COMPARING 2D AND 3D DIRECT MANIPULATION INTERFACES

ROBERT J. TEATHER

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

GRADUATE PROGRAMME IN COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

FEBRUARY, 2008

COMPARING 2D AND 3D INPUT DEVICES FOR 3D OBJECT MOVEMENT

by **Robert J. Teather**

A thesis submitted to the Faculty of Graduate Studies of York University in partial fulfilment of the requirements for the degree of

MASTER OF SCIENCE

COMPARING 2D AND 3D INPUT DEVICES FOR 3D OBJECT MOVEMENT

by **Robert J. Teather**

Examination Committee Members:

1. Doctor Wolfgang Stuerzlinger
2. Doctor Robert Allison
3. Doctor Parke Godfrey
4. Doctor Caitlin Fisher

ABSTRACT

This thesis extends an intuitive mouse-based 3D object movement technique to usage with 3D input devices. Three user studies that assess the value of virtual reality input devices and display technologies when using this technique are presented. In particular, the effects of several properties of the mouse including support and display orientation/input device movement congruency are examined, as are stereo rendering and head-coupled perspective. The results suggest that, overall, the mouse tends to outperform 3D devices – regardless of display conditions. However, when using the same constrained 3D movement technique as the mouse, these 3D devices fared better than when using unconstrained 3D direct manipulation.

ACKNOWLEDGEMENTS

Thanks go to my supervisor, Wolfgang Stuerzlinger, for providing his expertise and guidance, for his flexibility and understanding, and for keeping me on-track. Thanks also for providing funding for the user studies, which made this research possible.

Thanks to Dr. Rob Allison for use of his lab equipment for the first user study, and for agreeing to serve on my examination committee. Thanks also to Dr. Parke Godfrey and Dr. Caitlin Fisher for taking time out of their schedules to serve on my committee.

I'd also like to thank Dr. John Bonnett of Brock University, for use of his lab and equipment for the last two studies. Thanks also to Dr. Mike Duncan of Niagara College for first piquing my interest in research, directing me toward graduate studies, and for employing me through the first year and a half of my studies.

Thanks to my friends Andriy Pavlovych and Dusty Phillips for helpful insights into the research, assistance with video production, and for reviewing the draft of this thesis.

Finally, thanks to my girlfriend Vicky McArthur for her understanding, for putting up with me through it all, and for keeping me sane (more or less). Thanks also to my parents for their endless encouragement and support.

Without all of these people, this work would not have been possible.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1
Introduction

Ever since Sutherland first proposed the “ultimate” display in 1965 [Sutherland, 1965], computer scientists have been attempting to reproduce reality with computer technology. In the early to mid 1990’s, major strides were made toward this end, as the field of virtual reality (VR) began to mature. The attraction of this kind of computer interface is the supposed naturalness with which one could interact with the computer. However, despite the hype of the ‘80’s and ‘90’s, virtual reality remains largely inaccessible to novice users, and is not widely used outside of labs to this day.

One of the major reasons for this is the relative inefficiency of such interfaces when compared to the popular desktop computer metaphor. Though there are a variety of reasons for this, perhaps the greatest challenge is simply allowing users of such environments to manipulate virtual objects as easily as they can manipulate real objects. Effectively, to achieve this goal would require (near) complete transparency of all equipment and the user interface itself. Of course, such a computer system does not yet exist. However, even attaining object manipulation speed and precision on par with mouse-driven desktop interfaces would be a great stride forward.

There are many different 3D object manipulation techniques that use a variety of input devices to move objects in virtual environments. Many of these use higher-dimensional input devices (e.g., 3 to 6 degrees of freedom). This is problematic, as novice computer users tend to have difficulty even when first learning to use a mouse. Higher-dimensional input devices exacerbate this problem, even for expert computer users who are typically unfamiliar with such devices and the techniques developed for use with them. None of these techniques allow users to interact with virtual objects with the comparative ease with which they can perform conceptually similar tasks in 2D graphical user interfaces. For example, users frequently move desktop icons with a mouse via the familiar “drag ‘n’ drop” direct manipulation metaphor, but directly moving an object in 3D is far more difficult due to the additional axis of precise movement required.

The standard desktop mouse has been shown to be a good alternative to 3D input devices for constrained 3D object movement tasks, but requires the use of special software techniques to map its 2D input into 3D operations. So why not simply use a mouse for virtual object manipulation? While a mouse is certainly suitable for certain types of environments, there are situations that necessitate the usage of 3D input devices. Thus, the designer’s goal becomes to choose a manipulation technique that maximizes the efficiency of the user. The focus is taken off hardware and placed on software instead.

1.1 Thesis Objectives

This work is intended to address one aspect of why it is that, despite the hype and promise of VR, it still remains largely unused by the general public. This is addressed by comparing typical VR input techniques and display modalities to the most prevalent form of computer interface, the desktop metaphor, which despite the significant learning curve, is still commonly used by novices with little or no formal computer training. The hope is that novice users may be able to leverage their familiarity with the desktop interface when using VR systems that use similar manipulation techniques, thus improving the immediate usability of VR and 3D graphics systems.

One goal of this thesis is to determine which factors make interacting with three-dimensional virtual environments difficult for users, and to what extent these can be mitigated through well-designed software manipulation techniques. In particular, constraining 3D input to 2D movement operations is discussed. Part of this goal is to determine how well a 3D movement technique designed for use with the mouse can be adapted for use with a 3D input device. The question raised is “does constraining a 3D input device to 2D operation provide a more familiar interface for novice users?” Intuitively, reducing the degrees of freedom of the control should make the task simpler. Additionally, certain properties of the mouse, which appear to be advantages, are examined. This helps determine how important they are, and if their absence negatively affects user performance with 3D devices when using said 2D-constrained 3D movement technique. In other words, this goal is to determine if making 3D input devices behave more like a mouse can help novice 3D graphics users interact with virtual objects.

A second goal is to assess the value of two virtual reality display techniques when moving 3D objects with 2D and 3D input devices. The specific technologies discussed include stereoscopic graphics and head-coupled perspective, both of which are commonly used in VR systems to increase immersion. Despite previous work (reviewed in Sections 2.1.1 and 2.2) suggesting that they can enhance user performance when using 3D input devices, little to no work has been done to see if this applies to 2D input devices such as the mouse. Thus, a goal of this research is to determine whether user performance with 2D devices can also benefit from the use of stereoscopic graphics and head-coupled perspective.

1.2 Thesis Contributions

The primary contribution of this thesis is a series of experiments comparing various factors relating to interactive 3D object positioning. Although this topic has been previously addressed (see Chapter 2 for an overview), a hypothesis of this work is that 2D input devices, in particular the mouse, will tend to allow users to perform similar tasks more efficiently than using 3D input devices for object positioning. This is achieved through the use of a previously developed object movement technique used in the SESAME 3D conceptual design system, designed for use with a desktop computer [Oh, 2005; Oh & Stuerzlinger 2004; 2005]. Another hypothesis of this thesis is that 3D movement techniques based on 2D direct manipulation can be successfully adapted for usage with 3D input devices. Furthermore, these types of techniques will tend to outperform those based on 3D direct manipulation; that is, 3 degree of freedom object movement.

This thesis extends previous work [Oh & Stuerzlinger, 2005; Oh, 2005] into the realm of semi-immersive VR. In particular, fish tank VR systems are examined, due to their similarity to desktop computers. The first experiment directly compares 3D positioning with the mouse to that with a 3D tracking device. Evaluations of stereoscopic graphics rendering and coupling the scene viewpoint to the user's head pose are provided as well. Another aspect of this experiment is an evaluation of the use of stereoscopic graphics specifically when using the mouse and a 2D constrained movement technique.

In trying to break down the advantages of the mouse, an experiment involving physical support and device movement/display orientation congruency is also presented. Unlike a mouse, three-dimensional input devices are not constrained to use on a flat surface. Their movement planes can either be orthogonal to the display (as with a mouse) or can operate parallel to the display, with or without a supporting surface. This experiment addresses which of these properties of the mouse aid its performance, and which hinder it, and if these features can be used with 3D devices.

1.3 Thesis Outline

This thesis begins with an overview of related work on 3D manipulation. In particular, 3D manipulation with 3D input devices and the mouse are discussed. Hybrid interfaces merging aspects of both, and using physical support systems (e.g., tablets) are also covered. Additionally, the general issues of selecting and moving objects in 3D graphical systems are discussed.

Following the related work, Chapter 3 discusses some assumptions made in this thesis about the 3D manipulation task. The 3D movement technique used in the studies is then presented. In addition the extensions made to the SESAME system used in the experiments are described.

Chapter 4 presents the first preliminary study, which was designed to compare the mouse to a 3D wand input device. The study also compares the effects of head-coupled perspective and stereoscopic graphics for additional depth cues. This chapter lays the groundwork for the later experiments.

Chapter 5 presents two formal user studies designed to address the results of the study from Chapter 4. The first study of Chapter 5 attempts to determine which properties of the mouse make it well suited to 3D object movement (when using the SESAME movement technique). The second Chapter 5 study extends the results of the previous two studies, and is designed to determine if other factors were missed in the previous studies.

Finally, Chapter 6 concludes the thesis with a brief summary of the experimental results, and an overview of topics for future work.

Chapter 2
Previous & Related Work

This thesis addresses several related areas of three-dimensional user interface design, including 3D manipulation, comparing input devices and VR display technologies. Consequently, previous research from each of these areas is discussed below.

2.1 3D Manipulation and Selection

Manipulating 3D objects requires the handling of 6 degrees of freedom (DOFs). There are 3 independent axes of movement and 3 axes of rotation for every object. Prior to manipulating an object, it must be selected. Both selection and manipulation can be performed with 2D or 3D input devices. The differences between these are addressed below in Sections 2.1.1 and 2.1.2.

2.1.1 3D Selection and Manipulation with 6DOF Input Devices

A large body of VR research focuses on using 3D input devices such as 6DOF trackers and wands for 3D manipulation tasks. The motivation for this is that these types of devices allow the user to simultaneously position and orient a virtual object, thus theoretically providing a more efficient manipulation interface than input devices that control fewer DOFs. Many of these devices can operate without the need for a supporting surface, and thus are well-suited to environments where the user is standing or walking. Examples of such environments are those running in CAVEs [Cruz-Neira et al., 1992] or using head-mounted displays. In these types of environments, commercially available 3D tracking systems such as those provided by Intersense [Intersense, 2007] or Polhemus [Polhemus, 2007] are commonly used. These types of tracking systems often use combinations of acoustic, inertial, optical and electromagnetic tracking technology to determine the position of the device in 3D space. Gyroscopic sensors are typically used to determine its orientation. The tracked devices themselves tend to be small and fairly mobile, and are thus attractive solutions for mobile VR.

Two previous works presented taxonomies of 3D selection/manipulation techniques [Bowman et al., 1999; Poupyrev et al., 1998a]. Bowman et al. presented a fine-grained classification of techniques [Bowman et al., 1999], breaking down each technique by its method of selection, manipulation, and release, and further sub-dividing these groupings. This approach allows the enumeration of many manipulation techniques from basic “building blocks” provided.

Poupyrev et al. compared selection and manipulation with 3D ray-casting and a virtual hand metaphor [Poupyrev et al., 1998a]. Ray casting is a commonly used selection technique, which takes the position and orientation of a 6DOF device to generate an infinite 3D ray extending into a scene from the tip of a virtual hand. This ray is then checked for intersections with objects in the scene and the closest object is selected. Virtual hand metaphors use a 3D tracker mounted on the user’s real hand (or alternatively a 6DOF wand-style device) to control the movement and orientation of a virtual hand avatar in the scene. Selection is achieved by intersecting this virtual hand with the desired object, then pressing a button on the tracker. The findings of this study suggest that there was no significant difference between the virtual hand and ray-casting for selection. Each technique tested had advantages and disadvantages, depending on factors such as distance to the target, object size and visual feedback.

Several other studies have also looked at the use of 3/6DOF input devices for selection and targeting. Zhai et al. conducted a study of the *silk cursor*, a selection technique using transparency and volumetric selection for 6DOF selection tasks [Zhai et al., 1994]. They compared their semi-transparent volumetric cursor to a wire-frame volumetric cursor, as well as stereo to mono graphics. They found that in addition to significant differences by cursor type, the stereoscopic display significantly improved user speed and accuracy. Their results suggest that both (partial) occlusion and stereopsis are beneficial in depth perception, but using both simultaneously provides an even stronger depth cue.

Boritz and Booth conducted a series of studies on 6DOF input devices for 3D interaction tasks [Boritz & Booth, 1997; 1998]. They first studied the use of 6DOF input devices for selection tasks [Boritz & Booth, 1997]. In their study, they compared stereoscopic to monoscopic display with and without head tracking, as well as different target positions. After an initial homing task, where the user had to position the 3D cursor inside a target area, the main task involved moving the cursor to one of 6 possible target locations 10cm away from the home position along any of the positive or negative X, Y and Z axes. They found that target position had a significant effect on task completion time and accuracy. Movement along the Z axis ("near" and "far" as it was called in the study) took longer and was less accurate than movement in the X and Y directions. However, interaction effects with stereoscopic display mode showed that these differences were significantly lessened when users were provided with the additional depth cue. Their second study also considered orientation of the target [Boritz & Booth, 1998], requiring users to dock a cursor with a target, matching both position and orientation. Again, it was found that differences existed depending on the position moved to, but this was further complicated by interactions with the target orientation.

It is interesting to note that, with the exception of the docking task in Boritz et al.'s second study [Boritz & Booth, 1998], all studies mentioned above used only 3DOF of the six afforded by the 6DOF input devices used, during manipulation. In all but the docking study, the 6DOF input device was only used for positioning, not orientation.

Another study conducted by Zhai and colleagues [Zhai et al., 1996] compared the use of specific muscle groups for manipulation. This is an important consideration when directly comparing two different input devices. It has been suggested that the use of more dextrous muscles, namely the fingers, can aid in 6DOF manipulation tasks. Consequently, it is not uncommon to see glove-based interfaces used in virtual environments (for commercially available examples, see the Pinch Glove [FakeSpace, 2007], or CyberGlove [Immersion, 2007]). Zhai's study compared two input devices for a 6DOF docking task; one based on a 3D tracker mounted on the palm of a glove, and the other based on a 3D tracker inside a ball the user holds with their fingers. A thorough series of analyses showed that the “FingerBall” clearly outperformed the glove. The results suggest that the use of fine-motor control muscle groups, such as those in the fingers is beneficial in 6DOF manipulation tasks. The authors suggest this may be because the FingerBall permitted various parts of the arm to work together in unison, rather than in isolation [Zhai et al., 1996]. This conclusion was supported by later work comparing muscle groups in the fingers, wrist and forearm [Balakrishnan & MacKenzie, 1997]. The results of this work suggested that using these muscle groups *together* resulted in superior performance than just using the fingers alone for Fitt’s rapid aimed pointing tasks. In particular, they found that holding a stylus between the thumb and forefinger permitted better task performance than the fingers, wrist and forearm movements tested in their experiment. The authors conclude that certain muscle groups are likely better suited to certain types of movement tasks, and consequently, input devices that use specific muscle groups should be matched to the task at hand.

The above studies were based on 3DOF selection techniques. Other works suggest that 2DOF selection can, in fact, outperform 3DOF selection [Bowman et al., 1999; Ware & Lowther, 1997]. Bowman et al. presented a study that compared several techniques created from basic 3D interaction components, and evaluated them in a selection and manipulation test-bed [Bowman et al., 1999]. They found that selection based on ray-casting and occlusion was significantly faster than selection techniques requiring 3D hand or cursor movement. For manipulation, they found that the degrees of freedom of the manipulation task had a significant effect on task completion time. In fact, they note that it dominated the results, with techniques based on 2DOF motions significantly outperforming 6DOF techniques, on average. This supports the findings presented earlier by Ware and Lowther [Ware & Lowther, 1997].

Finally, various 3D interaction techniques rely on exaggeration or amplification of user motions, in some cases for both selection and manipulation. Two examples of these are Go-Go [Poupyrev et al., 1996] and HOMER [Bowman & Hodges, 1997]. The Go-Go technique is essentially a virtual hand that allows the user to interactively and non-linearly adjust the length of their virtual arm when manipulating an object in 3D. HOMER, on the other hand, uses 3D ray-casting selection, and automatically moves the user's virtual hand to the position of a selected object. Both of these effectively extend the motion of the user's hand, thus allowing them to manipulate remote objects without having to physically move closer to them. In general, manipulation techniques based on 2D ray casting behave similarly to this, as one can select an object at a distance if the ray intersects it. The advantage of these and similar techniques over "traditional" virtual hands is that the user is no longer limited by the physical length of their arms when reaching for virtual objects.

2.1.2 3D Selection and Manipulation with the Mouse

Most users are extensively familiar with 2D input devices, in particular the mouse. Furthermore, practically all commercially successful 3D graphics systems (including 3D modeling packages and computer games) use a mouse-based user interface. However, the use of a mouse for 3D interaction introduces the problem of mapping 2D mouse motions into 3D operations. While several solutions have been proposed, all of them require that users mentally translate 2D mouse movements into low-level 3D operations, which is difficult for users. However, there is evidence that 2D input devices can outperform 3D devices for certain 3D positioning tasks when using software techniques that *automatically* map mouse movement to 3D object movement in an intuitive way. This is typically achieved through the use of ray-casting.

Although ray-casting can be used with 6DOF devices, the technique also enables 3D selection with 2D input devices, such as the mouse. For this it suffices to use the 2D XY coordinates (i.e., the mouse cursor position) and generate a ray from the viewpoint through that 2D point on the display, and into the scene. Several previous works point out that 2D interface devices work well for 3D interaction when ray casting is used for selection and manipulation [Ware & Lowther, 1997; Bowman et al., 1999; Smith et al., 2001; Oh & Stuerzlinger, 2005]. Ware and Lowther conjecture that this is because situations where the user wishes to interact with totally occluded objects are rare. It is interesting to point out in this context that a 2D image of a 3D scene is fully representative of all visible objects in that scene. Since ray-casting allows the user to pick any (even only partially) visible object, it is sufficient to allow selection of objects [Ware & Lowther, 1997]. Ware and Lowther's study found that a ray-casting based 2D selection technique using a cursor rendered to a single eye in a stereo display was more accurate than a 3D selection cursor.

Manipulation is less straightforward than selection, since it is potentially a 6DOF task, and the mouse only affords the simultaneous manipulation of 2DOF. Thus, 2D input must be mapped to 3D operations via software techniques. Examples of this strategy are 3D widgets, such as “3D handles” [Connor et al., 1992], the “skitters and jacks” technique [Bier, 1987], or the use of mode control keys. The use of 3D widgets is the solution that is often used in most modeling and commercial CAD software [Connor et al., 1992; Straus et al., 2002]. These handles separate the different DOFs by explicitly breaking the manipulation down into its individual components. Small arrows/handles are provided for movement along each of the three axes, and orientation circles/spheres for each axis of rotation. This is usually complemented by different simultaneous orthogonal views of the same scene from different sides. Bier’s skitters and jacks technique [Bier, 1987] provides a similar solution, by interactively sliding the 3D cursor over objects in the scene via ray-casting, and attaching a transformation coordinate system to the object where it was positioned. The use of mode keys allows the user to change the 2DOFs the mouse is currently controlling by holding a specific key. For example, movement may default to the XZ plane, but holding the “shift” key during the movement may change the plane of movement to the XY plane instead. The limitation of these types of manipulation techniques is that users need to mentally decompose every movement into a series of 2DOF operations mapping to individual operations along the three axes of the coordinate system. They tend to increase the user interface complexity greatly and create the potential for the well-known problem of mode errors. Although practice mitigates these problems, software using these strategies tends to have a steep learning curve, requiring years of practice to master.

