INTERACTION TECHNIQUES FOR COMMON TASKS IN IMMERSIVE VIRTUAL ENVIRONMENTS

DESIGN, EVALUATION, AND APPLICATION

A Thesis

Presented to

The Academic Faculty

by

Douglas A. Bowman

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy in Computer Science

Georgia Institute of Technology

June 1999

INTERACTION TECHNIQUES FOR COMMON TASKS IN IMMERSIVE VIRTUAL ENVIRONMENTS

DESIGN, EVALUATION, AND APPLICATION

Approved:

____
Larry F. Hodges, Chairman

____
Gregory Abowd

____
Albert N. Badre

____
Elizabeth T. Davis

____
Jarek Rossignac

Date Approved: ____________________

ACKNOWLEDGMENTS

The author would like to thank the following people for their help and advice:

- His advisor, Dr. Larry F. Hodges
- The members of the thesis committee: Dr. Gregory Abowd, Dr. Albert Badre, Dr. Elizabeth Davis, and Dr. Jarek Rossignac
- David Koller for his work on the viewpoint motion control experiments
- Dr. Jean Wineman, Brian Wills, and Don Allison for their work on the VR Gorilla Exhibit
- Donald Johnson for his work on the testbed evaluations and the final version of the Virtual Habitat
- Mark Mine, Richard Stoakley, Jeff Pierce, Andrew Forsberg, Ken Hinckley, Matthew Conway, Barry Peterson, Ivan Poupyrev, Ernst Kruijff, Joseph LaViola, and Ben Watson for various collaborations and discussions
- The members of the 3D UI mailing list
- Drew Kessler for help with the SVE toolkit
- The Virtual Environments group at Georgia Tech
- The numerous experimental subjects who volunteered their time
- Dawn Bowman

TABLE OF CONTENTS

Introduction 1

1.1 Motivation 1
1.2 Definitions 4
1.3 Problem Statement 6
1.4 Scope of the Research 7
1.5 Hypotheses 8
1.6 Contributions 9
1.7 Summary of This Work 9
1.8 Summary of Recommendations 10
  1.8.1 Generic VE Interaction Guidelines 10
  1.8.2 Guidelines for the Design of Travel Techniques 10
  1.8.3 Guidelines for the Design of Selection Techniques 10
  1.8.4 Guidelines for the Design of Manipulation Techniques 10

Interaction in Virtual Environments 12

2.1 Human-Computer Interaction Concepts 12
2.2 Three-Dimensional User Interfaces 14
2.3 Perceptual and Cognitive Psychology Concepts 15
2.4 Evaluation of Immersive Virtual Environments 16

Design and Evaluation Concepts 18

3.1 Taxonomy and Categorization 18
3.2 Guided Design 20
3.3 Performance Measures 21
3.4 Range of Evaluation Methods 21
3.5 Testbed Evaluation 22
3.6 Models of Human Performance 23
3.7 Application of Results 23
3.8 Summary of Methodology 24

Travel 26

4.1 Introduction and Definitions 26
4.2 Related Work 26
4.3 Original Evaluation Framework 27
  4.3.1 Categorization of Techniques 27
  4.3.2 Performance Measures 29
4.4 Initial Experiments 30
  4.4.1 Spatial Awareness Experiment 31
  4.4.2 Absolute Motion Experiment 32
  4.4.3 Relative Motion Experiment 32
4.5 Expanded Evaluation Framework 33
  4.5.1 Task Characteristics 34

4.5.2 Environment Characteristics.... 34
4.5.3 User Characteristics.... 35
4.5.4 System Characteristics.... 36
4.5.5 Information Gathering Experiment.... 36
4.5.5.1 Method.... 37
4.5.5.2 Results.... 41
4.5.5.3 Discussion.... 42
4.6 Alternate Evaluation Framework.... 43
4.6.1 Taxonomy.... 43
4.6.2 Guided Design.... 44
4.6.3 Spatial Orientation Experiment.... 46
4.6.3.1 Method.... 48
4.6.3.2 Results.... 50
4.6.3.3 Discussion.... 54
4.7 Travel Testbed.... 56
4.7.1 Method.... 56
4.7.2 Results.... 61
4.8 Summary.... 66

