete a pre-questionnaire which is about demographics and your past experience with 3D computer applications and 3D media (e.g. 3D games and 3D movies), and your familiarity with using a computer and various user interfaces. Then you will have to take a stereo vision test using Multi-Target Anaglyph Stereo Test package. You will need to wear reversible red/green anaglyph glasses during the test. If you fail the stereo vision test, unfortunately you will not be able to continue the study.

Once you pass the test, we will demonstrate how to use the immersive virtual reality system. The HMD system consists of a helmet with two small displays and an adjustable lens system which produces stereoscopic viewing by presenting separate images to each eye. Stereopsis is an important depth cue generated by the differences between the two views of a scene seen from a person's two eyes. Head-tracking is a tracking system which tracks the position and orientation of a person's head to generate an optimal perspective 3D image. We will show how these technologies can be combined within a 3D application in the immersive virtual environment. We will train you on how to view the virtual scene and how to interact with the virtual environment using 3D input devices.

In the next step, you will perform a selection task using four different variations of the offset technique. A pair of the 3D virtual cursors will appear in the virtual environment that represents your hands positions. The four variations differ in their software mechanisms used to relate the 3D cursor positions to your physical hands positions. In particular, they differ on how much of a translational offset there is between your hands and the 3D cursors.

The selection task requires you to repeatedly select a virtual sphere that is highlighted in blue within a circle of spheres. The circle of spheres will appear at a pre-defined location in the virtual scene. You need to successfully select the highlighted sphere before you can proceed to the next target. You will repeat this circle of selection tasks multiple times using either the Razer Hydra game controller or the Leap Motion gesture controller. There will be a total of four sessions under each controller with each session using one of the four offset techniques. In between sessions, we will ask your opinions about your experience during that block of trials.

After you finish the entire experiment, you will take a final, post-experiment questionnaire.

Length of Participation

Participation should take approximately 50-60 minutes.

Risks and Benefits of Participation

While using virtual reality display systems, some people may experience slight symptoms of disorientation, nausea or dizziness. This is similar to the feeling experienced in wide-screen movies and theme park rides. We do not expect these to be strong or to last after participants leave the laboratory. If you feel uncomfortable and wish to stop the experiment, you are free to do so at any time without penalty by simply announcing your desire to stop.

The primary scientific benefit of the study is to explore the usability of virtual cursor offset techniques for interaction in an immersive virtual environment.

Volunteer Statement

You are a volunteer. The decision to participate in this study is completely up to you. If you decide to participate in the study, you may stop at any time. You will not be treated any differently if you decide not to participate in the study or if you stop once you have started. You can request to withdraw your data after the experiment is complete.

Confidentiality Statement

Information about your participation, including your identity, will be kept as confidential as possible. The following efforts will be taken to protect confidentiality and privacy:

1) The informed consent form will be kept in a locked filing cabinet, separate from the rest of the data.
2) All participants will be assigned a random ID consisting of two randomly-generated initials (initials will not correspond to participants' names). The participants will only be referred to by the assigned alphanumeric codes both in internal communication between researchers and in the form of written reports.
3) All questionnaries during the study will be kept in the Charlotte Visualization Center (room 412 in Woodward Hall) in a locked filing cabinet.
4) After one year, the files will be destroyed by investigator under the guidance of the responsible faculty. Paper documents will be shredded and electronic data will be deleted.

Statement of Fair Treatment and Respect

UNC Charlotte wants to make sure that you are treated in a fair and respectful manner. Contact the university's Research Compliance Office (704-687-3309) if you have questions about how you are treated as a study participant. If you have any questions about the actual project or study, please contact Jialei Li (jli42@uncc.edu), Dr. Isaac Cho (icho1@uncc.edu) or Dr. Zachary Wartell (zwartell@uncc.edu) at 704-687-8442.

Approval Date

This form was approved for use on *June 15th, 2016* for one year.

Participant Consent

I have read the information in this consent form. I have had the chance to ask questions about this study, and those questions have been answered to my satisfaction. I am at least 18 years old and I agree to participate in this research project. I understand that I will receive a copy of this form after it has been signed by me and the principal investigator of this research study.