Another approach is to constrain the movement of objects according to physical laws such as gravity and the inability of solid objects to inter-penetrate each other. Such constraints can also be used to limit object movement according to human expectations [Smith et al., 2001]. For example, chairs sit on the floor, and desk lamps sit on top of desks. A problem with this approach is that it lacks generality, as it requires object-specific constraints to be designed a priori for each available type of object in the virtual environment. As such, constraints may be suitable for games, as they typically include only a limited set of objects in a restricted environment. For systems that allow custom object creation, or have a very large number of objects available, more general approaches are preferable.

One such general approach is based on the observation that in the real world (almost) all objects are attached to other objects and hence objects in the scene remain in contact with other objects at all times [Oh & Stuerzlinger, 2004; 2005]. To achieve this, the movement algorithm uses the surfaces occluded by the moving object to determine the current movement surface, while still avoiding collisions. An extension also allows users to move objects partially behind other objects. If an object is moved over the background, it moves in free space on a plane orthogonal to the viewer. The result is that the object being moved always slides “over” the other surfaces in the scene in a predictable way. The algorithm does not use the notion of gravity; i.e., one can move objects from the floor to walls or onto the ceiling and back. For efficiency, most of the computations are performed in graphics hardware. It detects occlusion between objects using the depth/stencil buffers, and uses the entire area of an object's face to slide objects along the faces of other objects. Full implementation details are described in Ji-Young Oh's doctoral dissertation [Oh, 2005]. This thesis presents extensions to that work, adapting the sliding movement technique for use with 3D input devices.

It is also interesting to note that while a large number of games use a mouse for 3D *navigation* (e.g., first-person shooters such as Doom3, Half-Life, online games such as World of Warcraft, etc.), few games allow 3D *manipulation* of any degree. One exception, Black & White 2 from Lionhead Studios allows movement of 3D objects in the game world using the mouse as a metaphorical hand [Lionhead Studios, 2006]. Clicking objects picks them up and holds them in-hand. The game's physics engine constrains objects to move according to user expectations when objects are released, or thrown. However, orientation of objects is seldom, if ever, relevant to the game, and other than rotating the view around an object before grasping it, no facility is provided for rotating grasped objects.

Although it does not use a mouse, another example of object manipulation in games comes from Nintendo's Metroid Prime 3 [Nintendo, 2007]. The game requires frequent interaction with objects, allowing the player to pull the Wii's motion sensitive controller away from the screen to pull an object out of a terminal. The player must then twist the controller (about the Z-axis) to turn the object, and push it back to reinsert it into the in-game terminal. Another example is Konami's Elebits, [Konami, 2007] which allows much more robust object manipulation. It is possible to move and rotate objects in 3D using the remote, and the game's physics engine ensures that the objects behave in a fairly realistic way. However, the game seldom requires precise placement of objects, and instead encourages the user to haphazardly throw them around. Although highly constrained and limited, these examples suggest that object manipulation may become more prevalent in games as novel interface technologies appear. Consequently, finding intuitive 3D manipulation techniques for use with 3/6DOF input devices will become more important as these tasks become more common in commercially available games.

2.1.3 Physical Support, Passive Haptic Feedback and Proprioception

One property of the mouse that is simultaneously a great advantage and a great limitation is the fact that it requires a physical surface upon which to work. Not only does this help prevent fatigue by allowing the user to rest their arm but it also steadies the hand, preventing jitter that can result in decreased movement precision. However it also makes the mouse largely unsuitable for certain types of 3D environments such as CAVEs, since it constrains the input to locations where a tabletop surface or similar is present. This problem is exacerbated in virtual environments using head-mounted displays, as the user is also unable to see the device itself [Lindeman et al., 1999b].

Nevertheless, the support surface property has not gone unnoticed in the virtual and augmented reality communities. Mine et al. discuss the use of proprioception as a first step toward compensating for the lack of physical support surfaces and haptic feedback in most virtual environments [Mine et al., 1997]. Proprioception is the sense of the position and orientation of one's body and limbs. It allows one to tell, for example, the approximate position of one's hand relative to the rest of the body, even when the eyes are closed. Their work proposed the use of proprioception for fixed-body position and gestural controls in a virtual environment. For example, upon selecting an object for manipulation, a user could delete it by throwing it over their shoulder – a logical mnemonic that is difficult to invoke accidentally and employs the user's proprioceptive sense. They also developed user-centred widgets that behave like tools for indirect manipulation of objects at a distance. Unlike the object-centred widgets commonly used in 3D graphics applications [Connor et al., 1992], these widgets are centered at the user's hand, and operate like tools on other objects in the environment, rather than being present on every object in the environment. Their experiments showed that users were able to perform 6DOF docking tasks more effectively with objects attached to their hands, and preferred widgets centred on the hand more than those floating in space. The authors reason that proprioception made these techniques easier to use than the alternatives [Mine et al., 1997]. One problem with these types of approaches is that gestural interaction requires the user to memorize specific motions in order to activate the desired operation. Of course, this issue is mitigated through intelligent mnemonic design, such as the delete action mentioned above.

Later research built on this idea by adding actual mobile physical support surface to these types of environments. Most notable among these are the personal interaction panel [Szalavári et al., 1997], Poupyrev's virtual notepad [Poupyrev et al., 1998b], and Lindeman et al.'s HARP system [Lindeman et al., 1999a; 1999b]. These approaches present virtual interfaces overlaid over a real physical surface, often a pressure-sensitive tablet, which the user carries around with them. The virtual representation of the slate is registered with its real-world position, and can either feature 2D or 3D user interface widgets on it. The user typically interacts indirectly with the environment via the user interface displayed on the slate. In a sense, the idea behind these interfaces is to combine the best aspects of 2D and 3D user interfaces – a full 3D virtual environment, in which the user can navigate, coupled with and controlled by a more familiar 2D interface. These often use a 3D tracked input device (e.g., a stylus) to determine which UI widgets are being activated on the physical surface. Another similar idea is to utilize a secondary interaction surface such as a tablet PC [Chen et al., 2005]. Not only does this provide a physical support surface, but also a familiar interface displayed on a secondary touch-sensitive screen.

Another approach [Kohli & Whitton, 2006], reminiscent of Mine's work [Mine et al., 1997], does not require the use of a tablet or secondary display. Instead, the user's non-dominant hand is tracked, and a virtual tablet is rendered registered with the hand. This is based on the premise that it is sometimes inconvenient to carry a secondary display or physical prop. Passive haptic feedback is provided by pressing against one's own hand while interacting with widgets displayed on the virtual tablet. However, most of these approaches still involve a very strict separation of the 2D and 3D interface components, thus increasing the cognitive overhead for the user.

Finally, other work has compared 3D interaction on and off tabletop surfaces, to assess the importance of passive haptic feedback in an environment where the display and input space are coupled [Wang & MacKenzie, 2000]. Using a VR workbench, participants performed several object manipulation tasks with their hands, on the tabletop surface, above the tabletop surface, and with the tabletop surface completely removed. They found that object positioning was significantly faster due to the tabletop supporting surface, but that accuracy was slightly worse. There is a key difference between this work, and that presented in this thesis. Like a mouse, or bat ("flying mouse") [Ware & Jessome, 1988], the control space is disjoint from the display space in the present work. Users do not operate directly on virtual objects (e.g., with their hands), but instead manipulate the device to indirectly control objects in the environment.

2.2 Stereoscopic Graphics and Head-Coupled Perspective

Several studies have been performed to determine the benefit of head-coupled perspective in fish tank VR, including [Boritz &Booth 1997; 1998; Mulder & van Liere, 2000]. Previous work [Boritz & Booth, 1997; 1998] conducted studies with 6DOF input devices for 3D manipulation tasks in fish tank VR systems, and compared stereoscopic to monoscopic display with and without head tracking. Both studies showed that stereo viewing was significantly better than mono, allowing quicker task completion, but no significant effect was found for head tracking. The authors reason that their tasks required only minimal head movement after the initial discovery of target locations. They note that although positional error was reduced in the stereo viewing mode, there was no significant difference between stereoscopic and monoscopic for rotational error.

Limited field of view can be a concern in fish tank VR systems as it is difficult to physically turn around in them. This is because the display must always be in front of the viewer. It has been proposed that exaggerated head rotations in a fish tank VR system can compensate for this limited display space [Mulder & van Liere, 2000]. The authors used the idea of rotation amplification to double the effective scene rotation. For example, if a user turned their head 10 degrees to the left, the scene would rotate 10 degrees to the right about their head. They compared this technique to a scene rotation technique using a 3D wand, in a search and selection task. However, their findings suggested that the head-coupled mode provided no significant improvements in speed over the wand version. There was also no subjective preference for one technique over the other.

Previous work has also focused on amplifying user head rotations to virtually extend the range of vision using head-mounted displays [Jay & Hubbold, 2003]. As with fish-tank VR, head-mounted displays tend to have a very limited field of view, so users must physically exaggerate their head movement in order to achieve desired scene rotations. The authors instead used software to amplify head rotation, and consequently users did not have to exaggerate their motions. They found that participants made significant improvements in a visual search time when both their head and virtual hand rotations were amplified. Furthermore, subjective questionnaires found that users preferred the amplified state over the non-amplified one, as it made turning one's head in the virtual environment more similar to doing so in reality.

In general, the benefits of the extra depth cues provided by head-coupled perspective and stereoscopic graphics may be task dependent. It has been suggested that tasks with a higher depth complexity would benefit more from the addition of stereo graphics and/or head-coupled perspective. This is supported by previous work in which participants were able to more quickly trace a complex graph/tree structure when provided with the extra depth cues [Ware et al., 1993]. By contrast, the tasks in [Boritz & Booth 1997; 1998] were performed in simpler scenes, and required relatively few depth judgements by the users. In fact, the only depth judgements required were typically to ensure that the object being manipulated was within the extents of the target zone, along all three axes.

Chapter 3
Issues in 3D Manipulation

A variety of issues complicate direct comparisons between 3D positioning techniques or input devices. Some of these issues include perceptual factors (e.g., depth perception), presence or absence of constraints and physical characteristics of the devices themselves. This chapter deals with some of these issues, in an attempt to constrain the present research to a manageable subset of the myriad possibilities available for discussion.

3.1 Assumptions about 3D Manipulation

Several assumptions were made about the general 3D manipulation problem while undertaking the research presented in this thesis. Each assumption is based on empirical results from previous work, or conform to generally accepted practice in 3D UI design.

The first of these assumptions is that 3D manipulation tasks can typically be decomposed into the following three distinct phases:

1. The selection phase, during which the user indicates which object they intend to manipulate.
2. A positioning phase, where the selected object is brought into the vicinity of the target area.
3. A "fine-tuning" phase, where the object is rotated and positioned relative to the target.

The distinction between the first and second phase is the same as in Bowman et al.'s work [Bowman et al., 1999]. The third phase is based on the observation that few people, if any, extensively rotate and move an object simultaneously. Boritz and Booth conjecture that some amount of overlap appears to happen between movement and rotation tasks [Boritz & Booth, 1998], but overall, rotation time dominates the 6DOF manipulation task. While experts may rotate and translate an object simultaneously, novices do not appear to do so. This is likely due to the relative unfamiliarity with 6DOF devices that are capable of this feat, and the high degree of coordination required to do so.

A second assumption is that further decomposition of these manipulation phases appears unwarranted, at least for novice users. It is intuitively more cumbersome to manually move an object if the positioning phase is decomposed into movement along each of the three axes. The approach presented by the SESAME system [Oh, 2005; Oh & Stuerzlinger, 2004; 2005] is that the entire act of positioning an object be handled at once, without requiring the user to think in terms of movement along each of the three separate axes. This is somewhat similar to how movement of a 2D object works – no extra movement widgets are necessary, as mouse motion maps directly and intuitively to object motion. Note that 3D rotations introduce a whole new layer of complexity to the problem. Consequently, this thesis is limited to discussion of positioning objects – a 3DOF task – rather than the complete 6DOF manipulation problem that includes both positioning and orienting objects.

A third assumption is that ray casting is the best technique for selecting 3D objects, rather than direct 3D selection techniques. This follows from several previous works [Bowman et al., 1999; Poupyrev et al., 1998a]. Furthermore, other work confirmed that ray-casting works well even with 2D input devices and is in fact better suited to it than a 3D selection cursor [Ware & Lowther, 1997].

A fourth assumption is that all objects can be constrained to remain in contact with the remainder of the scene at (almost) all times [Oh, 2005; Oh & Stuerzlinger, 2004; 2005]. This is based on the observation that in the real world, gravity ensures that very few objects float in space. In other words, contact is a reasonable default for most virtual environments, with the exception of flight and space simulations. Experiments studying this type of contact-based constraint revealed that the contact assumption is particularly beneficial for novice users, but even experts benefit from it [Smith et al., 2001; Oh, 2005].

A fifth assumption is that collision avoidance is beneficial for 3D manipulation. Fine positioning of objects is greatly aided by the ability to slide objects into place with collision detection/avoidance [Kitamura et al., 1998]. One reason for the effectiveness of collision avoidance is that novice users of graphical systems often become confused when objects interpenetrate one another, and can have difficulty in resolving the problem once objects are in collision. After all, solid objects in the real world cannot interpenetrate one another. Hence collision avoidance is a reasonable default – similar to the contact assumption [Teather & Stuerzlinger, 2007].

The above assumptions are reasonable because they greatly improve the immediate usability of any such system. Without these design decisions such systems typically require a great amount of training and are only usable by experts. This is clearly unsuitable, as the goal of this research is to determine how to improve immediate usability of VR systems for novices.

3.2 The SESAME 3D Movement Technique

The 3D movement technique used relies on the ideas of contact-based sliding and collision avoidance, as described in Section 3.1. This algorithm ensures that the object being moved remains in contact with other objects in the scene at all times [Oh & Stuerzlinger, 2005]. Objects are selected via ray casting based on the cursor position. Following object selection, the user can simply move the input device to "drag" the object across the scene. This is inspired by the "click 'n' drag" metaphor popularized by desktop computing. The algorithm handles depth automatically and keeps the object under the cursor; i.e., an object simply slides across the closest surface to the viewer that its projection falls onto. Figure 3-1 depicts how mouse motion maps to object movement in this system. When moving the mouse forward, the selected cube first slides along the "floor" of the scene. Upon detecting contact with the larger block in the background, the selected (moving) cube then slides up and over the front side of the stationary cube. In other words, the forward mouse movement will alternatively move the cube along the Y or Z world axes, depending on contact detection with other surfaces that constrain its movement in that direction.

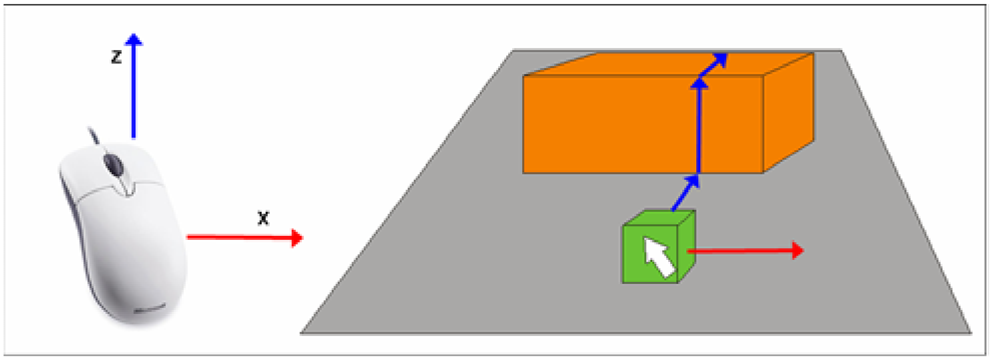


Figure 3-1 : Mouse motion to 3D movement mapping in SESAME.

Essentially, this technique reduces 3D positioning to a 2D problem, as objects can now be directly manipulated, and are moved via their 2D projection. This technique was chosen for two reasons. First, previous research has indicated that it is among the most efficient 3D movement techniques and novices learn it more quickly than other common techniques, such as 3D widgets [Oh & Stuerzlinger, 2005]. Moreover, the technique appears to be well-suited adaptation for usage with 3DOF/6DOF input devices.

Essentially, this technique ignores the third DOF of the 3DOF input device, and uses only movement in two directions, almost exactly as with a mouse. This allows other features and limitations of the 3D tracker to be explored while avoiding confounding factors due to the differences between the mouse and tracker. Figure 3-2 depicts the mapping of a 6DOF wand input device to object motion using the same 3D movement technique. In this case, movement of the device in the XY (vertical movement plane) is mapped to 2DOF, but the XZ plane (horizontal movement plane) could also be used. This would effectively make the tracker even more similar to the mouse.

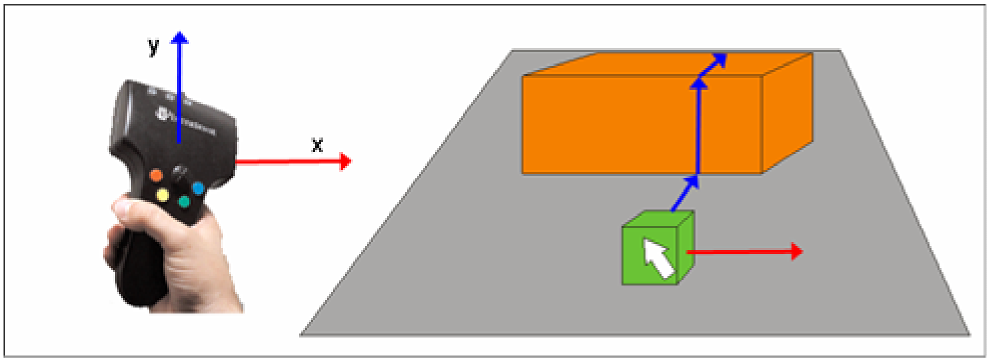


Figure 3-2 : Wand motion constrained to 2DOF operation via SESAME movement technique.

To contrast, Figure 3-3 depicts a 3DOF movement mode. In this mode, objects move according to the position of the 3D wand input device. Upon selecting an object, moving the wand up will move the object up. Moving it towards the screen will move the object into the scene, etc. A technique modeled after this type of movement is used in the preliminary study presented in Chapter 4, and in the second study presented in Section 5.2. This technique is representative of the so-called “virtual hand” techniques commonly used in virtual reality. Objects are selected via ray-casting, but it does not, however, use any of the advancements demonstrated by arm-extension techniques such as Go-Go [Poupyrev et al., 1996]. In a sense, this technique is probably most similar to the HOMER technique [Bowman & Hodges, 1997], which uses 3D ray-casting for selection and brings the virtual hand to the object upon selection. A key difference here is that there is no “virtual hand” moved toward the object. Selection is performed through the system cursor, and once an object is selected, it moves freely in space according to the movement of the 3D tracker. Collision avoidance can be optionally used with this technique as well, to provide a useful constraint.