Selection and Manipulation.... 67

5.1 Introduction and Definitions.... 67
5.2 Related Work.... 68
5.2.1 Interaction Metaphors.... 68
5.2.2 Evaluation of Techniques.... 69
5.3 Initial Evaluation and Design.... 70
5.3.1 Techniques Considered.... 70
5.3.2 User Study.... 72
5.3.3 HOMER Technique.... 73
5.4 Formal Evaluation Framework.... 75
5.4.1 Categorization of Techniques.... 75
5.4.2 Performance Measures.... 77
5.4.3 Outside Factors.... 78
5.4.3.1 Task Characteristics.... 78
5.4.3.2 Environment Characteristics.... 78
5.4.3.3 User Characteristics.... 79
5.4.3.4 System Characteristics.... 79
5.4.4 Guided Design.... 79
5.5 Selection/Manipulation Testbed.... 80
5.5.1 Method.... 81
5.5.2 Results.... 82
5.6 Summary.... 87

Interaction in a real-world VE application.... 88

6.1 Integrating Techniques into an Application: Issues and Challenges.... 88
6.2 The Virtual Habitat.... 89
6.2.1 Original VR Gorilla Application.... 89
6.2.2 Application to Environmental Design Education.... 90
6.2.3 Interaction Requirements.... 91
6.3 Interaction Design.... 92

6.3.1 Naïve Interaction Design. 93
6.3.2 Intermediate Design Iteration 95
6.3.3 Final Interaction Design. 100
6.4 Final Usability Evaluation 101

Conclusions and Future Work 104

7.1 VE Interaction Guidelines 104
7.1.1 Generic VE Interaction Guidelines 105
7.1.2 Guidelines for the Design of Travel Techniques 105
7.1.3 Guidelines for the Design of Selection Techniques. 107
7.1.4 Guidelines for the Design of Manipulation Techniques 107
7.2 Formal Design & Evaluation Frameworks 108
7.3 Focus on Applications and Usability 109
7.4 Future Work. 109
7.4.1 Automatic Interaction Design and Performance Modeling 109
7.4.2 Cross-task Interaction Techniques. 110
7.4.3 Comparison with Usability Engineering 111
7.4.4 Interaction in Other Display Modalities 112

Appendix A: Standard User Questionnaire 113

Appendix B: Comfort Ratings Form 115

Appendix C: Complete results of the travel testbed experiment 116

Appendix D: Complete Results of the Selection/Manipulation Testbed Experiment 118

References 123

Vita 132

LIST OF TABLES

Table 4.1 Average Values of Overall Score for Each Tested Treatment Combination in the Information Gathering Experiment; Higher Scores are Better........................ 41
Table 4.2 Number of Subjects Observed Using Common Strategies for Each Travel Technique.. 53
Table 4.3 Mean Completion Times (seconds) for Naïve Search Task (Standard Deviation in Parentheses)... 62
Table 4.4 Mean Completion Times (seconds) for Primed Search Task, with Targets not Within View from Start Location (Invisible) or In View from Start Location (Visible).. 64
(*normalized times – seconds per 100 meters).. 64
Table 5.1 Mean Manipulation Time Results by Technique from Testbed Evaluation....... 84
(* The linear mapping used in these cases was a one-to-one physical to virtual hand mapping).. 84
Table 5.2 Interaction Between Required Accuracy and Degrees of Freedom for Manipulation Time (seconds)... 85
Table 6.1 Mean Usability Ratings for the Intermediate Virtual Habitat Interaction Design 99
Table 6.2 Mean Usability Ratings (standard deviations in parentheses) for the Final Virtual Habitat Interaction Design 102
(* features changed since the previous iteration)..102
Table C.1 Results of Primed Search Task 116
Table C.2 Demographic and Comfort Rating Summary 117
Table D.1 Speed Results for Selection Task..118
Table D.2 Speed Results for Manipulation Task...119
Table D.3 Demographic and Comfort Rating Summary 122