________________________________ ____________________
Participant Name (PRINT) DATE

____
Participant Signature

________________________________ ____________________
Investigator Signature DATE

Pre-questionnaire

1. Your Given ID (Instructor only):

2. Your age:

3. Your gender:

4. Occupational Status:

   Undergraduate student ___

   Master Student ___

   PhD Student ___

   Research Assistant/Fellow ___

   Faculty/Staff ___

   Other: ___________________

5. Your major:

6. Are you colorblind? Yes / No

7. Do you have 20/20 eyesight (or corrected 20/20)? Yes / No

8. Do you have any disabilities or injuries that might limit your ability to use either your left or right arm, hand and/or fingers in every tasks such as writing, painting, using a computer mouse or advanced game controller? Yes / No

9. How much arm or hand fatigue did you feel right now?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

10. Are you familiar with using a mouse and keyboard? Yes / No

11. Are you familiar with VR devices? Yes / No

12. Have you ever felt motion sick (dizziness or nausea) while playing a computer game or viewing a large, screen movie before? Yes / No

1. How often do you use a computer in your daily activities?

(Never) (A Great Deal)

1 2 3 4 5 6 7

2. How often do you play 2D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

3. How often do you play 3D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

4. How often do you play computer games (of any kind) on a computer/PC?

(Never) (A Great Deal)

1 2 3 4 5 6 7

5. How often do you play computer games using a game console, such as Nintendo®, Xbox®, Sony PlayStation® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

6. How often do play computer games using a game console with a motion capture device, such as Xbox Kinect®, Sony PlayStation Move® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

Stereoscopic 3D refers to a display that creates a true 3D image that appears to pop-out in front of and behind the screen. These displays are used in movie theaters, television sets and computer monitors. Most stereoscopic 3D display technologies known to consumers require they wear special glasses.

7. How often do you watch stereoscopic 3D movies in the theater?

(Never) (A Great Deal)

1 2 3 4 5 6 7

8. How often do you play computer games or watch stereoscopic 3D movies on an in-home television?

(Never) (A Great Deal)

1 2 3 4 5 6 7

3D User Interface is a human-computer interface where the user views 3D computer graphics and interacts with those graphics by traveling through the 3D environment, and/or manipulating virtual objects in the environment. 3D user interfaces may or may not use stereoscopic 3D displays like CAVE, HMD (Samsung Gear VR, Oculus Rift, and HTC Vive). Also 3D user interfaces may or may not use advanced 3D input devices such as the Microsoft Kinect, PlayStation Move, Nintendo Wii, Leap Motion, etc.

9. If you have used any 3D user interfaces before, then describe what 3D user interfaces you have used and mention what type of display and input device technology you used with them.

____

____

____

____

Arm Fatigue Ratings

Your Given ID (Instructor only):

Device Type: **Razer Hydra**

1. How much arm fatigue did you feel with the **No Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

2. How much arm fatigue did you feel with the **Fixed-length Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

3. How much arm fatigue did you feel with the **Go-Go Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

4. How much arm fatigue did you feel with the **Linear Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

Device Type: **Leap Motion**

5. How much arm fatigue did you feel with the **No Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

6. How much arm fatigue did you feel with the **Fixed-length Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

7. How much arm fatigue did you feel with the **Go-Go Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

8. How much arm fatigue did you feel with the **Linear Offset condition**?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

Post-questionnaire

Your Given ID (Instructor only):

Use the list below for reference when answering the following questions. You may write just the letter (A-D) that labels the offset technique you want to refer to in your answers.

Conditions:

A. No Offset
B. Fixed-Length Offset
C. Go-Go Offset
D. Linear Offset

1. How easy was it to select the sphere in No Offset technique (A) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

2. How easy was it to select the sphere in Fixed-Length Offset technique (B) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

3. How easy was it to select the sphere in Go-Go Offset technique (C) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

4. How easy was it to select the sphere in Linear Offset technique (D) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

5. Which condition (A/C/D) is the easiest to complete the task when the circle of spheres is close by and within reach? Why?

____

____

____

6. Which condition (B/C/D) is the easiest to complete the task when the circle of spheres is far away and out of reach? Why?

____

____

____

7. Which condition (A-D) do you prefer to complete the selection task overall? Why?

____

____

____

8. Does the distance to the circle make any difference in completing the tasks? Why?

____

____

____

9. How easy was it to select spheres using Razer Hydra?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

10. How easy was it to select spheres using Leap Motion?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

11. Which device do you prefer for the selection task overall? Why?

____

____

____

12. If you have any comments on this study, please give us feedback.

____

____

____

APPENDIX D: EXPERIMENT DOCUMENTS FOR CHAPTER 7

Informed Consent for
Evaluating Usability of Virtual Cursor Offset Techniques in Immersive Virtual Environments

Project Purpose

In this study we will investigate usability of cursor offset techniques on 7DOF docking tasks in an HMD (Head-Mounted Display) system.

Investigators

Jialei Li, Computer Science
Isaac Cho, Computer Science
Zachary Wartell, Computer Science

Eligibility

You may participate in this study if you are 18 years of age or older and if you have 20/20 vision or corrected vision to 20/20 (i.e. you can clearly read text on a computer workstation monitor), and can comfortably use your arms and fingers and communicate in spoken English.