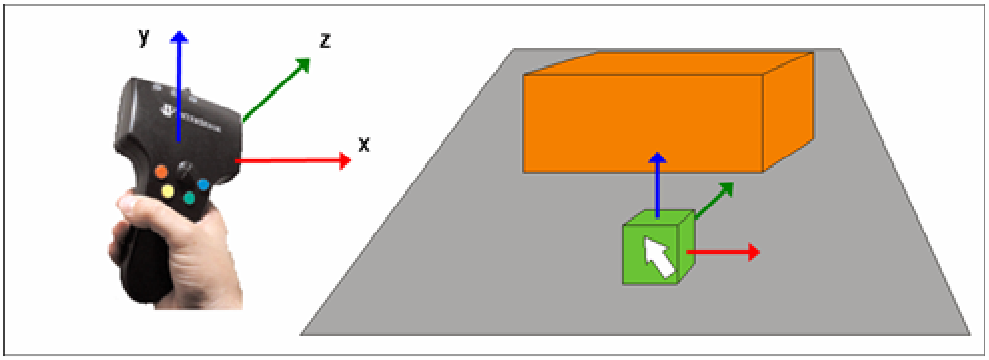


Figure 3-3 : "Wand3D" movement - full 3DOF motion with the wand.

The use of the tracker in Wand3D mode is probably most similar to Ware's "bat" input device [Ware & Jessome, 1988]. It was suggested here that correspondence between the movement of the device and movement of objects is likely more important than spatial correspondence. This is also how a mouse behaves. Thus unlike many VR systems, it was decided to not register the position of any input device with the objects being manipulated display, and instead use the relative motions of the input device to control movement of the objects (and cursor).

3.3 Extensions to the SESAME System

SESAME (which stands for Sketch, Extrude, Sculpt And Manipulation Easily) was previously developed as a desktop 3D conceptual design system [Oh, 2005; Oh & Stuerzlinger, 2004; 2005]. The goal of that research was to develop an intuitive 3D content creation system, for conceptual design; that is, high-level, "general picture" design tasks, as opposed to the detail-oriented tasks that the bulk of CAD and 3D modeling packages deal with (see for example, 3D Studio Max, www.autodesk.com). During the design phase, designers tend to create the overall vision of their work prior to focusing on details, and SESAME was created to support this type of task [Oh, 2005]. SESAME was shown to be quite effective for this task, but more importantly to this work, it provides the intuitive 3D movement technique described in Section 3.2. This thesis extends previous work done with SESAME by adapting it to a “fish tank VR” environment. The details of these extensions are listed below.

3.3.1 Stereo graphics

Binocular disparity is the property that arises from the lateral separation of our eyes. Consequently, each eye perceives a slightly different view of the same scene. This can be easily seen by focusing on an object, and then covering each eye in turn – the object appears to move. Stereopsis allows our brain to fuse these two different perspectives into a 3D representation, allowing us to more easily perceive the distance between objects [Hsu et al., 1996]. Stereopsis or stereo vision is thus considered one of several depth cues. Other strong cues include occlusion (closer objects block more distant ones in the same direction), perspective (more distant objects appear to be smaller) and motion parallax (apparent motion of objects based on movement of the viewer).

The illusion of stereoscopic depth can be created in a virtual environment by generating left and right-eye images, and through the use of some additional hardware technology, ensuring that only the corresponding eye sees each image. In this research, this was accomplished through the use of so-called “active” or frame-sequential stereo, which requires a high refresh rate (typically CRT) display and shutter glasses. The shutter glasses use liquid crystal to block each eye in synch with the appropriate image being displayed on screen. Typically, this must be done at around 120Hz for proper fusion of images to occur, and to reduce flicker. If done properly, this creates the illusion of depth in the display. When using a CRT monitor for this, the effect is like looking into a box with a glass front, similar to an aquarium, and is thus dubbed “fish tank virtual reality” [Ware et al., 1993].

Stereo graphics were added to SESAME for the purpose of this research. This was accomplished by adding a second virtual camera to the system, whose position was linked to the first, with a distance of approximately 6cm between them. Any viewpoint movement or rotation was simultaneously performed by both cameras, to ensure that both were always viewing the same scene from the slightly different viewpoint required for stereo fusion to occur.

This was the first step toward a (fish tank) VR version of SESAME, as the overall effect of stereoscopic graphics is to increase the immersion of the virtual scene.

3.3.2 3D Tracking

The second major modification of the SESAME system used in this thesis was the addition of 3D tracking. SESAME was originally designed for exclusive use with a desktop mouse. The addition of 3D tracking code, for use with an Intersense IS-900 system permitted the use of 3D wands and trackers. These devices were subsequently used as input devices, and for head-tracking, in various parts of this research.

Head-tracking permits support of another depth cue mentioned above, namely motion parallax. When moving one's head in the real world, one's perspective on the scene changes slightly. Assuming the eyes remain looking forward, closer objects appear to move more rapidly than distant ones, relative to the viewer. This is the case when using a camera as well. This can be observed by looking out a car window while driving; close objects (e.g., guard rails, telephone poles, etc.) move by rapidly, while distant objects (e.g., mountains, the horizon, clouds, etc.) move by very slowly. This is another way in which we can gauge the relative distances to objects.

Coupling the virtual viewpoint to the user's head position provides this extra depth cue in VR. This can be accomplished through the use of a head-tracking device, and linking head movements to camera movements. This extension to SESAME was used in the initial preliminary study, presented in Chapter 4. Overall, when used with a fish tank VR system, the intent of such techniques (when coupled with stereo graphics) is to make it appear as though one is looking into a box with objects inside, and moving one's head appears to create the same relative motion that one would expect in the real world.

The same 3D tracking code was also used for tracking the position of the wand/tracker input devices used in some conditions. This was of primary interest to this thesis; namely, how do these devices compare to lower dimensional devices such as the mouse. As described in section 3.2, the movement technique originally developed for use with the mouse was also adapted for use with a 3D tracker input device, to effectively constrain the operation of the device to a plane.

Chapter 4
Preliminary Study

The primary goal of this study was to determine if the 2D-inspired SESAME movement technique could be adapted for use with 6DOF/3DOF input devices, and if so, how well. The use of stereoscopic graphics and head-coupled perspective are also examined. As previously discussed, these techniques have been shown to be somewhat beneficial in 3D manipulation/positioning experiments using 6DOF input devices. However, there has been little or no work addressing if these techniques provide benefits when using 2DOF input devices, in particular the mouse. It seems reasonable to hypothesize that they may benefit 3D positioning tasks with the mouse as well, since they affect how the user perceives, and thus, interacts with the scene.

4.1 3D Positioning Techniques Used

For the initial study, three different positioning techniques were compared. All of these used 2D ray casting for selection of 3D objects. The first technique used the mouse, with the assistance of SESAME's 3D sliding movement algorithm [Oh, 2005], discussed in Section 3.2.

The second input technique used a 3D wand input device, but used only two axes of motion, as described in Section 3.2 (see figure 3-2). The Y (up-down) motion of the wand was mapped to cursor movement in Y on the screen, and the X (side-to-side) motion of the wand was mapped to cursor movement in X. This technique used the

SESAME sliding movement algorithm as well. In effect, this creates a "mouse emulation" mode, although with a direct mapping between cursor movement and device movement (i.e., moving the device up moves the cursor up). No physical supporting surface was used with this technique in this experiment. This movement mode was used to investigate the differences between 2DOF and 3DOF input devices, and is referred to as "WandSlide" for the rest of this thesis.

The third input technique also used the 3D wand, but object movement was directly mapped to 3D position of the device. Although this mode did not use the SESAME sliding algorithm, selection was still based on ray casting. Upon selection of an object, the object moves in 3D according to the 3DOF motion of the wand. No collision detection/avoidance was used in this mode, which makes this a "raw" 3D direct manipulation mode. This is representative of traditional VR object movement techniques and is referred to as the "Wand3D" technique for the remainder of this thesis.

4.2 3D Positioning Tasks Used

Two different positioning tasks were used in this study. The first task, depicted in Fig. 4-1, involved the selection and movement of the central red target cube to the top of the pedestal. This task was based on a similar task used in previous work [Bowman et al., 1999]. It was chosen because the motion required to position the cube is relatively simple to perform with any input device and can thus serve as a representative "abstract" movement task. While somewhat overly simplistic, it may give an indication of how a series of movements comprised of such simple, short movements performs with different positioning techniques. Also, because the cubes were positioned in the foreground, and the target pillar was placed in the background, it was hypothesized that this task would help analyze any potential benefits of the extra depth cues provided by stereoscopic graphics and head tracking.

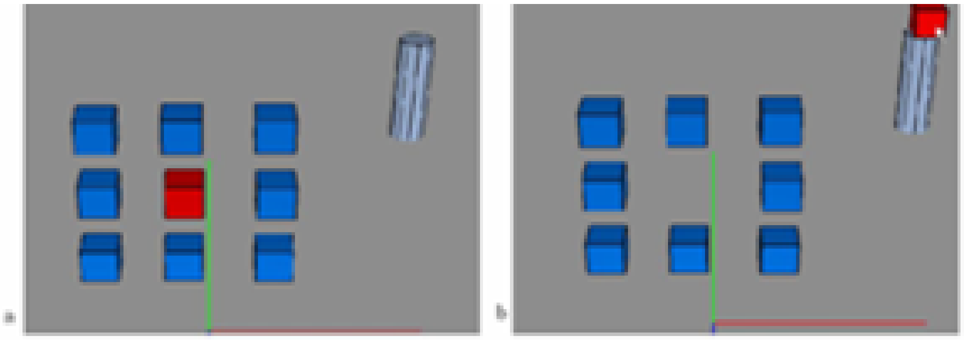


Figure 4-1 : The cubes task, starting and target scenes.

The second task was the assembly of a chair from several pieces (see figure 4-2). This task was chosen as a representative real-world assembly task. It is harder than the cube placement task, as it requires the accurate placement of multiple objects. This task was also previously used to compare the mouse sliding movement technique to 3D widgets [Oh & Stuerzlinger, 2005].

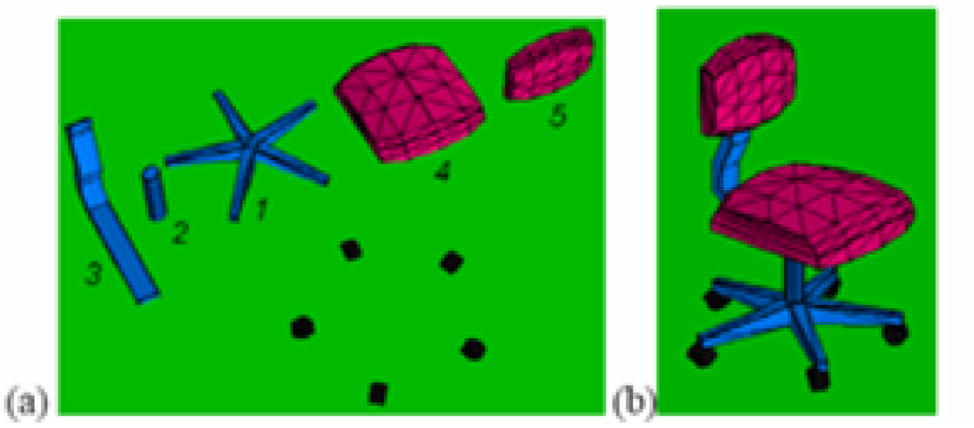


Figure 4-2 : The chair task, starting and target scenes.

Note that this task cannot be adequately handled by techniques that use only gravity and collision avoidance. The chair task involves the backrest (part #5 in fig. 2), that must be attached horizontally to the support behind it (part #3). Using gravity alone it is impossible to perform this attachment. The sliding paradigm easily handles cases like this, as the backrest can be slid up the support up to the desired position. The object then remains fixed to the position where it was released. Hence, the sliding movement technique is believed to be more appropriate for assembly tasks compared to traditional approaches such as 3D widgets.

4.3 Hypotheses

Hypothesis 1: 3D positioning technique

The first hypothesis was that mouse mode would outperform the two conditions using the 3D input device. In addition to the effect of extensive user familiarity with the mouse, the reduced hand jitter in this condition should favour 2D input over 3D input. Another factor that should play a role here is the absence of collision detection in the Wand 3D mode.

Hypothesis 2: Stereoscopic graphics

The second hypothesis was that the addition of stereoscopic graphics would improve the participants' ability to position objects in 3D, thus reducing task performance time and improving accuracy due to the extra depth cue provided. In other words, stereo should make it easier to perform 3D object positioning, even with 2D input devices.

Hypothesis 3: Head-coupled perspective

The third hypothesis was that the addition of head coupled perspective would also improve accuracy. The extra motion depth cue provided by head coupling should assist users in gauging depth better, thus obviating the need to rotate the entire scene.

4.4 Participants

Twelve paid volunteers participated in the study, with age ranging from 23 to 34 years, mean age 25.7 years. Seven participants were male. Nine of the twelve reported using a mouse for 10 or more years, the remainder reported 5 – 10 years of experience. Since approximately 8% of the population is incapable of seeing stereo depth [Hsu et al., 1996], participants were also screened for stereoscopic viewing ability.

Participants' game playing habits were also recorded, as it is possible that they were a confounding factor. It was found in a course project experiment on 3D object movement that gamers tend to skew the results of studies of 3D interaction with 2D input devices [Teather, 2006]. In that experiment, gamers performed significantly better that those with limited video game experience. Only one participant reported playing games more often than once per week. Two others reported playing games roughly once every week, and the rest played approximately once per month, or less frequently. Based on similar reasoning, participants were also asked about prior experience with 3D modeling tools. The majority of the participants had little to no experience with 3D modeling, with seven having never used such software, and the remaining five only using it approximately once per month, or less frequently.

4.5 Apparatus

Tasks were performed in a fish tank VR system. The system was an AMD Athlon 64 1.81GHz with 1GB of RAM, and an NVIDIA Quadro FX3400 graphics card. A standard desktop optical mouse was used as input device in one condition, and an Intersense 6DOF wand was used in the other two. Stereoscopic graphics were presented using a Stereographics emitter and CrystalEyes shutter glasses. The Intersense IS-900 was used for 3D tracking of the participants' head and the wand input device. The head tracking sensor was mounted on the shutter glasses. The display was a Silicon Graphics monitor at 1024x768 @ 120HZ. The software used was a modified version of the SESAME software written in C++ with OpenGL. The modifications were Intersense tracking code and stereo pair rendering, as discussed in Section 3.3.

In stereo mode, using the system cursor with stereo graphics produces a "dual cursor" effect when a user focuses on the cursor. To avoid this, the software was modified to only draw the mouse cursor synchronized with the dominant eye, as discussed in [Ware & Lowther, 1997]. This one-eyed cursor was aligned to the position of the operating system cursor, to allow accurate selection of objects as required by the experimental tasks. Some of the objects being moved were relatively small, thus a degree of precision was required to perform the tasks. This one-eyed cursor helped improve precision of the participants. Figure 4-3 depicts the experimental setup.

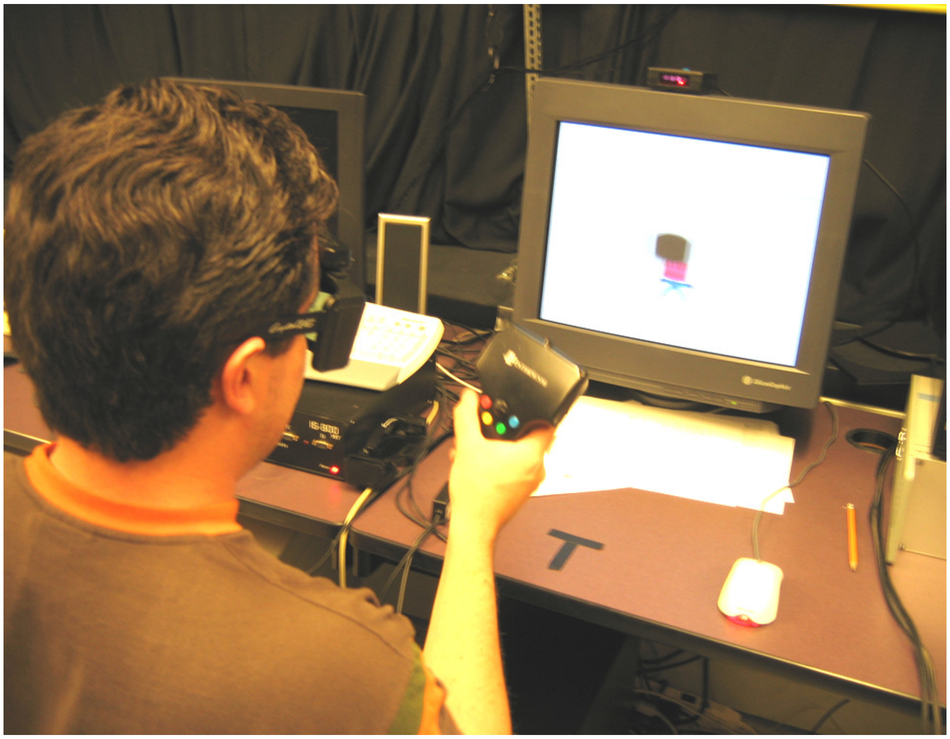


Figure 4-3 : Experimental equipment setup for preliminary study.

4.6 Procedure

In each trial, participants used one of the three movement techniques from Section 4.1 to complete either the chair or cubes task. For the chair task, participants were informed of the order in which parts should be assembled, and were asked not to move the chair's wheels. Hence, they started with part #1 in Figure 4-2 (the base of the chair). This ensured that the experiment was not testing 3D construction skills, but rather the input techniques and display modes. Prior to both tasks, participants were given a brief practice period in each movement mode of up to 5 minutes (total) to familiarize them with the 3D sliding movement algorithm used in the system, as well as the various input devices. During the experiment, participants repeated each task twice for each condition.

In all trials, participants were asked to complete the assembly or placement task as quickly and accurately as possible. Prior to each trial, participants were informed of the status of each of the experimental factors, namely, whether head-tracking and stereo graphics were on or off, and which input device and technique to be used for the trial.

4.7 Design

The experiment was a 3×2×2×2 design. The independent variables were movement technique (Mouse, WandSlide and Wand3D mode), display mode (monoscopic or stereoscopic), head tracking (enabled or disabled), and task (chair assembly or cube placement) respectively. All factors were within-subjects and there were two repetitions of each condition. The orderings of display type, head tracking mode, input device and task were counter-balanced with a balanced Latin square to compensate for possible asymmetric learning effects across conditions. Participants wore shutter glasses during all trials, to mitigate confounding effects of the glasses themselves. The glasses were inactive during trials that did not use stereo.

Every participant completed every combination of movement technique and display mode twice, for a total of 48 trials each. Participants took approximately 1 hour to complete this series of trials. In total, 576 trials were performed.