LIST OF ILLUSTRATIONS

Figure 2.1 General Taxonomy Format 19
Figure 2.2 Flowchart of Design and Evaluation Methodology 25
Figure 4.1 Taxonomy of Travel Techniques for Immersive Virtual Environments 28
Figure 4.2 Environment for the Spatial Awareness Experiment 31
Figure 4.3 The Relative Motion Experiment Environment 33
Figure 4.4 Outside View of a Three-Dimensional Corridor 38
Figure 4.5 Interior of a Corridor from the Information Gathering Experiment 39
Figure 4.6 Example Completed Corridor Map with Four Word/Location Pairs, One Unpaired Location, and Two Unpaired Words 40
Figure 4.7 Alternate Taxonomy for Travel Techniques with Detail on Position Indication Subtask 44
Figure 4.8 Route-planning Technique Using Virtual Map and Stylus 45
Figure 4.9 Inside View of a Corridor with a Target Object 46
Figure 4.10 Views of Four Corridor Types used in the Spatial Orientation Experiment: Top left: 2D, right angles, top right: 2D, non-right angles, bottom left: 3D, right angles, bottom right: 3D, non-right angles 48
Figure 4.11 Mean Error in Various Treatment Combinations for Analysis 1 (left) and Analysis 2 (right) 51
Figure 4.12 Mean Time in Various Treatment Combinations for Analysis 1 (left), and Analysis 2 (right) 52
Figure 4.13 Example Obstacles from the Travel Testbed Experimental Environment 57
Figure 4.14 Target Object from the Travel Testbed Experimental Environment Including Flag and Required Accuracy Radius 58
Figure 4.15 Virtual (top) and Physical (bottom) Views of the Map Dragging Travel Technique 60
Figure 4.16 Interaction Between Task Type and Technique for Think Time on Search Tasks 65
Figure 5.1 Mapping Function for the Go-Go Technique: R_r=Physical Hand Distance, R_v=Virtual Hand Distance. Reproduced from (Poupyrev et al, 1996) 71
Figure 5.2 Stretch Go-Go Technique, with Gauge 71
Figure 5.3 Time Sequence of the HOMER Technique 74
Figure 5.4 Taxonomy of Selection/Manipulation Techniques 76
Figure 5.5 Example Trial Setup in the Selection/Manipulation Testbed 81
Figure 5.6 Interaction of Selection Technique with Object Distance for Selection Time Measure 83
Figure 5.7 Interaction of Selection Technique with Object Size for Selection Time Measure 83
Figure 5.8 Logarithmic Scale Graph of Interaction Between DOFs and Accuracy 86
Figure 6.1 The Virtual Reality Gorilla Exhibit 89
Figure 6.2 Embedded Audio and Text Information in the Virtual Habitat 91

Figure 6.3 Virtual Menus in the CDS System.. 94
Figure 6.4 Constrained Object Manipulation in CDS with Ray-Casting..................... 95
Figure 6.5 Physical Devices used in the Virtual Habitat Application......................... 96
Figure 6.6 User's View of the Interface Tablet in the Virtual Habitat........................ 97
Figure 7.1 Example Taxonomy and Technique Components: If Performance Results for Techniques A, B, and C are Known, the Performance of Technique D can be Inferred...110
Figure 7.2 Simplified Taxonomies Linked Together by Cross-Task Techniques..........111

SUMMARY

Human-Computer Interaction (HCI)* in three dimensions is not well understood, and there are few 3D applications in common use. Moreover, the complications of 3D interaction are magnified in immersive virtual environment (VE) applications: characteristics such as inaccurate tracking and lack of access to traditional input devices cause the design of user interfaces (UIs) and interaction techniques (ITs) for immersive VEs to be extremely difficult. Despite these difficulties, we maintain that there are complex applications for which immersive VEs are desirable, so special attention needs to be paid to the design and implementation of ITs for these applications.

A large percentage of interactions that take place in immersive VEs fall into a small number of general categories, which include travel (movement of the user's viewpoint from place to place), selection (indicating virtual objects within the environment), and manipulation (setting the position and/or orientation of virtual objects). Given techniques with good performance characteristics for these three interactions, a large number of complex and effective VE applications could be built. In this research we studied ITs for these three universal tasks in the context of a formal, systematic framework, including the design of novel ITs and empirical, comparative evaluations of techniques.

This thesis presents several important results of the use of this methodology. First, we have developed new ITs perform well in a variety of application scenarios. Second, we have designed general testbeds for IT evaluation that may be reused for future performance comparisons. Third, we have obtained a large set of empirical results regarding the performance of ITs. These results led to general principles and guidelines (section 7.1) that can be applied to VE systems to improve performance. Finally, we validated these results by applying them to a real-world VE application, and showing that its usability was measurably improved as a direct result. The results presented in this thesis should be useful and important to anyone developing a VE system with even a moderate amount of interaction complexity.

* For precise definitions of this and other key terms, see section 1.2.