Overall Description of Participation

In the first step, you will need to complete a pre-questionnaire which is about demographics and your past experience with 3D computer applications and 3D media (e.g. 3D games and 3D movies), and your familiarity with using a computer and various user interfaces. Then you will have to take a stereo vision test using Multi-Target Anaglyph Stereo Test package. You will need to wear reversible red/green anaglyph glasses during the test. If you fail the stereo vision test, unfortunately you will not be able to continue the study.

Once you pass the test, we will demonstrate how to use the immersive virtual reality system. The HMD system consists of a helmet with two small displays and an adjustable lens system which produces stereoscopic viewing by presenting separate images to each eye. Stereopsis is an important depth cue generated by the differences between the two views of a scene seen from a person's two eyes. Head-tracking is a tracking system which tracks the position and orientation of a person's head to generate an optimal perspective 3D image. We will show how these technologies can be combined within a 3D application in the immersive virtual environment. We will train you on how to view the virtual scene and how to interact with the virtual environment using 3D input devices.

In the next step, you will perform a docking task using four different variations of the offset technique. A pair of the 3D virtual cursors will appear in the virtual environment that represents your hands positions. The four variations differ in their software mechanisms used to relate the 3D cursor positions to your physical hands positions. In particular, they differ on how much of a translational offset there is between your hands and the 3D cursors.

The docking task requires finding a colored target tetrahedron above the ground plane and placing the docking tetrahedron so that it aligns with the target tetrahedron. The docking tetrahedron is located at the center of the ground plane with a random size and random orientation. You need to move the docking tetrahedron to the target tetrahedron by adjusting its position and orientation. You also need to match the colors of the faces of the two objects. When completed, you should click a button to finish the task. You will need to repeat this task multiple times using either the Razer Hydra game controller or the Leap Motion gesture controller. There will be a total of four sessions under each controller with each session using one of the four offset techniques. In between sessions, we will ask your opinions about your experience during that block of trials.

After you finish the entire experiment, you will take a final, post-experiment questionnaire.

Length of Participation

Participation should take approximately 50-60 minutes.

Risks and Benefits of Participation

While using virtual reality display systems, some people may experience slight symptoms of disorientation, nausea or dizziness. This is similar to the feeling experienced in wide-screen movies and theme park rides. We do not expect these to be strong or to last after participants leave the laboratory. If you feel uncomfortable and wish to stop the experiment, you are free to do so at any time without penalty by simply announcing your desire to stop.

The primary scientific benefit of the study is to explore the usability of virtual cursor offset techniques for interaction in an immersive virtual environment.

Volunteer Statement

You are a volunteer. The decision to participate in this study is completely up to you. If you decide to participate in the study, you may stop at any time. You will not be treated any differently if you decide not to participate in the study or if you stop once you have started. You can request to withdraw your data after the experiment is complete.

Confidentiality Statement

Information about your participation, including your identity, will be kept as confidential as possible. The following efforts will be taken to protect confidentiality and privacy:

1) The informed consent form will be kept in a locked filing cabinet, separate from the rest of the data.
2) All participants will be assigned a random ID consisting of two randomly-generated initials (initials will not correspond to participants' names). The participants will only be referred to by the assigned alphanumeric codes both in internal communication between researchers and in the form of written reports.
3) All questionnaries during the study will be kept in the Charlotte Visualization Center (room 412 in Woodward Hall) in a locked filing cabinet.
4) After one year, the files will be destroyed by investigator under the guidance of the responsible faculty. Paper documents will be shredded and electronic data will be deleted.

Statement of Fair Treatment and Respect

UNC Charlotte wants to make sure that you are treated in a fair and respectful manner. Contact the university's Research Compliance Office (704-687-3309) if you have questions about how you are treated as a study participant. If you have any questions about the actual project or study, please contact Jialei Li (jli42@uncc.edu), Dr. Isaac Cho (icho1@uncc.edu) or Dr. Zachary Wartell (zwartell@uncc.edu) at 704-687-8442.

Approval Date

This form was approved for use on *June 15th, 2016* for one year.

Participant Consent

I have read the information in this consent form. I have had the chance to ask questions about this study, and those questions have been answered to my satisfaction. I am at least 18 years old and I agree to participate in this research project. I understand that I will receive a copy of this form after it has been signed by me and the principal investigator of this research study.