4.8 Results

A repeated-measures ANOVA on the task completion times for all trials was performed. A significant main effect for positioning technique ($F_{2,22}=34.348$, $p < .001$) was found. Tukey-Kramer post-hoc analysis revealed that all three techniques were different; the Mouse technique (mean 19.3s) outperformed the WandSlide technique (27.3s), which in turn outperformed the Wand3D technique (33.6s). ANOVA failed to find a significant effect of both stereo and head-tracking. Participants performed significantly better upon the second repetition of each trial ($F_{1,11}=19.24$, $p < .01$). The mean completion time was 40.9s for the chair task, and 12.5s for the cube task; these were also significantly different, ($F_{1,11}=64.053$, $p < .001$). Beyond that, there were no significant differences, with the exception of a significant interaction between task and positioning technique ($F_{2,22}=17.574$, $p < .001$).

Task completion times were also analyzed on an average time per-object basis. These numbers were found by dividing the task completion times by the number of objects moved in the scene. In this case, the average task completion time per object for the chair task was 10.2s; this was still significantly different from the cube task's completion time per object of 12.5s ($F_{1,11}$=23.67, p < .0005). However, the significant interaction between task and positioning technique noted above for total task completion times was not found for per-object completion times ($F_{2,22}$=0.94, ns).

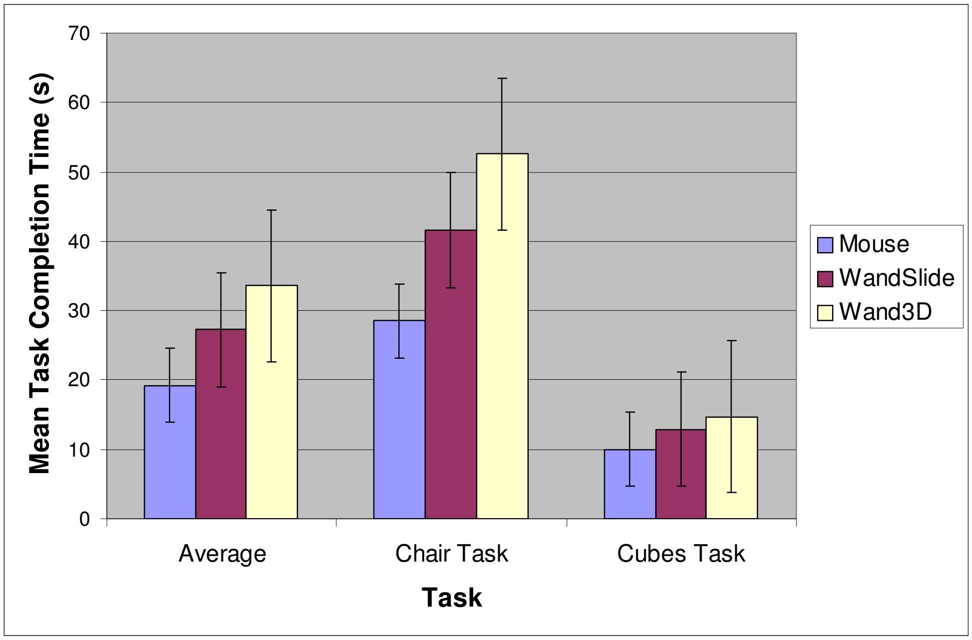


Figure 4-4 : Mean task completion time by task and movement technique.

Accuracy was measured by summing the total straight-line 3D error distance for each object in the scene compared to the target scene. There was a significant difference in accuracy for positioning technique ($F_{2,22}$=17.122, p < .001) and for task ($F_{1,11}$=17.172, p < .001). The mean total error by positioning technique was 4.8 cm for the Mouse mode, 5.89 cm for WandSlide, and 15.6 cm for Wand3D. Post-hoc comparisons indicated no significant difference in accuracy between the Mouse and WandSlide modes – both of these modes were significantly more accurate than the 3DOF movement technique. Stereo display mode was also found to significantly improve accuracy ($F_{1,11}$=7.982, $p < .05$). The mean positioning error by stereo mode was 11.48 cm for stereo and 13.62 cm for mono.

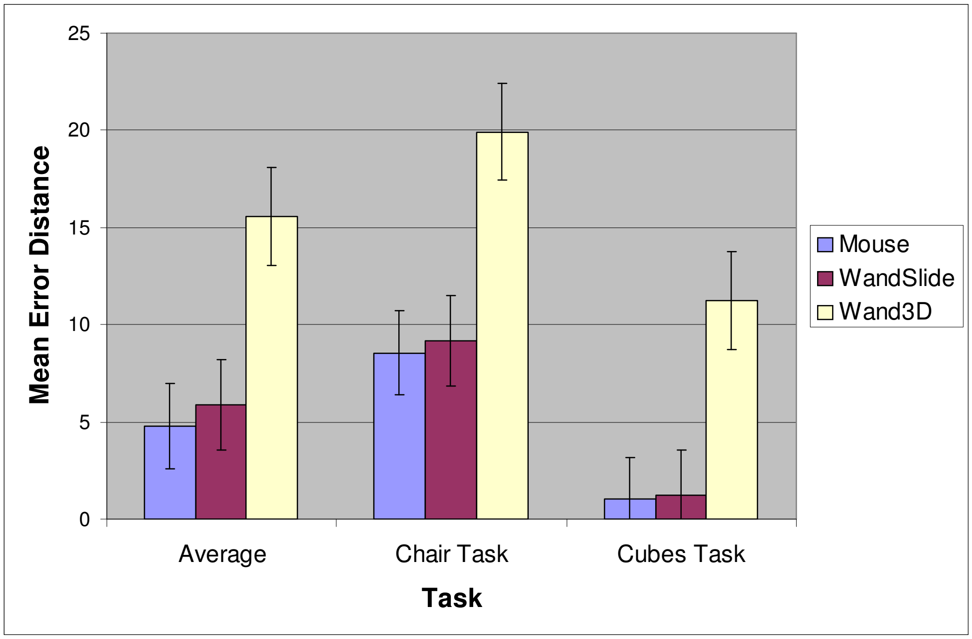


Figure 4-5 : Mean error distance by task and movement technique.

A significant interaction effect was also found between stereo and positioning technique ($F_{2,22}$=5.86, $p < .01$), indicating that stereo had a greater effect in the Wand3D mode than in the other two modes. These results are summarized in Figure 4-6.

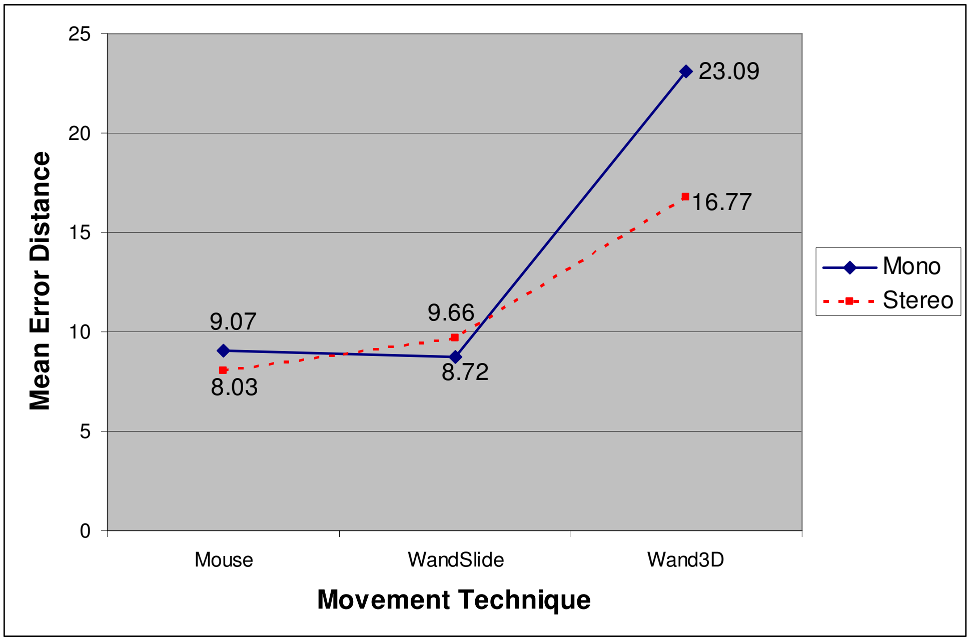


Figure 4-6 : Interaction effect between movement technique and stereo mode.

4.8.1 Demographic Analysis

It was suspected that previous experience playing video games and using 3D modeling software may skew the results of this experiment. To determine if gamers performed better than non-gamers at these types of tasks, the participants were grouped based on their response from the questionnaire. If they responded that they played games at least several times a month (a 3 or higher on the 5 point scale used) they were classified as a "gamer" for the purpose of this analysis. Five of the twelve participants fell into this category.

Due to the relative complexity of the task, only the chair task was analyzed for this. A one-way ANOVA on the chair task completion times revealed no significant difference between gamers and non-gamers ($F_{1,286}$=1.76, p > .05). The means for the groups were 42.36s for non-gamers, and 38.83s for gamers. However, it is possible that the several outliers depicted in Figure 4-6 may have influenced the results. The figure suggests that on average, there may be a significant difference.

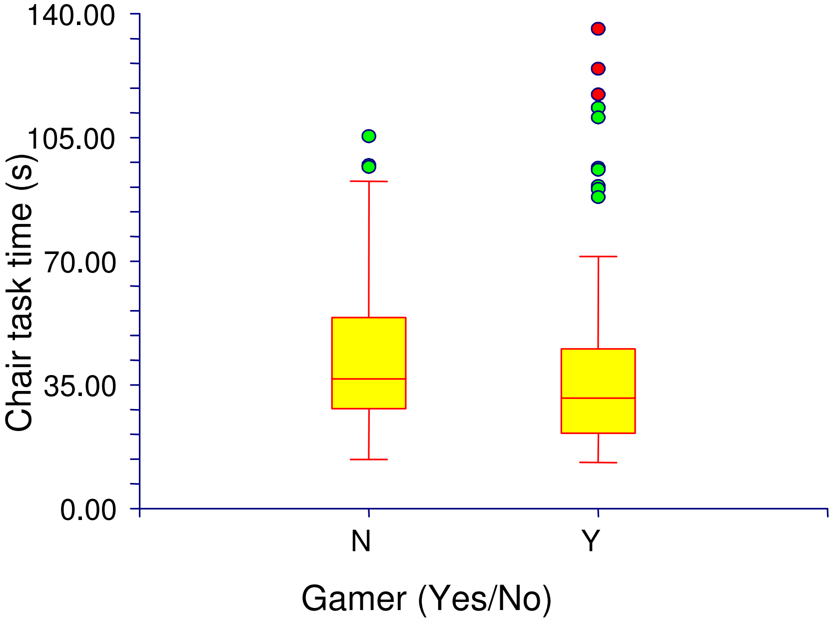


Figure 4-7: Box plot for chair task completion times by gamer experience. Higher values are worse.

The three worst times (two of which were the same participant in two different conditions) were removed, and the analysis was repeated. This time, a significant difference was found ($F_{1,283}$=5.40, $p < .05$), with gamers (mean 36.63s) significantly outperforming non-gamers (mean 42.36s).

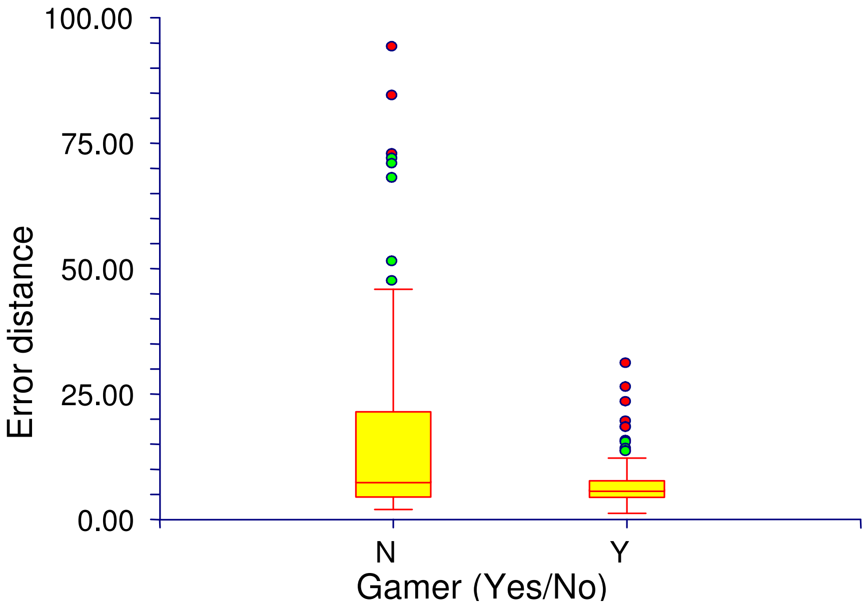


Figure 4-8 : Box plots of chair task error distances by gamer experience. Higher values are worse.

This analysis was also performed on accuracy for the chair task. A significant difference was found between gamers and non-gamers ($F_{1,286}$=36.57, p < .00001). Gamers were significantly more accurate (mean error distance of 6.95 units) than non-gamers (mean error distance of 16.56 units). Outliers were not removed, as they did not appear to affect the significance of the result. Figure 4-7 shows the box plot of these values. Given the large number of outliers, however, it suggests that this data was likely not normally distributed.

Note that in statistics, parametric tests such as ANOVA rely on specific assumptions [Pagano, 2007]. One such assumption is that the data being analyzed is normally distributed, and the other is that variance between populations is more or less the same (homogeneity of variance). The ANOVA test is relatively robust with respect to violations of these assumptions. However, in general, when these assumptions are violated, parametric tests are unreliable, and nonparametric tests should be used instead to ensure validity of the results [Pagano, 2007]. This is also the case when the data being analyzed are non-continuous (e.g., results from a questionnaire are typically not normally distributed as they are non-continuous). A nonparametric equivalent to ANOVA is the Kruskal-Wallis test, which does not rely on these assumptions. Because the data being analyzed was most likely not normally distributed, a Kruskal-Wallis ANOVA was conducted. This also indicated that they were significantly different (H_1=18.46, $p < .00005$). This test is used again later as well to compare survey scores and subject preferences for significant differences.

It would be interesting to also similarly analyze prior 3D modeling experience. However, too few participants had enough regular experience with such tools. Consequently, this analysis was not performed.

4.9 Discussion

Prior to conducting the experiment, it had been considered to split the conditions into two separate experiments; one to compare *just* the movement techniques, and the other to compare *just* the display modes. However, this would have made determining potential interactions between conditions nearly impossible. Since these interactions were of particular interest, it was decided to include all in this single experiment. This allows analysis to determine if the addition of stereo and/or head-tracking aided any specific positioning technique more than others.

For example, the combination of stereo and head tracking may aid the mouse more than the 3D input device, or vice versa. If the display conditions and input device conditions had been split into two separate studies, this would be very difficult to discover. It was thus decided that later studies could address specific components, and split this large design into several smaller ones, based on the findings from this study.

4.9.1 Movement Techniques and Input Devices

The significant difference in speed between tasks was unsurprising. Obviously, the cube placement task was far simpler than the chair assembly, requiring only a single precise object placement, rather than multiple actions.

The fact that full 3DOF movement with the wand took longer than the other two modes confirmed the first hypothesis in section 4.3. There are several likely causes for this result.

The first, as mentioned, is the participants' familiarity with the mouse compared to the wand. Essentially, the participants were already experts with the mouse but had no experience with the wand. This gives an advantage to the mouse. Second, because the Wand3D condition used neither collision detection nor front-face sliding like the Mouse and WandSlide movement modes, participants required additional time to accurately position the selected object in 3D. Some participants commented on this, that the lack of collision detection and/or collision feedback made it difficult to judge when the object was positioned correctly. Another aspect is that hand jitter and fatigue combined with the relative sensitivity of the wand reduced the accuracy of the Wand3D technique significantly, compared to the other two techniques. It is likely that the participants took extra time trying to correct for this reduced accuracy, eventually giving up when the scene looked “good enough”. This is substantiated by the significantly worse accuracy with this technique. However, despite using the same input device, this factor did not appear to affect the WandSlide technique. This is most likely because the collision avoidance and contact-based sliding overcame the hindrance caused by jitter.

Third, observations made during the experiment suggest that participants came to rely on the front-face sliding movement after they had been exposed to it in the WandSlide and Mouse conditions, often leaving objects floating well in front of their intended target in the Wand3D condition – an oversight that the 2D sliding algorithm automatically accounts for. This even occurred during stereo and head-tracked trials, where it had been hypothesized that the additional depth cues provided would aid the users’ accuracy (Section 4.3). This suggests that the input technique has a much stronger effect on accuracy and speed than either stereo or head tracking.

Finally, the 2D sliding algorithm used in both the Mouse and WandSlide modes effectively reduces the dimensionality of the movement task from 3D to 2D. This is a clear benefit over “full 3D” movement techniques, as the user is only required to position the object accurately in two dimensions rather than three. Phrased differently, the user is only required to line up the image of the object being moved with the image of the target. Given that the task required only movement of objects on a planar surface, this technique maps very well to the functionality of a mouse. This is undoubtedly one reason for its relatively better speed and accuracy than the wand. This suggests that smart 3D movement algorithms can overcome the limitations of an input device (e.g., degrees of freedom) and can allow such input devices to outperform devices that initially seem to be better suited to the problem. Although this is technically no longer a 3D positioning task, but rather a 2D positioning task, the end result is the same – the object has been moved to a new 3D location in the scene. However, constraining operation of the wand to 2D (i.e., ignoring the depth component, and using the same movement technique) also tended to permit better performance than full 3DOF operation. It is also possible that a movement task requiring free space positioning would be better suited to the wand than 2DOF techniques.

It is interesting that there was no significant difference in accuracy between the Mouse and WandSlide modes, while there was a significant difference in speed. It is possible that the reason for this is that the table on which the mouse slides provides a firm foundation upon which the participants' can rest their hand – and thus improves accuracy. Another factor is that the friction between mouse and table enables users to fairly rapidly stop their movement, compared to stopping a wand movement in the air. The wand, however, does not provide these benefits. Previous findings support this as well [Chen et al., 2005; Lindeman et al., 1999b; Poupyrev et al., 1998b]. Furthermore, the 2D sliding algorithm makes it quite easy to correct minor misplacements very quickly, hence the participants seemed more inclined to trade a bit of time for improved accuracy in this condition. Correcting such mistakes in 3DOF mode requires a significantly greater amount of work due to jitter and the additional axis that needs to be controlled simultaneously.

Despite the relative quantitative performance of the input techniques, several of the participants commented that they found the Wand3D mode to be the most fun to use. Given the recent success of the Nintendo Wii game console, which uses a similar input device [Nintendo, 2007], this is not surprising. However, several other participants also commented that it was frustrating to use, and that they preferred using the mouse. Interestingly, no participants chose the WandSlide technique as their favorite. "Fun factor" is an important consideration in interface design as well, especially for games. These findings suggest that if 3DOF interaction techniques could be made as effective as 2DOF techniques they may be a clear winner.

Finally, as expected, gamers tended to significantly outperform non-gamers. This is most likely due to the similarity between these types of 3D environments and games. It seems plausible that playing games would serve as a type of training for performing these types of tasks in 3D. It is possible that giving non-gamers a longer training period may have eliminated the differences between these groups.