CHAPTER I

INTRODUCTION

1.1 Motivation

Immersive virtual environments (VEs) made their debut in the late 1960s when Ivan Sutherland created the first system involving a tracked head-mounted display (HMD) and real-time three-dimensional computer graphics (Sutherland, 1968). The system was crude, and the amount of computing and rendering power was minuscule, compared to today's technology, but all of the basic components that make up the virtual reality (VR) systems of the 1990s were present in Sutherland's prototype.

Since that time, there have been over thirty years of continuous research in the area of virtual environments. New hardware technology is continuously in development that allows us to render more complex 3D scenes at interactive frame rates. Graphics displays have seen tremendous improvement: we are able to display millions of different colors simultaneously on a very large screen at a refresh rate so fast that the human eye cannot perceive the flicker (Foley et al, 1990). There are many different tracking technologies available which provide 3D position and orientation data for multiple receivers simultaneously (Meyer and Applewhite, 1992). Technologies are being developed which provide input to other human sensory modalities besides vision. Haptic devices allow a VE user to seemingly "touch" virtual objects (Gomez, Burdea, and Langrana, 1995). Spatial sound creates the illusion of audio sources coming from certain locations in the 3D space (Durlach, 1991). There is even research into the use of olfactory input in virtual environments (Dinh et al, 1999).

VE research has not focused entirely on hardware; software advances have also been made. Algorithms have been implemented and refined in the areas of model simplification, level of detail culling, geometry database management, texture mapping, lighting and shading, hidden surface elimination, and so on. All of these algorithms allow us to present a more complex and more realistic environment, while still maintaining real-time frame rates. Also, large software systems have been created expressly for the purpose of aiding the development of virtual environment applications (e.g. Kessler et al, 1998). These VE support systems can handle rendering, model maintenance, lighting, interfaces with trackers and other input devices, etc. This allows the developer to focus on the components which distinguish his VE application from others: the environment itself and the behavior of the application (e.g. response to button presses, animation, and interaction with virtual objects).

What does the virtual environment community (primarily university researchers, small commercial ventures, and hobbyists) have to show for these thirty years of advancement in hardware and software specifically targeted at immersive VEs? Certainly, the degree of realism and complexity has increased, and making the virtual world more believable in this way may lead to a higher sense of immersion, or presence, for the user. But what applications have emerged into more common use outside of the laboratory? Surprisingly, our experience in the field indicates that there are very few VE applications in common use. To understand why, we should examine those applications that have become useful, and determine their common characteristics that allowed their success. Three such applications are architectural walkthrough, psychotherapy, and VE gaming (we discuss flight simulation and training, two other applications used for real work, below).

Architectural walkthrough (Brooks, 1992) was perhaps the application which brought VE technology into the public eye more than any other. The basic idea is simple: the user can be immersed within a 3D model of an architectural space, and view it and move about it from a first-person perspective, as she would in an actual building. In this way, architects can verify the appropriateness and visual impact of their designs, engineers can study physical aspects of the space, and prospective clients can assess the current status of the project and suggest changes before a structure is even built. Why are VEs needed for this task, rather than simply viewing 3D models on a computer screen? One possible reason is that the user is immersed within the model, and can use her proprioceptive and kinesthetic senses to evaluate the space in a natural manner. Furthermore, this application requires only one additional component over those first proposed by Sutherland: some method of moving the user's viewpoint about the space.

Applications in the field of psychotherapy (Hodges et al, 1995, North, North, and Coble, 1996) have emerged rapidly since the early 1990s. One of the most well-known areas, which is beginning to see practical usage, is the treatment of various phobias. A common method of therapy for phobias is called graded exposure. The patient is placed in a situation in which the fear is triggered, but only slightly. He remains there with the therapist until he has mastered his fear in that situation, at which point a slightly more intense situation is presented. In this way, the patient gradually becomes able to deal with his fear. For example, to treat acrophobia, the fear of heights, the patient might be taken to a second floor balcony, then a fifth floor balcony, then the roof of a ten story building. This treatment has been shown to be effective, but also time-consuming, potentially embarrassing for the patient, and sometimes costly. The only requirement for exposure therapy is that the patient feel present in a situation which triggers his fear, which makes this application a natural one to try in a VE. The treatment can now take place in the therapist's office, without the time, embarrassment, or cost associated with traditional exposure therapy. Unlike architectural walkthrough, VE exposure therapy does not even require a means for the user to move about. It is usually sufficient for the user to be able to sense the environment and to look around (using head tracking), so that the fear stimulus can be perceived.