______________________________ ______________________
Participant Name (PRINT) DATE

____
Participant Signature

______________________________ ______________________
Investigator Signature DATE

Pre-questionnaire

1. Your Given ID (Instructor only):

2. Your age:

3. Your gender:

4. Occupational Status:

   Undergraduate student ___

   Master Student ___

   PhD Student ___

   Research Assistant/Fellow ___

   Faculty/Staff ___

   Other: ___________________

5. Your major:

6. Are you colorblind? Yes / No

7. Do you have 20/20 eyesight (or corrected 20/20)? Yes / No

8. Do you have any disabilities or injuries that might limit your ability to use either your left or right arm, hand and/or fingers in every tasks such as writing, painting, using a computer mouse or advanced game controller? Yes / No

9. How much arm or hand fatigue did you feel right now?

(Not At All) (Very Painful)

1 2 3 4 5 6 7

10. Are you familiar with using a mouse and keyboard? Yes / No

11. Are you familiar with VR devices? Yes / No

12. Have you ever felt motion sick (dizziness or nausea) while playing a computer game or viewing a large, screen movie before? Yes / No

1. How often do you use a computer in your daily activities?

(Never) (A Great Deal)

1 2 3 4 5 6 7

2. How often do you play 2D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

3. How often do you play 3D computer games?

(Never) (A Great Deal)

1 2 3 4 5 6 7

4. How often do you play computer games (of any kind) on a computer/PC?

(Never) (A Great Deal)

1 2 3 4 5 6 7

5. How often do you play computer games using a game console, such as Nintendo®, Xbox®, Sony PlayStation® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

6. How often do play computer games using a game console with a motion capture device, such as Xbox Kinect®, Sony PlayStation Move® or other?

(Never) (A Great Deal)

1 2 3 4 5 6 7

Stereoscopic 3D refers to a display that creates a true 3D image that appears to pop-out in front of and behind the screen. These displays are used in movie theaters, television sets and computer monitors. Most stereoscopic 3D display technologies known to consumers require they wear special glasses.

7. How often do you watch stereoscopic 3D movies in the theater?

(Never)						(A Great Deal)
1	2	3	4	5	6	7

8. How often do you play computer games or watch stereoscopic 3D movies on an in-home television?

(Never)						(A Great Deal)
1	2	3	4	5	6	7

3D User Interface is a human-computer interface where the user views 3D computer graphics and interacts with those graphics by traveling through the 3D environment, and/or manipulating virtual objects in the environment. 3D user interfaces may or may not use stereoscopic 3D displays like CAVE, HMD (Samsung Gear VR, Oculus Rift, and HTC Vive). Also 3D user interfaces may or may not use advanced 3D input devices such as the Microsoft Kinect, PlayStation Move, Nintendo Wii, Leap Motion, etc.

9. If you have used any 3D user interfaces before, then describe what 3D user interfaces you have used and mention what type of display and input device technology you used with them.

____

____

____

____

Arm Fatigue Ratings

Your Given ID (Instructor only):

Device Type: **Razer Hydra**

1. How much arm fatigue did you feel with the **No Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

2. How much arm fatigue did you feel with the **Fixed-length Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

3. How much arm fatigue did you feel with the **Go-Go Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

4. How much arm fatigue did you feel with the **Linear Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

Device Type: **Leap Motion**

5. How much arm fatigue did you feel with the **No Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

6. How much arm fatigue did you feel with the **Fixed-length Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

7. How much arm fatigue did you feel with the **Go-Go Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

8. How much arm fatigue did you feel with the **Linear Offset condition**?
(Not At All) (Very Painful)

1 2 3 4 5 6 7

Post-questionnaire

Your Given ID (Instructor only):

Use the list below for reference when answering the following questions. You may write just the letter (A-D) that labels the offset technique you want to refer to in your answers.

Conditions:

A. No Offset
B. Fixed-Length Offset
C. Go-Go Offset
D. Linear Offset

1. How easy was it to adjust the tetrahedron in No Offset technique (A) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

2. How easy was it to adjust the tetrahedron in Fixed-Length Offset (B) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

3. How easy was it to adjust the tetrahedron in Go-Go Offset technique (C) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

4. How easy was it to adjust the tetrahedron in Linear Offset technique (D) condition?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

5. Which condition (A/C/D) was the easiest to complete the task when the tetrahedron was close by and within reach? Why?

____

____

____

6. Which condition (B/C/D) was the easiest to complete the task when the tetrahedron was far away and out of reach? Why?

____

____

____

7. Which condition (A-D) do you prefer to complete the docking task overall? Why?

____

____

____

8. Does the initial size of the tetrahedron make any difference in completing the tasks? Why?

____

____

____

9. How easy was it to align tetrahedra using Razer Hydra?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

10. How easy was it to align tetrahedra using Leap Motion?

(Very Easy) (Very Difficult)

1 2 3 4 5 6 7

11. Which device do you prefer for the docking task overall? Why?

____

____

____

12. If you have any comments on this study, please give us feedback.

____

____

____