4.9.2 Stereo and Head-Tracking

Hypothesis 2 in Section 4.3, regarding stereoscopic graphics was partially confirmed by the significant effect observed on accuracy. This conforms to previous studies, and as indicated, the extra depth cue allowed the users to more easily perceive the distances between objects. Surprisingly, however, head tracking had no effect on accuracy or completion time. This is likely because the participants seldom appeared to *intentionally* use the head tracking. One possible reason for this is that they simply forgot about it during the trials when it was active, despite being informed about the status of each factor at the beginning of each trial. It is also possible that they did not understand the full value of head tracking or felt the effect was too subtle to be useful. One participant even commented that the scene rotation by head movement would be more useful if the camera movement was exaggerated beyond realism. A third possibility is again related to the apparent reliance of the users on the front-face sliding movement algorithm – the users may have been assuming that the objects were sliding and that this feature was ensuring their accuracy, hence they felt they had no need to use the head tracking. Objects were often left floating far in front of the target, but appeared properly positioned in 2D. A subtle shift of the head in head-tracked mode would have revealed the distance between the cube and the target.

It has also been previously suggested that more complex scenes require more reliance on stereo and head-tracking [Boritz & Booth, 1997]. Because both of the scenes used in this experiment were fairly simple, consisting of only a few objects, only minimal view movements were required by the participants to determine the relative 3D location of the objects, which is yet another way to explain the lack of effect.

Finally, an interaction effect was found between input technique and stereo mode. This indicated that stereo graphics aided accuracy in the 3DOF mode more than either of the other two constrained modes. Earlier, it had been hypothesized that all positioning techniques, regardless of device, would benefit from the use of stereo, due to the additional depth cues provided. However, it seems that the benefit was minimal for the 2D constrained modes. This again suggests that movement technique characteristics have a stronger effect than display mode characteristics, at least for this type of movement task.

Chapter 5
Input Device User Studies

This chapter presents two related user studies, intended as follow-ups to further address the results of the preliminary study (see previous Chapter). The first of these studies, focused on support and orientation (Section 5.1). It was designed to determine the importance of a physical supporting surface for the hand when performing 3D object movement tasks. It also addresses device/display orientation congruence – namely, does matching, or mismatching the orientation of the device movement plane to the orientation of the display affect performance.

The second of these studies, the mouse and tracker study (section 5.2) is motivated by the results of the support and orientation study and also the preliminary study. As with the preliminary, it directly compares the mouse to the 3D tracker in a variety of conditions to determine if factors that were not considered in the study in section 5.1 may be responsible for the differences found in the preliminary study.

As mentioned in Section 4.7, splitting the display and input factors had been previously considered. These studies exclusively examine input factors, and ignore display factors.

5.1 Orientation and Support Study

The purpose of this study was to address differences between the input devices used in the preliminary study. There are a variety of factors which could have resulted in the mouse outperforming the Intersense 6DOF wand. User familiarity may play a big factor here; most people use a mouse extensively in day-to-day computing and have very limited experience with 3D devices. Another factor is the dimensionality of the task. It is intuitively more difficult to accurately position an object in 3D space than in 2D space, mainly due to the additional degree(s) of freedom in which the object can move.

Another factor is that the mouse requires a supporting surface on which to operate. This supporting surface reduces arm fatigue and hand jitter of the user, providing an advantage over the "free-floating" movement associated with most 6DOF devices. On the other hand, this is also a disadvantage for the mouse, as it is then unsuitable for virtual environments that require full 6DOF movement or for VR setups where a supporting surface is impractical or intrusive (e.g., CAVEs).

5.1.1 Comparing Input Devices

The goal of this study was to determine the relative importance of various factors that distinguish 3D interaction with a mouse from interaction with 6DOF input devices. Thus, the study compares interaction with and without a supporting surface, as well as the effects of input device movement orientation and display orientation. However, directly comparing two different input devices is problematic since it can be extremely difficult to account for all possible confounding factors that affect their performance.

One potentially confounding factor is clearly any differences in control space orientation [Wigdor et al., 2006]. Another is different hand positions used with different input devices. Both of these factors also relate to specific muscle groups that may be more or less developed and can affect fine motor control [Zhai et al., 1996]. In particular, input devices that use fine-motor control muscle groups, such as those in the fingers, can benefit precision manipulation. However, allowing several muscle groups in the arm to work together, rather than in isolation can be even better. This is supported by later work comparing muscle groups in the fingers, wrist and forearm [Balakrishnan & MacKenzie, 1997]. Their results show that using multiple muscle groups together tended to perform better than just using the fingers alone. Technical properties such as tracking accuracy and jitter levels, and physical properties such as size and weight can also impact performance. Furthermore, large differences in movement distances and/or cursor speed may also play a role.

Consequently, the test environment was designed to eliminate as many of these factors as possible. One of the main decisions for the first study was to use a 3D tracker as the input device for *all* conditions. However, users were also required to hold a mouse in the palm of their hand. This “flying mouse” device combination is very similar to the Bat [Ware & Jessome, 1988].

To evaluate the supporting surface while keeping the input device constant, users were required to move the tracker/mouse on a table. This effectively uses the tracker to emulate a mouse. However, the devices are not identical, as the mouse permits “clutching”; i.e., picking up the device to reposition it for long distance movements. Since the 3D tracker is an absolute positioning device, a direct mapping between a rectangular region on (or off) the supporting surface and the display was used. Thus, the tracker behaves similarly to a graphical tablet or "puck"; i.e., device position in a rectangular region maps directly to screen position.

In all conditions of this study, the SESAME sliding movement technique was used with the 6DOF tracker. In terms of the preliminary study (Chapter 4), the tracker was in "WandSlide" mode at all times, with one key difference: in some conditions, the device movement plane orientation was vertical, as in the preliminary study; in other conditions, it was horizontal, like a mouse.

5.1.2 Hypotheses

Hypothesis 1: Supporting surface

Based on the results of previous work [Poupyrev et al., 1998b; Lindeman et al., 1999a; 1999b; Wand & MacKenzie, 2000; Chen et al., 2005], it was hypothesized that participants would perform better in the supported conditions. The physical surface allows the user to rest their arm and hence reduces hand jitter, improving accuracy. Due to the inherent speed/accuracy trade-off in this type of object movement task, it was predicted that speed would also improve, as they would have to spend less time trying to accurately position objects.

Hypothesis 2: Display/Device orientation

A second hypothesis was that the conventional desktop display/device orientation combination would prove to be the best, due to the participants' familiarity with it. However, disregarding this factor, it was believed that users would generally perform better in conditions in which the movement plane of the input device matched that of the display, due to the direct mapping of input motions to cursor movement.

5.1.3 Participants

Sixteen paid participants took part in the study. Their ages ranged from 18 to 28, with a mean of 22.45 years. Only one participant was female. The average mouse usage for the group was 11.9 years. All participants used the mouse with their right hand.

Participants were asked about their previous experience with 3D games, since it is possible that gamers could outperform non-gamers in these types of tasks. They were also asked about their previous experience with 3D modeling software (e.g., 3DS Max and Maya) for the same reason. Responses were scored on a scale from 1 (never use/play) to 5 (use/play every day). The average score was 1.9 for 3D modelling software and 3.1 for 3D games, suggesting that most of the participants never or very seldom use 3D modelling software, and play games semi-frequently. In fact, only one participant responded that they never play games. Most replied that they at least played semi-frequently, so the bulk of the participants were considered to be approximately equal in this regard.

5.1.4 Apparatus

Tasks were performed in a desktop VR system (Figure 5-1), consisting of a desktop PC with stereoscopic graphics and 3D input. This was an Intel Pentium 4 at 3GHz with 512MB RAM, and an NVidia Quadro FX3400 graphics card. Two SGI monitors with

800 x 600 at 120 Hz were used for stereo display. Brightness and colour of these displays was adjusted to be as similar as possible. One monitor was positioned upright, and the other was supported on its back with hard Styrofoam. The horizontal monitor was inclined about 10° for more ergonomic viewing, while still maintaining approximate orthogonality to the vertical monitor. LCD shutter glasses and a Stereographics emitter were used for stereo viewing. Room lights were dimmed to equalize glare across both displays, since this could affect stereo viewing.

Figure 5-1 : The experimental equipment setup for the support study. Note: A different chair, which had no arm-rests, was used in the experiments.

An Intersense IS900 was used for tracking the 3D position of the user's right hand. In this hand, participants also held an optical mouse and its buttons were used to record “click” events. The optical sensor of the mouse was taped over. All cursor/object movement was recorded only by the 3D tracker, which was mounted on the back of a nylon glove worn in all conditions. Figure 5-2 depicts the position of tracker and mouse on a hand.

Figure 5-2 : Hand tracker and mouse for support study.

Since the tracker is an absolute positioning device, a small rectangle (15x11.25 cm) was marked out on the table, to visualize the mapping of movement to cursor movement on the screen. This area has the same height/width ratio as the screen. Upon starting each trial, the software registered the position of the tracker as the bottom left corner of the screen, and placed the cursor there. Participants were required to place their hand in that position at the start of each trial.

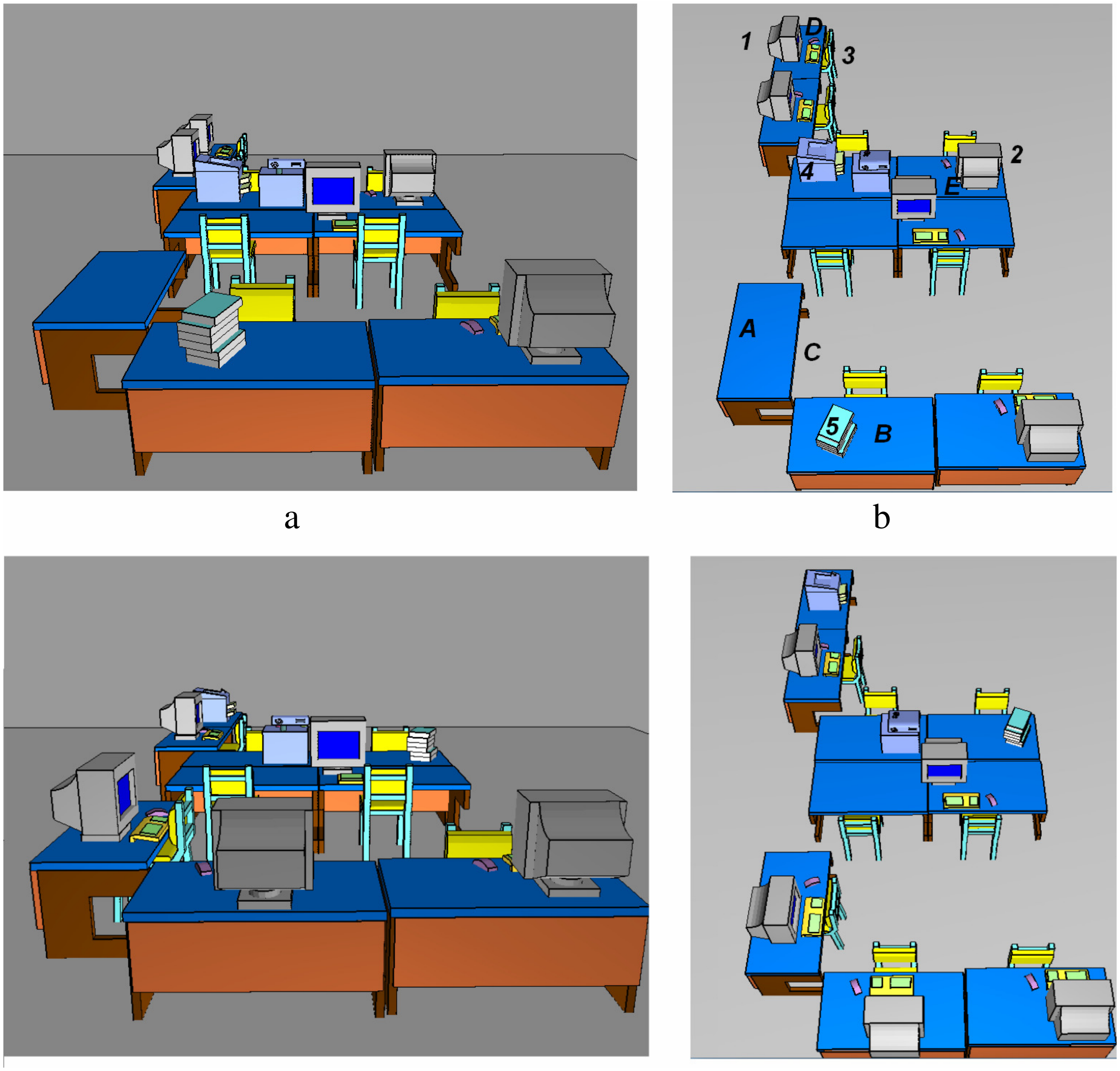
Hand support was provided by a table in the horizontal device movement condition and a sturdy cupboard on top of the table for the vertical input device movement condition (see Figure 5-1). These were moved out of the way in the unsupported conditions. Small marks on the floor and tabletop ensured that the physical supports were always in the same position when in use. The chair used did not have arm-rests, to prevent users from "cheating" and using this for support during the unsupported conditions. Thus the only hand support provided was from the table, or the cupboard in the appropriate support condition.

The software used was the same modified fish tank VR version of SESAME used in the preliminary study. The software was written in C++ and OpenGL/GLUT. The modifications to the software included stereo pair rendering to generate the stereoscopic graphics effect, and 3D tracking code for head tracking and to enable the use of 3D input devices. However, head-tracking was disabled throughout this study, and participants could freely move their heads. Stereoscopic graphics were enabled in all conditions.

5.1.5 Procedure

After an introduction to the experiment and signing informed consent forms, each participant was seated in front of the system and given the shutter glasses and the hand-tracker glove to wear. They were then given a single practice trial to familiarize themselves with the task and the sliding movement technique.

The experimental task (Figure 5-3) involved moving several pieces of furniture around a computer lab virtual environment. Participants were initially presented with a low-angle view of the scene, similar to Figure 5-3a.



a b

c d

Figure 5-3 : The experimental task – a) View of starting condition (what the participants saw in the first study), b) Overhead view of starting condition (for illustration only), c) View of target scene, d) Overhead view of target scene.

The task required that they move two computer stations to foreground desks, as well as a chair. A printer had to be moved from the second row to the back-most desk, and a stack of books from the front-most desk to the second row, right-most desk. Overall, the task involved moving object *1* to position *A*, object *2* to position *B*, and so on, as depicted in Figure 5-3b. Figure 5-3d shows the completed scene from an overhead view. Although complex, the task was intended to assess performance in a fairly realistic scenario, rather than examine abstract motions. This task was selected to make the results more generalizable.

Moving a computer station involved moving both the monitor and the keyboard. Users were not required to move the mouse objects in the model, because a pilot study found that it was too small to be selected reliably in some of the conditions. Thus the mouse object was excluded to ensure that the task could be completed under all conditions. In total, each trial involved the movement of 7 virtual objects, of sizes ranging from relatively small (the books) to relatively large (monitor and printer).

A certain degree of selection accuracy was also required in this task. For example, selecting the top book in the stack would only move that book; participants had to select the bottom book to move the entire stack.

Participants were given continuous verbal feedback throughout the experiment as well as reminders on the ordering if they showed signs of confusion about which object to move next. After two or three repetitions, they were usually able to remember the sequence without aid from the experimenter. Scene rotation was enabled, and participants were allowed to change the viewpoint (accomplished via a drag on the background of the scene). However, participants were encouraged to use a top-down view, similar to Figure 5-3b, as it made the task easier. Virtually all of them changed the viewpoint to this perspective in each trial.

Participants were also encouraged to take breaks between trials, particularly in the vertical device conditions, as these were the least ergonomic and most fatiguing. A counterbalanced ordering also helped ensure that participants did not spend extended periods of time in these conditions. Following the experiment, they were surveyed for subjective preferences as well.

5.1.6 Design

The experiment was a 2×2×2×4 within-subjects design. The independent variables were display orientation (vertical and horizontal) input device movement orientation (vertical and horizontal), support (supported or unsupported) and trial number (1 through 4), respectively. Figure 5-4 depicts all 8 combinations of the independent variables.

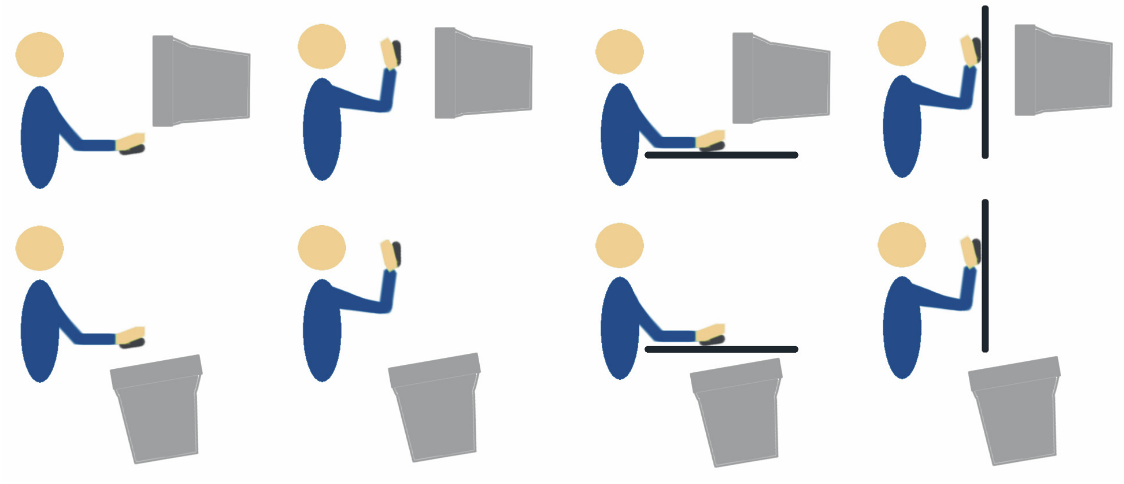


Figure 5-4 : The eight experimental conditions. The left four represent the unsupported conditions, and the right four represent the supported conditions. The top four represent the vertical display, and the bottom four represent the horizontal display.

The orderings of support and device orientation were counterbalanced according to a balanced Latin square to compensate for learning effects across conditions. To reduce the effect of the relatively large time required to switch the display between the top to bottom monitor, half of the participants completed all trials in the vertical display condition first, followed by the horizontal display condition. The other half used the horizontal display first followed by the vertical. Participants performed the task a total of 32 times. Overall, it took approximately 1 hour to complete the series of trials.

5.1.7 Results

The dependent variables were task completion time and accuracy. Accuracy was measured by summing the straight-line distances between object positions at the end of the task compared to the target scene. Mean task completion times and accuracy measures with standard deviations are shown in Figures 5-5 and 5-6.

A repeated measures ANOVA found no significant main effect on completion time for display orientation ($F_{1,511}=0.25$, ns), device movement orientation ($F_{1,511}=0.48$, ns), or hand support ($F_{1,511}=0.05$, ns). A significant effect for trial number ($F_{3,511}=8.07$, $p<.05$) was found, indicating that participants got faster with practice. An interaction between trial number and device orientation fell just short of significance ($F_{3,511}=2.73$, $p=.055$).