VE entertainment and game applications have also become popular in recent years. This has most often taken the form of location-based entertainment (LBE) through companies such as Virtuality™, which involves a complete VE system installed in some permanent location, with users paying for each game. In any case, most of the games available for such systems can be characterized as first-person "shoot-em-up" games, in which the user moves through the virtual environment shooting his enemies. In many ways, the requirements of these games are similar to those for architectural walkthrough: real-time 3D graphics, head tracking, and some technique for moving through the environment. The only additional requirement is some sort of weapon that can be aimed and fired at the enemies in the game.

What do these applications have in common? It seems that they all benefit from the enhanced sense of presence that an immersive virtual environment provides. "Being there" is what makes these systems more compelling or useful than the same 3D graphics rendered on a screen, with no head tracking. However, we also claim that each of these applications requires very little in terms of user interactivity. In applications such as exposure therapy, the user is mostly passive, simply looking around the space using standard head tracking. In the walkthrough and entertainment applications, the user may be more active (moving through the space, shooting, etc.), but the actions are very simple and repetitive. We would call this a high frequency but a low complexity of interaction.

There are, however, a small number of applications being used for real work which have more complex characteristics of interaction. These include flight and vehicle simulation, which has been in use for many years, and training applications such as those used by NASA for simulation of astronaut "space walks." Although these applications are more complex, the interaction is designed in a manner very specific to the system, and not in a way that could be extended to other types of applications. As Fred Brooks pointed out in his 1999 keynote address to the IEEE Virtual Reality conference, this is most often done by replicating the devices that the user would interact with in the real-world situation (e.g. the throttle and flight stick, or the spacesuit controls) and using those to drive the simulation. Because of this specificity to the application domain, we claim that there is little that we can learn in general about VE interaction from such systems.

On the other hand, many more application areas have been proposed and researched for immersive VEs. The architectural community wants to take the walkthrough to the next step and be able to not only view, but also design artifacts in a VE (Bowman, 1996, Mine, 1997). Prototype scientific visualization applications have been developed (Bryson and Levit, 1992, Taylor et al, 1993), in which scientists can interactively view complex simulations and structures, and also change the parameters of the simulation, move and regroup elements, and so on. Educational applications have been proposed (Dede, Salzman, and Loftin, 1996) that allow students to learn about certain concepts by engaging themselves in a virtual laboratory, and viewing the effects of changes first hand. The list goes on.

However, we have not seen these applications in common use. It is our opinion that this is not because they are inappropriate for immersive VEs, but because their requirements for interaction are much more complex than the applications discussed previously. These systems require not only head tracking and a method of movement, but also the ability to select objects, to pick up, position, orient, and place objects, to change the system mode, to control the speed of a simulation, etc. One could argue that these applications are not in the mainstream due to the limitations of technology (input devices, trackers, displays, etc.), but researchers have been attacking the technology problem for thirty years. Our claim, on the other hand, is that because little research has been devoted to the *user interface* and *interaction techniques* for immersive VEs, the resulting prototype applications are not as usable as they need to be, and therefore do not see real-world usage. We must ask the question, "Given the current state of VE technology, is it possible for a virtual environment system to simultaneously be immersive, have complex interaction, and exhibit high levels of usability?"

Why is it difficult to develop appropriate user interfaces and interaction techniques for immersive virtual environments? Shouldn't interaction in VEs be completely natural, replicating the real world? Some have argued that this should be the case (Nielsen, 1993). Considering the applications we wish to develop for VEs, however, such natural interaction would be woefully inadequate. Instead, we want to extend the user's physical, perceptual, and cognitive capabilities so that real work can be performed in a VE that could not be done easily in another setting. Therefore, we need new techniques for interaction.

Why is the current state of the art not good enough? Interaction research (human factors, human-computer interaction, user interfaces, etc.) is almost as old as the first computers. Many usable applications have been developed for the traditional desktop metaphor which have extremely complex interaction requirements. However, interaction in immersive VEs faces many difficulties that make it not only harder to develop, but also fundamentally different, than traditional user interfaces.

Desktop interfaces are inherently more constrained than immersive interfaces. Most desktop applications use only two dimensions, which map directly to the 2D control of a mouse. The mouse rests on a surface, so it does not have to be held continuously by the user, and this allows the user to position it very accurately. Text entry is simple and standardized with a keyboard. On the other hand, input devices for immersive VEs are generally three-dimensional, and must be held in place continuously, resulting in lower accuracy. Tracking devices also have inaccuracies, as well as latency which causes the displayed image to lag behind the actual tracker positions. Text entry is generally extremely difficult or impossible, because the user cannot use a standard keyboard while wearing an HMD and/or holding other input devices. Besides these problems, most common HMDs have lower resolution than monitors, so that screen space is even more valuable.