Another analysis involved splitting all trials into two groups: one where input device movement orientation and display orientation matched, and one where they did not. There was no significant difference ($F_{1,511}=0.02$, ns). The effect of display orientation ordering was also investigated. Participants who first completed the vertical display and then the horizontal, had a mean completion time of 65.52s and were significantly faster than the 67.24s for participants who did the horizontal display first ($F_{1,511}$=5.06, p<.05). However, if the first trial from each condition is excluded, this difference was not significant ($F_{1,383}$=2.26, p>.05).

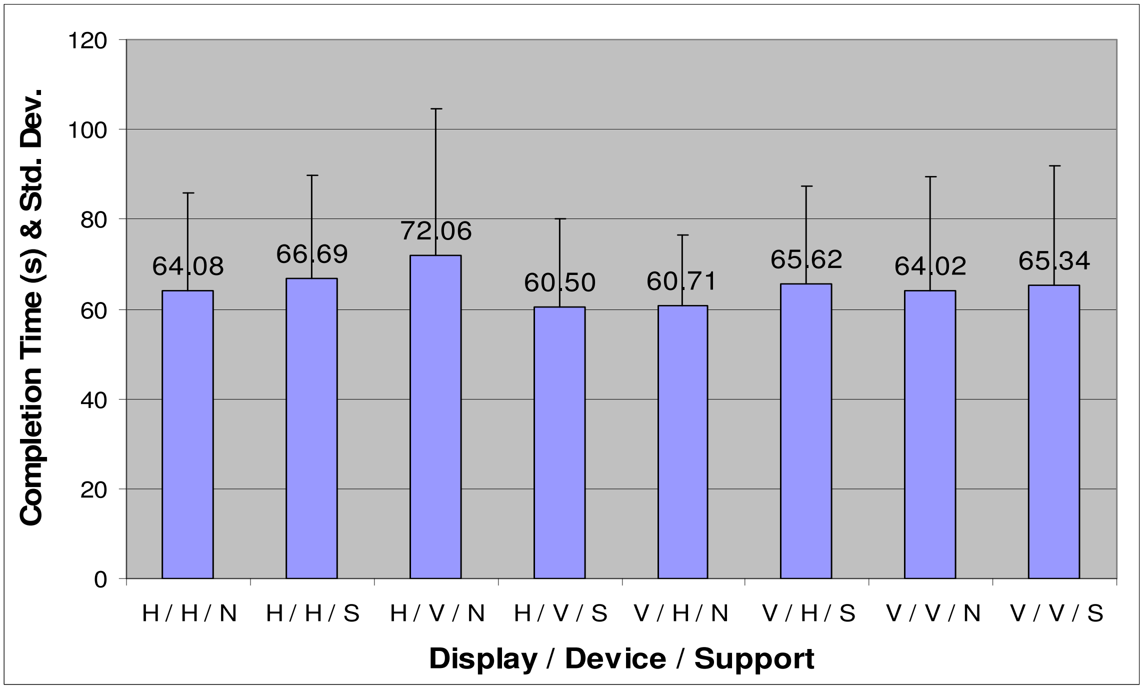


Figure 5-5 : Mean task completion times by condition, with standard deviations.

Due to a software logging error, one accuracy log file was lost. Thus, only 511 such measures were recorded. For accuracy, no significant difference was found in the three conditions: display orientation ($F_{1,510}$=0.95, ns), device orientation

($F_{1,510}$=1.44, p >. 05) and support ($F_{1,510}$=0.17, ns). No significant effect for display ordering was found on accuracy ($F_{1,510}$=0.44, ns).

Fourteen of the sixteen participants replied to the questionnaire. Of these responses, half preferred support, and half did not. The display/device orientation combinations were ranked in order of preference on a scale of 1 to 4, with 1 being most preferred. The ranks for these combinations were analyzed with a Kruskal-Wallis ANOVA and were found to be significantly different (H_3=26.32, p<0.0001). The mean rankings for each combination were 1.42 for the "standard desktop" (vertical display, horizontal device = "VH") configuration, 2.14 for the "HH" condition, 2.86 for the "VV" condition, and 3.57 for the "HV" configuration.

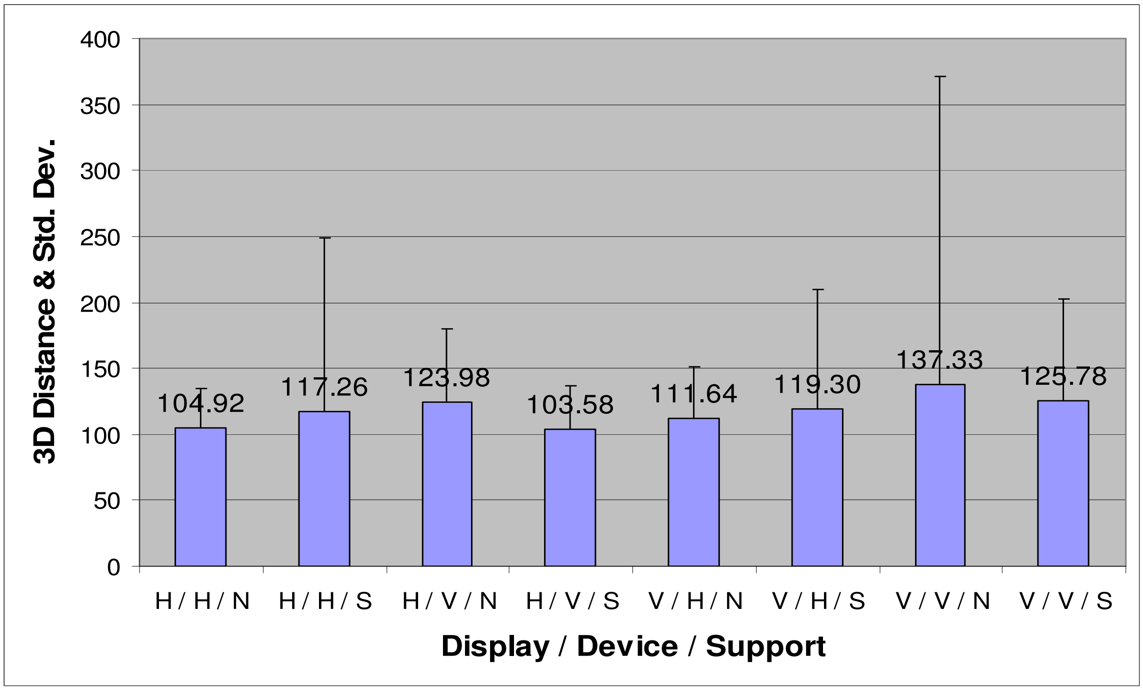


Figure 5-6 : Mean error distance by condition, with standard deviations. Higher values are worse.

These results were again analyzed by demographic information, in particular, prior gaming experience. Participants were grouped into two groups: “gamers” who responded that they played games at least as often as “several times per week” on the survey and “non-gamers” who played games less often than this, for either console games, or computer games using the mouse and keyboard. Analysis of variance on the two groups showed no significant difference in speed between gamers and non-gamers ($F_{1,510}$=0.22, ns). The average task completion time for gamers was 64.24s and 65.26s for non-gamers. Figure 5-7 summarizes these results, and is also broken down by the type of games played by respondents.

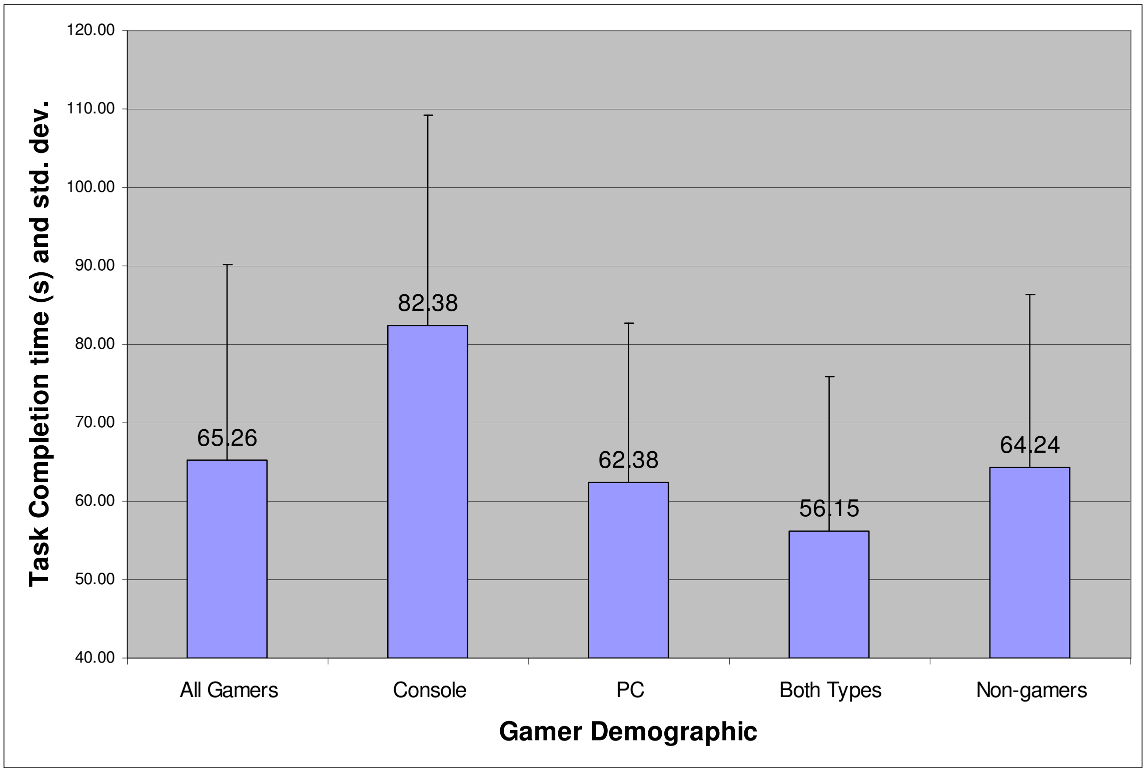


Figure 5-7 : Task completion times by gamer demographic.

The console and PC categories indicate gamers who play just one type of game or the other. The "both" category indicates gamers who play both console and PC games. The "All Gamers" category contains all of these together, for comparison to the "Non-gamers" category.

A similar analysis was performed for accuracy. Eight outlier data points were removed to avoid skewing the data, because they were between eight and ten standard deviations away from the mean. A significant difference was found in accuracy between gamers and non-gamers ($F_{1,501}=6.70$, $p < .01$). Surprisingly, gamers performed *worse* than non-gamers for accuracy, with mean error distances of 111.54 units and 103.15 units respectively. This is depicted in Figure 5-8, again, broken down by specific type of game playing experience.

Analyzing these results by specific type of game experience had been considered. Differences in the input devices used in console games, or PC games may have accounted for some of the variability. However, if this grouping had been done, only two of the participants would be in the "PC gamers" category, and only three would be in the "console gamers" category. Five would be in the "plays both types" category, and the remaining six would be in the non-gamers category. Because these groups are so uneven in size, it would likely have dramatically skewed the result towards significant differences. Thus this analysis was not performed to avoid such biased results. The raw means are shown in the figures for convenience, but the reader is encouraged to keep in mind that these were not analyzed for significant differences.

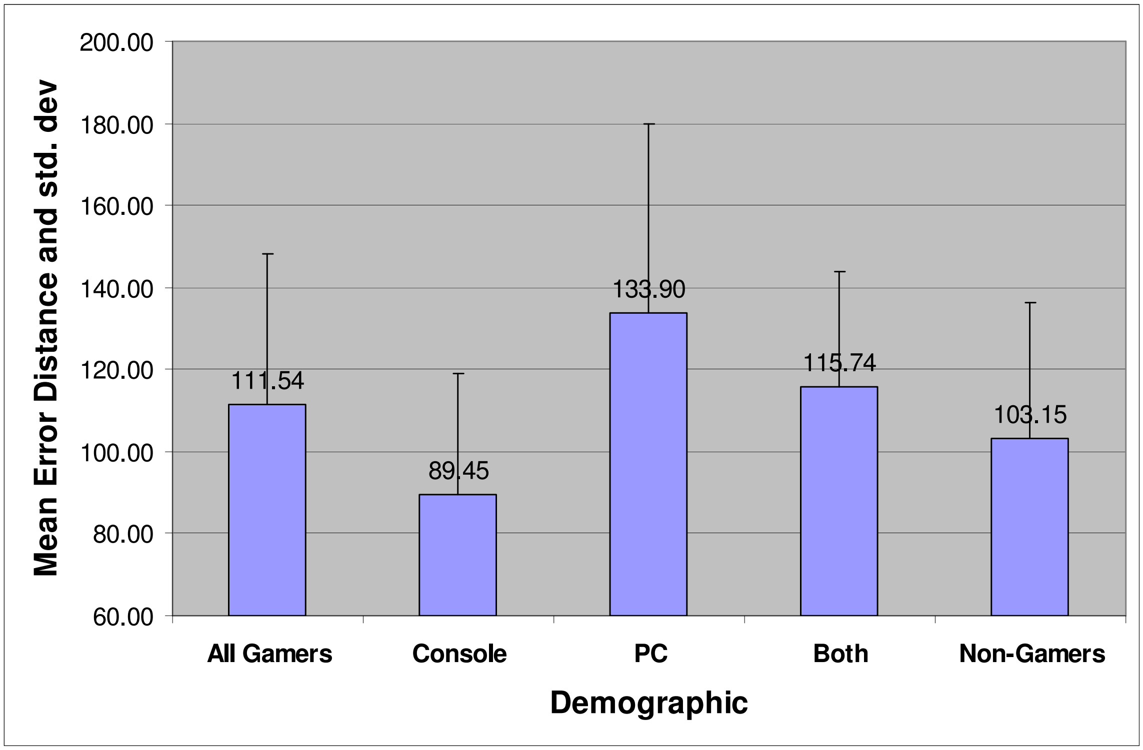


Figure 5-8 : Mean error distance by gamer demographic.

5.1.8 Discussion of Device and Display Orientation

The results of this study are inconclusive and did not determine if input device orientation and display orientation affect performance in constrained 3D movement tasks. Moreover, the statistical power of all tests was fairly low (in the range 0.1–0.2), suggesting that many more participants would be required to reliably detect significant results for the conditions. The maximum difference between similar conditions is also less than 20%, i.e., the magnitude of any potential effect is also limited. Only the nearly significant interaction between trial and device orientation shows that participants were almost significantly better with the horizontal device condition by the fourth repetition compared to vertical. Considering that significant improvements were observed with practice, it seems likely that this interaction effect could become significant with additional repetitions. However, it is not surprising that users might get better faster with the horizontal device; not only is this condition more ergonomic but it is also more familiar due to its similarity to the mouse.

During the experiment, participants were often observed moving the device diagonally in the unsupported conditions. This was perhaps the most natural motion for the unsupported conditions. However, this was impossible in the supported conditions, as the supporting surfaces physically prevented it – device movement was constrained to either the vertical or horizontal 2D plane. This could explain why no significant effect was found for device orientation. However, if motion was diagonal in *all* unsupported conditions, one might expect asymmetric learning to occur: users should get better faster in the unsupported conditions. However, no evidence of this was found. This may suggest that proprioception alone is insufficient for users to accurately move in a single plane of motion in free space. Several participants' comments also support this: they were able to constrain their hand motion to the 2D plane if they watched their hand, but not when relying solely on proprioception (i.e., without looking at their hand).

Since display ordering showed an effect on task completion times, it seems that counterbalancing was not completely successful. However, the effect was quite small (about 2% difference) and disappeared when the first trial from each condition is excluded (i.e., the difference disappears with practice). In addition, nothing is evident in terms of accuracy. Thus this might be caused by the relative unfamiliarity of a horizontal display.

One potential confound in this study was that participants were allowed to freely rotate the scene. However, observations during the experiment show that the scene rotation itself took only about 1–2 seconds – a very small percentage of the overall time. Moreover, virtually every participant rotated to (nearly) the same overhead view in each trial.

Overall, the lack of significant effects prompted the design of the following study, which focused on the support condition. Consequently, all other factors where no significant differences were found were "collapsed" and only the vertical display and the horizontal device movement conditions were used in the second study. This was done to decrease the variability between conditions and to focus on any potentially significant effects.

5.2 Mouse and 3D Tracker Study

The design of this study was prompted by the somewhat surprising results from the previous experiment (see Section 5.1.7, and the discussion in Section 5.1.8). In that study, none of the investigated factors differed significantly from one another. Apparently, physical support alone may not be the reason for the success of the mouse in the preliminary study of Chapter 4. Hence this study was intended to determine what other features of the mouse make it such a good input device for constrained 3D positioning. Consequently, the mouse was again compared directly to the 3D tracker in several conditions, including the 2D movement modes used in the previous study, as well as a full 3DOF movement mode used in the Chapter 4 preliminary study.

5.2.1 Hypothesis

The first hypothesis of this study was that the mouse would outperform the tracker in all conditions. This could indicate that the most plausible explanation for the results of the first study is one of the features that was not investigated in that study. One such feature is tracking resolution – an optical mouse has much higher resolution than a 3DOF tracker, and thus allows for more precise input. Based on the results of the preliminary study, and other research [Oh & Stuerzlinger, 2005], it was also predicted that an unconstrained 3DOF tracker would be slower than all other conditions, including the 2D constrained tracker conditions.

5.2.2 Participants

Ten paid participants took part in the study. Their ages ranged from 19 to 26 years, with a mean age of 22.1 years. Five were male, and five were female. They had been using a mouse for an average of 13.4 years. All used the computer mouse with their right hand.

As before, previous experience with 3D modelling software and video games was recorded on a scale of 1(never use) to 5 (use daily). The average response was 2.0 for 3D modelling software and 3.4 for gaming, suggesting infrequent usage of modelling software and semi-frequent game playing habits, a similar demographic to that in the previous study.

5.2.3 Apparatus

Tasks were performed in the same desktop VR system, using the same displays and stereoscopic system. Only the vertical monitor was used in this study.

This study used an optical mouse as well as the same IS900 tracker used in the previous study. One of the five conditions used the mouse with its speed set to match the tracker as closely as possible and all acceleration/enhancements disabled. All other conditions used the 3D tracker in a variety of modes. Participants wore the tracker and glove in all conditions to mitigate confounding effects due to equipment changes. In addition, they held a mouse with a "top-down" (normal) grip in all conditions. This was to avoid confounds that could arise if, for example, a 3D wand input device was used in the unsupported conditions. Such confounds could occur due to differences in the muscle groups used to perform the motion, since one typically holds a wand-type input device with their hand rotated approximately 90° to how they hold a mouse [Zhai et al., 1996; Balakrishnan & MacKenzie, 1997]. Assuming the muscle groups required to move the mouse *and* the hand tracker were more or less the same, the study design likely rules this confound out.

The table was used to support the mouse and the supported tracker conditions. The tracker again operated as an absolute positioning device. Most of the tracker conditions used the same 15x11.25 cm rectangle to represent the mapping to the screen. However, one condition increased the area to 30x22.5 cm to investigate the effect of an increased relative tracking resolution. This mode provided approximately a one-to-one correspondence between screen size and input area.

The fifth condition used the tracker in full 3DOF positioning mode. Selection was still done via 2D ray casting, but once selected, objects could be freely moved along all three world axes (without sliding). Collision avoidance was still enabled in this mode. Object movement was directly mapped to tracker position: moving the tracker up caused the object to move upwards in the scene; moving the tracker towards the screen caused the object to move "into" the scene, etc. Speed of object motion in this condition was set to be virtually identical to the other conditions (excluding the large area tracker condition).

The software used was also the same fish tank VR version of SESAME used in both previous studies. Stereoscopic graphics were enabled, but head-tracking was not.

5.2.4 Procedure

Participants were first introduced to the experiment and signed consent forms. They were then given a practice trial to familiarize themselves with the task. In addition, they were given verbal feedback throughout the experiment until they were able to remember the task without aid (typically within 2 or 3 trials). The task was the same as in the previous experiment.

Since practically all participants rotated the scene to an overhead view in the first study, this was set as the default viewpoint and scene rotation was disabled in this study.

Following completion of the experiment, participants were surveyed for subjective preferences.

5.2.5 Design

The study was a 5×6 within-subjects design. The first factor was input technique and the second was trial number. Five input techniques were compared: mouse, "mouse emulation", "large area mouse emulation" (30x22.5 cm mapping), "air-mouse emulation" (as mouse emulation but without support), and 3DOF mode. Note that the "mouse emulation" mode was identical to the supported horizontal device condition from the first study. Similarly, the "air-mouse emulation" mode was identical to the unsupported horizontal device condition from the first study.

Participants performed a total of 30 trials each. In total, it took them approximately 1 hour to complete the experiment.

5.2.6 Results

The dependent variables were again task completion time and accuracy. ANOVA showed a significant difference in task completion time between the five conditions, ($F_{4,295}=61.19$, $p<0.0001$). Tukey-Kramer post hoc analysis indicated that the mouse condition was significantly faster than all other conditions. All three of the 2D tracker conditions were not significantly different from one another. Finally, the unconstrained 3DOF tracker condition was significantly slower than all others. The mean times for these conditions are visualized in Figure 5-9.

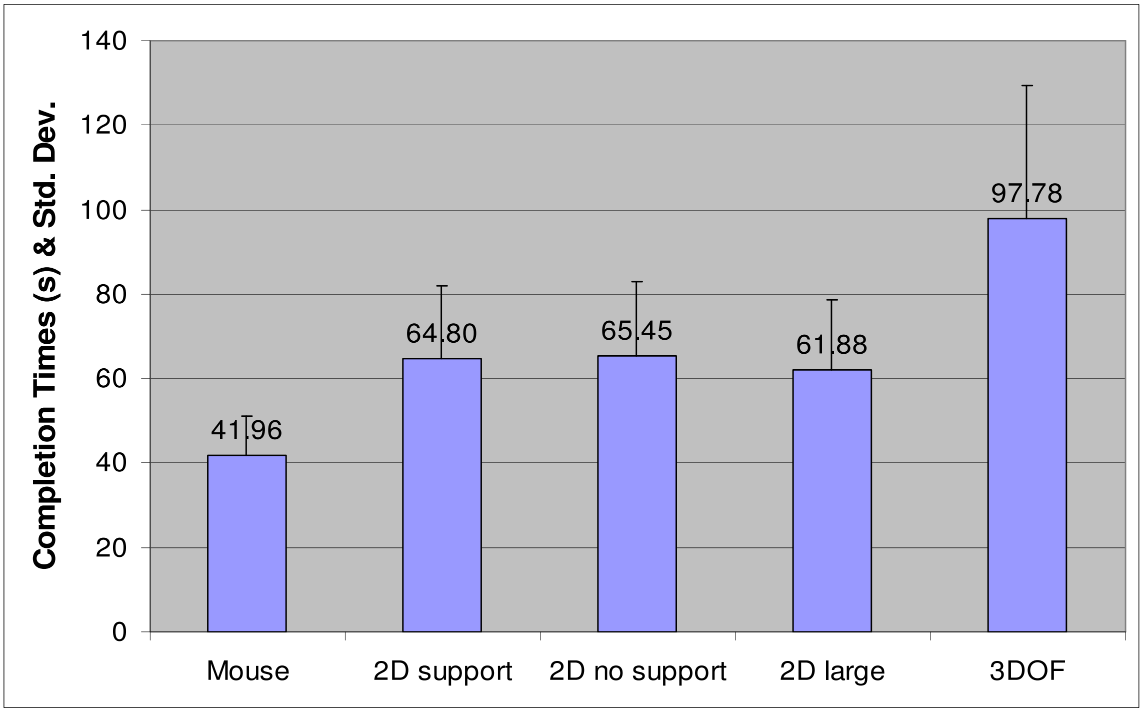


Figure 5-9 : Mean task completion times by condition, with standard deviations.

A significant difference was found in accuracy between the five conditions ($F_{4,290}$=4.65, p<0.005). Tukey-Kramer post-hoc analysis revealed that the mouse and mouse emulation conditions were significantly more accurate than the 3DOF condition. However, no other conditions were significantly different. Figure 7 summarizes the mean error distances for each condition.

This time, participants clearly preferred support, with an average of 1.4 on a 5 point Likert scale (1 being best). Ranks for the 5 movement techniques were analyzed with a Kruskal Wallis ANOVA and were found to be significantly different (H_4 = 12.52, p<.05), with mean preference scores of 1.6 for the mouse, 3 for "mouse emulation", 3.6 for "air mouse emulation", 3.3 for "large area mouse emulation", and 3.5 for the 3DOF tracker condition. Post-hoc analysis revealed that preference for the mouse technique was significantly higher than all other techniques, with the exception of the "mouse emulation" technique. There was no significant difference in preference between the remaining three techniques.

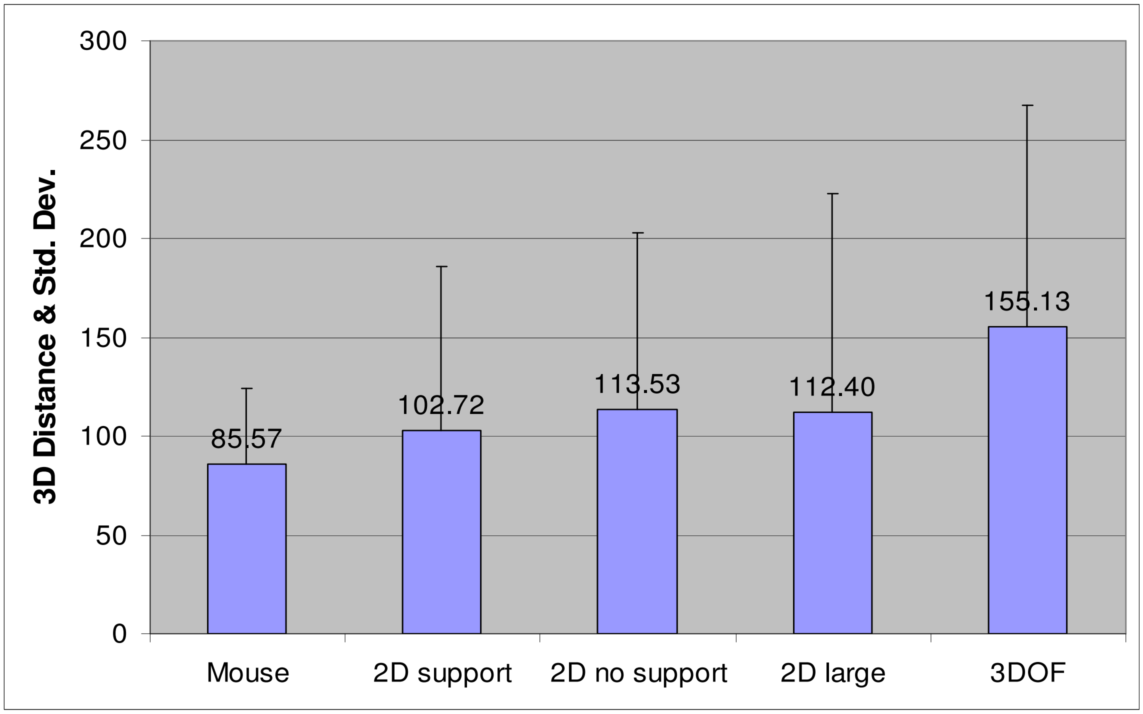


Figure 5-10 : Mean error distance by condition, with standard deviations. Higher values are worse.

An analysis was also performed to determine if participants who regularly played video games performed better than those who did not. If participants responded that they played games "several times a week" or more frequently (a 4 or 5 out of 5 on the response scale) they were labeled as a gamer of the appropriate type. Thus participants were split into four groups depending on their responses to the demographic questionnaire: those that did not play games regularly, those that played console games regularly, those that played PC games using a mouse/keyboard regularly, and those that played both types of games regularly. However, this resulted in only a single participant ending up in each of the "console games only" and "PC games only" groups, and four participants in each of the "non-gamers" and "both types" groups. One-way ANOVA showed a significant difference between these four groups ($F_{3,296}$=4.18, $p < .01$).

Because two of the groups only had a single member in it, these results should be taken with a grain of salt, as individual differences between participants are likely responsible. In other words, these results probably do not say anything about general trends among gamers and non-gamers. Consequently, the participants were reorganized into two groups: those that played any type of game and those that did not. By this method of grouping, there were 4 non-gamers and 6 gamers. A one-way ANOVA on these groups revealed no significant difference in task completion times between ($F_{1,298}$=0.46, ns). However, a significant main effect for accuracy was found ($F_{1,293}$=5.86, $p < .05$); gamers were significantly *less* accurate than non-gamers with mean error distances of 124.27 and 97.69, respectively.

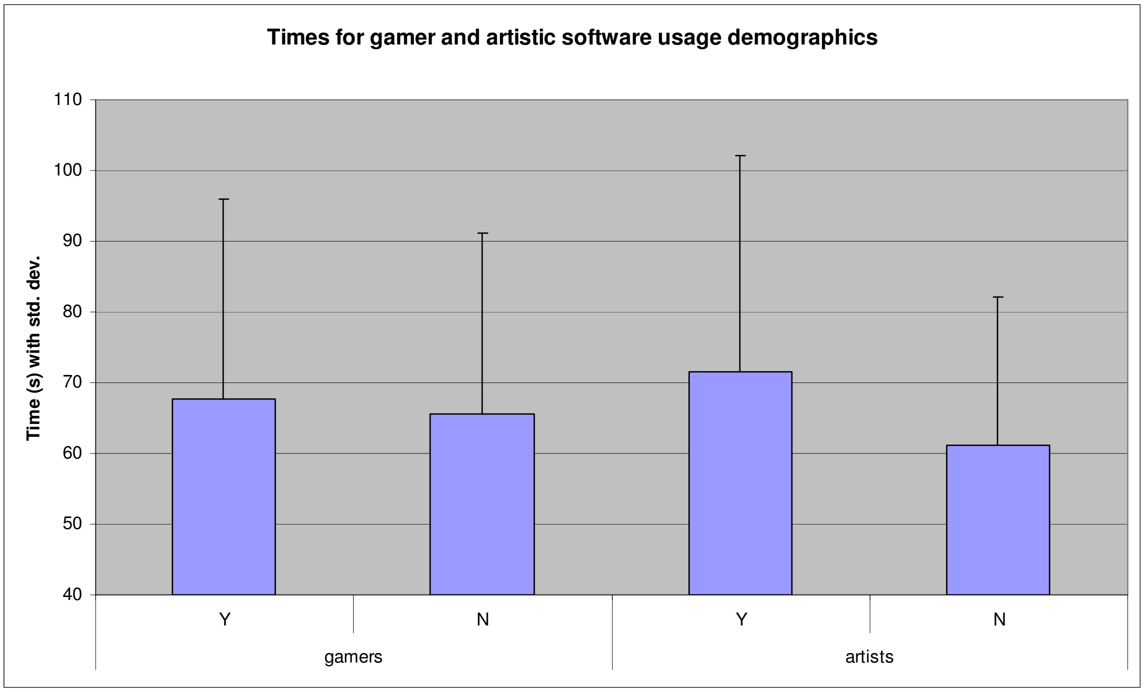


Figure 5-11 : Mean task completion times and standard deviations by gamer and usage of 2D art software demographics

Other demographic information was also examined. Only two of the participants reported using 3D modeling software several times per month, or more frequently, and both of these were also gamers. Consequently, participants' usage of 2D artistic software (e.g. Photoshop) was looked at instead. As with the average game response, the average response for 2D artistic software usage was quite high, at 3.7 on a 5 point scale. As a result, participants were grouped into two groups, frequent users of such software (dubbed "artists"), who reported a 4 or higher and infrequent users, who reported a 3 or lower. This split the participants in two even groups. One-way ANOVA indicated that participants who regularly used 2D artistic software were significantly faster ($F_{1,298}$=11.96, $p < .0001$) than those who did not, with mean times of 61.5s and 71.6s, respectively. No significant difference in accuracy was found ($F_{1,293}$=0.00, ns) between groups based on use of 2D artistic software.

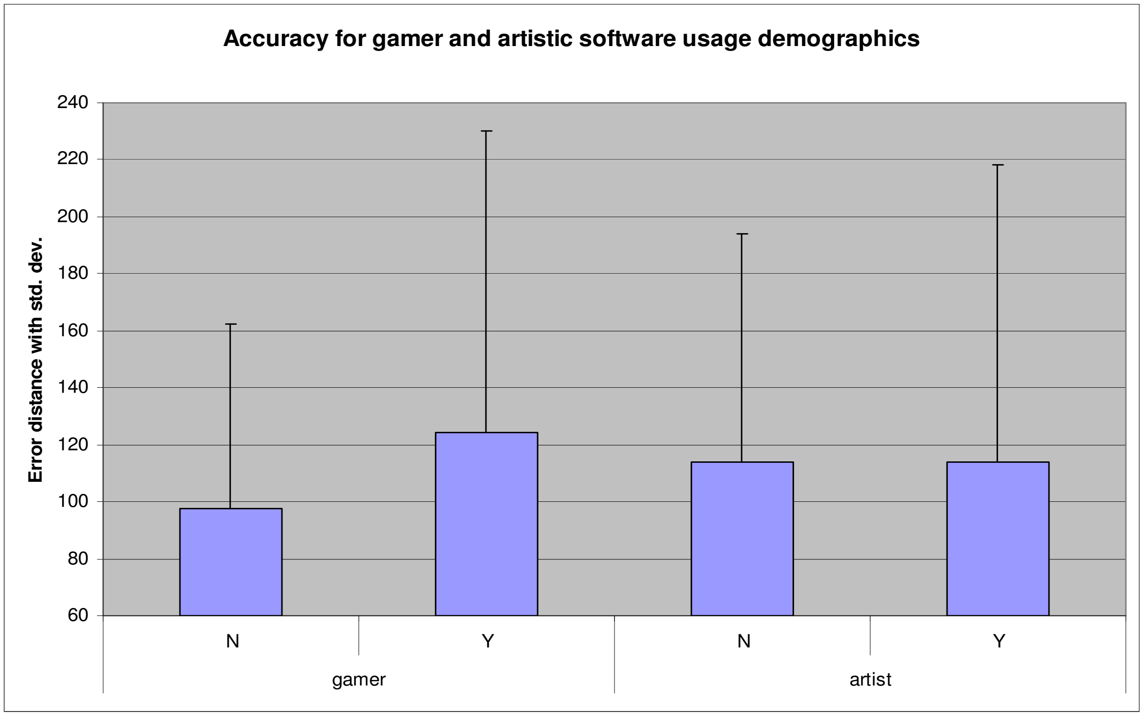


Figure 5-12 : Mean accuracy and standard deviations by gamer and usage of 2D art software demographics.

5.3 Overall Discussion

As discussed above, one concern in the orientation and support study was that allowing scene rotation might have confounded the design. Participants might have been moving objects from different screen locations. To further address this, two conditions that were present in *both* studies were analyzed: “mouse emulation” and “air mouse emulation”. If the viewpoint rotation had confounded the results, this might have been reflected as significant differences between the identical conditions across experiments. However, comparing all trials for these conditions indicates that neither speed ($F_{3,244}=1.03$, $p>.05$) nor accuracy ($F_{3,243}=0.47$, ns) were significantly different. Analyzing only corresponding unsupported conditions and supported conditions also failed to show any significant differences.

As the second study of Chapter 5 had two more repetitions than the first, the additional learning may have resulted in better performance. To account for this, these analyses were repeated on only the first 4 trials. Again, one-way ANOVA showed no significant difference in speed ($F_{3,204}=1.38$, $p>.05$) or accuracy ($F_{3,203}=0.13$, ns). Also, neither the “air-mouse emulation” nor the “mouse emulation” conditions showed any significant differences across experiments. Given that scene rotation time was small compared to the overall times and that no significant differences were found between identical conditions across studies, it seems plausible to conclude that scene rotation probably did not confound the first study of Chapter 5.

Another issue is that the complexity of the task used in both studies increased the variability, thus making it harder to detect significant differences between conditions. As discussed, the task was selected to improve the external validity of the results – perhaps at the cost of internal validity. However, participants were given a “recommended” ordering of object movements during practice, and almost all adhered to it. Additionally, when they showed signs of confusion as to which object to move next, the experimenter would provide verbal instructions according to the recommended ordering. Thus it is likely that these results still address major aspects of the research goals set out earlier.

5.3.1 Physical Support

The lack of effect for support appears to contradict previous findings [Lindeman et al., 1999; Wang & MacKenzie, 2000]. However, one difference is that previous work [Lindeman et al., 1999] used a two-dimensional task: direct manipulation of 2D shapes in a plane. Moreover, unlike other previous work [Wang & MacKenzie, 2000], the input space in the current experiments was disjoint from the display area, which is characteristic of the mouse condition. This was also a feature of the Bat input device, which matches relative movements of the input device to virtual object movement [Ware & Jessome, 1988]. Thus it is likely that the differences in these conditions also caused the differences in results. It is possible that a different input strategy that registers the display with the input device (e.g., a stylus/touch-screen) may benefit more from support compared to unregistered approaches.

Another possible explanation for the lack of differences is that the 2D sliding movement technique used here made the 3D movement task equally difficult (or easy) for all input conditions in the orientation/support study. Thus, the sliding technique may have had much more influence on the results than any of the investigated factors. This is substantiated by the results of the preliminary study of Chapter 4, which also reported "three-tiered" results: tracker conditions using the sliding movement technique were better than the 3DOF technique, with both being outperformed by the mouse. However, it is important to realize that a cross-device comparison with *different* input mapping techniques does not just evaluate the devices – it also evaluates the techniques!

The subjective findings from the first Chapter 5 study suggested that participants were undecided as to the benefits of support. Comments made by participants ranged from “I didn’t like vertical support at all” and “Support felt a bit stubborn” to “Lack of support didn’t seem to affect the results” and “Unsupported conditions were uncomfortable”. However, users clearly preferred support in the second study, as well as combinations of conditions that more closely resemble a desktop environment. Since these conditions performed best, this is more consistent with previous findings about the benefits of support.

5.3.2 Equipment Differences

The extensive familiarity of people with the mouse must be considered. Prior to the using 2D constrained tracker conditions for the first time, participants were warned that although the device felt (physically) like a mouse, it did not behave quite like one: the tracker used absolute positioning, and thus did not require clutching. Participants sometimes tried to clutch to move the cursor more quickly but this had no effect since the device tracked equally well on or off the table. Clutching occurred most often in the large area tracker condition in the second study. This is a potential reason why the large area tracker condition did not perform as well as the mouse emulation, despite the increased relative spatial resolution. However, as the control-display (C-D) ratios for the conditions were the same and input was linear (i.e., no acceleration), one would not expect a difference [MacKenzie & Riddersma, 1994]. Another potential reason is that the differences are due to variations in muscle usage for the larger interaction area, but as the range of motions is not that different, this is also improbable.

The main motivation behind including a large tracking area condition in the second study was a concern about the potential effects of resolution. According to specifications, the IS900 offers 0.75mm resolution, which translates to 200 samples inside a 15cm distance. This was mapped to 800 pixels on the screen. This mismatch in resolution may have degraded performance of the 3D tracker relative to the mouse. In practice, the tracker delivers a bit better precision, so this is a conservative estimate. However, the mouse has a much higher tracking resolution than a 3DOF tracker. Optical mice offer between 400-1800 dpi [Logitech, 2007], which corresponds roughly to 0.05-0.01mm resolution, i.e., between one and two orders of magnitude better than the tracker.

This difference in tracking resolution is arguably the most plausible explanation for the outcome of the mouse and 3D tracker study. The overall familiarity of users with the mouse, the presence/absence of support and differences in how the devices moved are less probable, but cannot be ruled out. Most likely due to the relative unfamiliarity, the unconstrained 3DOF tracker mode showed the strongest learning effects in the first few trials. An ANOVA was performed to determine after which trial participants no longer improved significantly. The last significant improvement in speed occurred between trials 2 and 3 ($F_{1,18}=4.41$, $p<.05$). In other words, starting with the 3rd trial there were no observable learning effects and the learning curves effectively flatten off even for the 3DOF mode. Although it is impossible to predict long-term learning effects from only 6 trials, the evidence suggests that it is unlikely that more training would allow the 3DOF mode to match the other conditions without extensive, long-term training.

5.3.3 Muscle Groups

To avoid confounds, all conditions used the same "top-down" grip on the mouse, with the tracker on the wrist in all conditions. Such confounds could arise if, for example, a 3D wand input device was used in the unsupported conditions. This is because different muscle groups would be used to perform motions, since one typically holds a wand-type input device with the hand rotated ~90° relative to how one holds a mouse. This is also supported by previous work [Balakrishnan & MacKenzie, 1997; Zhai et al., 1996], which showed that using different muscle groups affects performance in 6DOF docking [Zhai et al., 1996] and Fitt's tasks [Balakrishnan & MacKenzie, 1997]. Since our experimental task was made up of several of these simple motions, differences between devices would likely be exaggerated. Consequently, the same device combination was used throughout the experiments to ensure that (approximately) the same muscle groups were used in all conditions, and thus provide a more level playing field.

One participant pointed out that they noticed they moved the mouse with their fingers for fine motions. Since the tracker was mounted on the back of the hand, fine motor control motions, such as adjusting the mouse with the fingertips, were unlikely to have been recorded. This may also account for the differences found.

Chapter 6
Conclusion

This thesis presented research extending a 3D movement technique, designed for use with a mouse, to use with 3D input devices. It also presented three evaluations of this work. These experiments compared the idea of 2D constrained 3D manipulation to standard approaches with the mouse and to full 3DOF movement. Stereoscopic graphics rendering and head-coupled perspective were also examined. The user studies presented in Chapters 4 and 5 suggest that the mouse is not only capable of handling, but is also well-suited to the problem of *constrained* 3D object movement. Consistent with previous findings, stereoscopic graphics were found to have some benefit, and the benefits head-coupled perspective were comparatively small, or statistically insignificant.

Moreover, the studies also demonstrated that 2D to 3D mapping techniques designed for use with the mouse can be successfully adapted for use with 3D input devices, typically resulting in the "3-tiered" results seen in Chapter 4 and in Section 5.2. In these studies, the mouse was the best performer, but the 3D input devices performed better when constrained to 2D operation than it did in 3DOF mode. This was the case even when simple 3D collision avoidance was used. This suggests that the SESAME sliding movement technique may aid 3D positioning more than traditional VR/graphics constraints, at least for certain types of positioning tasks. Ideally, a positioning technique using the 3D device could be developed that could actually outperform the mouse;

however, considering the limitations of current 3D tracking technology, this seems unlikely to happen soon, and indeed, highly dependent on the specific type of task being performed.

The second round of studies also compared various properties of the mouse to the 3D tracking device. These experiments showed that surprisingly, the presence or absence of a supporting surface did not appear to affect the speed or accuracy of 3D movement tasks when using the sliding movement technique. Additionally, no difference was found between matching and mismatching the movement orientation plane of the device to the display. It seems plausible that this is because the movement technique had a much stronger effect on user performance than the other two factors, perhaps even providing a virtual “proxy surface”. However, the mouse was still shown to outperform the 3D device, even in very similar conditions, using the same movement technique. It is likely that some combination of tracking resolution or user familiarity is the cause of these results, but this is a subject that requires further study.

Overall, the purpose of this research has been to determine if immediate usability of VR manipulation techniques by novices can be improved by using movement techniques more similar to the desktop “drag ‘n’ drop” paradigm. The results of the experiments appear to suggest that this may be the case; however, further research is still required. In the meantime, the results suggest that the mouse is certainly a viable input device for fish tank VR systems. Designers of VR systems are encouraged to look at 2DOF alternatives if possible, or at least consider constraining 3/6DOF devices to 2DOF operation. Both of these strategies showed improvements beyond the raw 3D positioning commonly used in many VR systems.

6.1 Future Work

Following are some potential areas for future research on this topic.

6.1.1 Tracking Resolution, Support and Orientation

As the results of the orientation/support study were inconclusive, it would be beneficial to repeat the study and control for some of the sources of variability. The final study also raised issues such as device tracking resolution. A follow-up study may help to determine how important tracking precision really is in these types of tasks, relative to the display and device movement orientation. However, one must account for the different grip and working space, which relates to the muscle groups involved in device usage. A related avenue for future research is further analysis of the differences between muscle groups used to operate various devices. In particular, if accurate finger tracking in free air could be achieved, would this improve performance of the 3D devices to mouse-like levels? One possibility would be to investigate tablets, as these devices provide high precision and are well suited to the sliding 3D movement technique.

One factor that was considered for the orientation/support study, but ultimately not examined was the effect of *scene* orientation compared to display and device orientation. For example, do users perform better in environments where the orientation of the scene matches that of the display (e.g., top-down view, horizontal display versus side-view, horizontal display). However, as the experiment was already becoming too large, this was excluded from the present research, with the intent to revisit it in the future.

6.1.2 Rotations

Due to the additional complexity of 3D rotations, only the 3DOF task of object positioning was examined in this thesis. A natural extension to this work would be to look at the 6DOF general manipulation task. One possibility here would be to use a 3D tracker input device, constrained to 2D, as in the "WandSlide" technique presented earlier. Thus, only two of the three positional DOFs would be used for movement. The additional 3DOFs afforded by the input device could then be used to allow simultaneous object rotation during constrained movement. This raises the question of whether constraining the rotations to 2DOF or 1DOF may also make the task easier (hence a total of 3DOF or 4DOF task) or if novice users are comfortable with full 3DOF rotations while performing 2DOF movement tasks (hence a 5DOF task).

Although there has been some evidence to suggest that expert users *may* simultaneously rotate and move an object [Boritz & Booth, 1998], this is still somewhat unclear. It would be interesting to find a definitive answer to this question, perhaps by performing docking task studies, and comparing the task to sequential positioning and orientation components. Were it determined definitively that users tend to *not* simultaneously move and orient objects, this would be another strong argument in favour of lower-dimensional input devices using rotation-constraining techniques, since they are unlikely to take advantage of the extra degree of freedom anyway.

6.1.3 Beyond Fish Tank VR

The work in this thesis presented extensions of the SESAME system from a desktop-based system to a fish tank VR system. The difference between these environments is primarily the addition of stereoscopic graphics, head-coupled perspective and 3D input devices. It would be very interesting to see how well the same sliding movement technique (and potentially rotation techniques, as discussed above) would extend to more immersive virtual environments such as CAVEs, or head-mounted displays. This presents new challenges with determining how to register the extents of the "screen". In the current studies, the bottom left corner of the screen was registered to the position of the tracking device when the software started up. Clearly, if the user is able to walk around in the environment, this would be unsuitable, as they would quickly leave the tracking square for the device cursor. One approach to solving this might be to make the tracking area relative to the position of another tracker, worn on the head, or waist. This would rely on the user's proprioception sense to keep their hand in the tracking region. Another possibility might be to "re-acquire" the screen extents periodically, if the hand-tracker/wand is left relatively still for a reasonable period of time (e.g., 2 seconds). This way, when the user stops moving, their tracking area "catches up" with them after a temporary delay.

Another interesting possibility would be to extend this movement technique into augmented or mixed reality environments. A first step in this direction would be to allow users of an AR environment to move virtual objects against other virtual objects that are spatially registered with real-world objects. However, an interesting long-term goal would be to allow interaction between virtual and real objects, for example, sliding virtual objects along real-world, physical objects. This might be accomplished by creating invisible virtual replicas of all real objects in a room, registered to the correct position of their real duplicate. This would effectively maintain a database of the positions and approximate geometry of “real” objects in a room that could be used for the purpose of determining movement offsets via the sliding movement algorithm. In a mixed reality environment, the overall effect might appear as though the virtual objects are sliding over real objects in the scene.

6.1.4 Game Experience

The research presented in this thesis suggested that users who play a lot of 3D games also tend to do perform differently than non-gamers in these types of experiments. This is likely due to the similarities between the graphical software used and video games. In some cases, they performed better, but in others, they performed worse (particularly for accuracy). It is unclear if this is the result of training due to games, for example, a learning effect that transcends to similar experiences. Another possibility is that people with inherently superior spatial abilities tend to be attracted to games as well. When one considers that pilots in training may use Microsoft Flight simulator software to log some of their simulator hours, it seems plausible that games may serve a useful purpose in training for other tasks. Due to the limited nature of the survey questionnaires, it is also unclear if the gamers tended to play specific types of games, and if there was any correlation between the type of game played and their performance. If certain types of games could be shown to improve spatial ability, and to train users in 3D computer-driven manipulation tasks, this could be beneficial in training for tasks such as 3D modeling and computer-aided design, or remote robot control, telehaptic surgery, etc. Of course, determining if this is the case would be extremely difficult, and would likely require the use of longitudinal studies on non-gamers.

Bibliography

Balakrishnan, R., and MacKenzie, I. S. (1997). Performance Differences in the Fingers, Wrist, and Forearm in Computer Input Control. *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '99*. 303-310.

Bier, E. (1987). Skitters and Jacks: Interactive 3D Positioning Tools. *Proceedings of Interactive 3D Graphics, I3D '86*. 183-196.

Boritz, J. and Booth, K. (1997). A Study of Interactive 3D Point Location in a Computer Simulated Virtual Environment. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST '97*. 181-187.

Boritz, J. and Booth, K. (1998). A Study of Interactive 6 DOF Docking in a Computerised Virtual Environment. *Proceedings of the Virtual Reality Annual International Symposium.* 139-146.

Bowman, and D. Hodges, L. (1997). An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments. *Proceedings of ACM Symposium on Interactive 3D Graphics '97*. 35-38.

Bowman, D., Johnson, D., and Hodges, L. (1999). Testbed Evaluation of Virtual Environment Interaction Techniques. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST '99*, ACM Press. 26-33.

Chen, J., Narayan, M.A. and Perez-Quinones, M.A. (2005). The Use of Hand-held Devices for Search Tasks in Virtual Environments. *Proceedings of the IEEE Virtual Reality 2005 workshop on New Directions in 3DUI.* 15-18.

Conner, D.B., Snibbe, S., Hemdon, K., Robbins, D., Zeleznik, R. and van Dam, A. (1992). Three Dimensional Widgets. *Proceedings of Symposium on Interactive 3D Graphics, I3D '92.* 183 – 188.

Cruz-Neira, C., Sandin, D., DeFanti, T., Kenyon, R., and Hart, J. (1992). The CAVE: Audio Visual Experience Automatic Virtual Environment. *Communications of the ACM, 35, 6.* 64-72.

FakeSpace Systems. (2007). *Pinch Gloves®*. Available: http://www.fakespacesystems.com/pinch.htm. Accessed December 7, 2007.

Hsu, J., Pizlo, Z., Chelberg, D., Babbs, C. and Delp, E. (1996) Issues in the Design of Studies to Test the Effectiveness of Stereo Imaging. *IEEE Transactions on Systems, Man and Cybernetics, Part A, 26, 6.* 810-819.

Immersion Corporation. (2007). CyberGlove®. Avalable: http://www.immersion.com /3d/products/cyber_glove.php. Accessed December 7, 2007.

Intersense Inc. (2007). *IS-900 Motion Tracking System Technical Overview*. Available: http://www.isense.com/uploadedFiles/Products/White_Papers/IS900_Tech_Overview_Enhanced.pdf. Accessed December 7, 2007.

Jay, C., and Hubbold, R. (2003). Amplifying Head Movements with Head-Mounted Displays. *In Presence: Teleoperators and Virtual Environments, 12, 3*. 268 – 276.

Kitamura, Y., Yee, A. and Kishino, F. (1998). A Sophisticated Manipulation Aid in a Virtual Environment Using Dynamic Constraints Among Object Faces. *In PRESENCE, 7 (5)*. 460-477.

Kohli, L. and Whitton, M. (2005). The Haptic Hand: Providing User Interface Feedback with the Non-Dominant Hand in Virtual Environments. *Proceedings of Graphics Interface.* 1-8.

Konami of America. (2007). *Elebits.* Available: http://www.konami-data.com/officialsites/elebits. Accessed December 7, 2007.

Lindeman, R.W., Sibert, J.L., Hahn, J.K. (1999a). Towards Usable VR: an empirical study of user interfaces for immersive virtual environments. *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '99.* 64-71.

Lindeman, R.W., Sibert, J.L., Hahn, J.K. (1999b). Hand-Held Windows: Towards Effective 2D Interaction in Immersive Virtual Environments. *Proceedings of IEEE Virtual Reality, VR'99.* 205-212.

Lionhead Studios. (2006). *Black and White 2.* Available: http://www.lionhead.com/bw2/. Accessed December 7, 2007.

Logitech. (2007). *Logitech MX518 Gaming-Grade Optical Mouse.* Available: http://www.logitech.com/index.cfm/mice_pointers/mice/devices/187&cl=ca,en. Accessed December 7, 2007.

MacKenzie, I.S. and Riddersma, S. (1994). Effects of output display and control-display gain on human performance in interactive systems. *Behaviour & Information Technology, 13*, 328-337.

Mine, M., Brooks, F. and Sequin, C. (1997). Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction. *Proceedings of the 24th annual conference on computer graphics and interactive techniques*. 19-26.

Mulder, J. and van Liere, R. (2000). Enhancing Fish Tank VR. *Proceedings of IEEE Virtual Reality, VR2000.* 91-98.

Nintendo of America. (2007). *Nintendo Wii*. Available : http://wii.nintendo.com/. Accessed December 7, 2007.

Oh, J.-Y. (2005). Desktop 3D Conceptual Design Systems. *Doctoral dissertation, York University.*

Oh, J.-Y. and Stuerzlinger, W. (2004). A System for Desktop 3D Conceptual Design. *In Virtual Reality, 7.* 198-211.

Oh, J.-Y. and Stuerzlinger, W. (2005). Moving Objects with 2D Input Devices in CAD Systems and Desktop Virtual Environments. *In proceedings of Graphics Interface.* 195-202.

Pagano, R. (2007). Understanding Statistics in the Behavioural Sciences, 8th Edition. Thomsom Wadsworth, Belmont, California, USA. Chapter 17.

Polhemus (2007). *Polhemus corporate website*. Available : www.polhemus.com. Accessed December 7, 2007.

Poupyrev, I., Billinghurst, M., Weghosrt, S., and Ichikawa, T. (1996). The Go-Go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR. *Proceedings of the ACM Symposium on User Interface Software and Technology, UIST '96*. 79-80.

Poupyrev, I., Weghorst, S., Billinghurst M. and T. Ichikawa. (1998a). Egocentric object manipulation in virtual environments: empirical evaluation of interaction techniques. *Proceedings of Eurographics,* 41-52.

Poupyrev, I., Tomokazu, N. and Weghorst, S. (1998b) Virtual Notepad: handwriting in immersive VR. *Proceedings of IEEE VR'98*. 126-132.

Smith, G., Salzman T. and Stuerzlinger, W. (2001). 3D Scene Manipulation with 2D Devices and Constraints. *Proceedings of Graphics Interface.* 135-142.

Strauss, P. and Carey, R. (2002). An Object-Oriented 3D Graphics Toolkit. *Proceedings of SIGGRAPH*. 341-349.

Sutherland, I. (1965). The Ultimate Display. *Proceedings of International Federation of Information Processing IFIP '65, 2*, 506-508.

Szalavári, Z. and Gervautz, M. (1997). The Personal Interaction Panel, A Two-Handed Interface for Augmented Reality. *Proceedings of Eurographics '97.* 335-346.

R. Teather. Constraining VR: An Evaluation of Constraints in 3D Scene Assembly, *Course Project*, *York University, Dept. of Computer Science and Engineering*. May 2006. Available: http://www.cse.yorku.ca/~rteather/constraints.pdf. Accessed March 19, 2008.

Teather, R. and Stuerzlinger, W. (2007). Guidelines for 3D Positioning Techniques. *Proceedings of ACM FuturePlay 2007,* 61-68.

Wang, Y. and MacKenzie, C. L. (2000). The Role of Contextual Haptic and Visual Constraints on Object Manipulation in Virtual Environments. *Proceedings of the SIGCHI conference on Human factors in computing systems*. 532-539.

Ware, C., Arthur, K. and Booth, K. (1993). Fish Tank Virtual Reality. *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '93.* 37-42.

Ware, C. and Jessome, D. R. (1988). Using the Bat: A Six-Dimensional Mouse for Object Placement. *Proceedings of Graphics Interface '88*. 119-124.

Ware, C. and Lowther, K. 1997. Selection Using a One-Eyed Cursor in a Fish Tank VR Environment. *ACM Transactions on Computer-Human Interaction*, 4, 4, 309-322.

Wigdor, D., Shen, C., Forlines, C. and Balakrishnan, R. (2006). Effects of display position and control space orientation on user preference and performance. *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '06*. 309-318.

Zhai, S., Milgram, P., and Buxton, W. (1996). The Influence of Muscle Groups on Performance of Multiple Degree-of-Freedom Input. *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '96.* 308-315.

Zhai, S., Buxton, W. and Milgram, P. (1994). The "Silk Cursor": Investigating Transparency for 3D Target Acquisition. *Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '94*. 459-464.

Zhai, S. and MacKenzie, I. S. (1998). Teaching old mice new tricks: Innovations in computer mouse design. *Proceedings of Ergon-Axia '98 - the First World Congress on Ergonomics for Global Quality and Productivity*. 80-83.