All of these difficulties combine to make usable immersive interfaces much more problematic to design than their desktop counterparts. This is not to say that all previous user interface research is invalid for immersive VEs. On the contrary, certain high-level guidelines and concepts (e.g. Norman, 1990) apply perhaps even more to VEs than traditional systems, because the user interface must be even more transparent and intuitive in order to overcome the other limitations. However, because of the fundamental differences between traditional and immersive interfaces, new research is required that focuses solely on interaction in immersive VEs. Indeed, in his 1999 IEEE Virtual Reality keynote address, Dr. Fred Brooks stated that finding the best ways to interact with virtual environments was one of the five most important open questions in the field.

In this work, therefore, we are taking initial steps in a research program to develop an understanding of interaction techniques and user interfaces for immersive virtual environments. The goal will be both a qualitative understanding, as in user interface guidelines, as well as a quantitative model of performance and usability. Our contribution will be to evaluate and analyze the most common interactive tasks required by VE applications, as well as to categorize and evaluate various interaction techniques designed for these tasks. We will show the effectiveness of our evaluation by applying the results to an application designed for real-world usage.

1.2 Definitions

Before beginning our discussion of interaction techniques for virtual environments, it is important that we define each of the major terms that relate to this work, so that the boundaries and components of the problem are well understood. Some of these terms have disputed definitions, and we do not claim to offer the final word on these terms. We simply intend to provide definitions that allow the reader to understand the use of these terms in this thesis. The terms that we define here are relevant to virtual environments, user interfaces and interaction, and important technologies used in VEs.

- **Virtual Environment (VE)**: A three-dimensional model of a space displayed to a human user from an egocentric point of view using real-time 3D computer graphics. A single object model, viewed from the outside in, is not a VE by our definition. Motion and point of view orientation are generally controlled by the user, not the system. Thus, a first-person computer animation also does not qualify as a VE. VEs often include other sensory information, such as auditory or haptic cues.

- **Virtual Reality (VR)**: The experience of being within a VE. We prefer not to use this term, as it is associated with unrealistic hype and expectations portrayed in popular media.
- **Real-Time**: Displayed at a frame rate that ensures that images move smoothly as the view direction changes. The minimum frame rate that is considered to be real-time might be as low as 10 Hz, or as high as 30 Hz.
- **Immersion**: The feeling of "being there" that is experienced in some VEs. A VE user is immersed when he feels that the virtual world surrounds him and has to some degree replaced the physical world as the frame of reference. Immersion may take place in other media, such as films or even books.
- **Presence**: A synonym for immersion.
- **Immersive**: Surrounding the user in space. A VE is described as immersive when the computer-generated environment appears to enclose the user, and when the parts of the physical world that are not integral system components are blocked from view. In a head-mounted display (HMD), the graphics always appear on screens coupled to the user's head, but this produces the illusion that the VE surrounds the user completely. In a flight simulator, the graphics appear out the window, and are updated as the plane "turns" so that the VE seems to surround the user. The physical cockpit of the simulator is not blocked from view, but it is part of the simulation. For HMD or stereoscopic spatially immersive display (SID) systems, head tracking is required to make the system immersive.
- **Human-Computer Interaction (HCI)**: The exchange of information between human beings and computers during a task sequence for the purpose of controlling the computer (from the point of view of the human) or informing the user (from the point of view of the computer). This interaction usually has the goal of increasing human productivity, satisfaction, or ability (Hix & Hartson, 1993).
- **User Interface (UI)**: The hardware and software that mediate the interaction between humans and computers. The UI includes input and output devices, such as mice, keyboards, monitors, and speakers, as well as software entities such as menus, windows, toolbars, etc (Hix & Hartson, 1993).
- **Interaction Technique (IT)**: A method by which the user performs a task on a computer via the user interface. An IT may be as simple as clicking the mouse button, or as complex as a series of gestures. There may be many possible ITs for any given interaction task. The IT may be influenced by the input device used, but is separate from it. The same input device may be used for many ITs for the same task; conversely, it may be possible to implement a given IT using several different input devices.
- **Head-Mounted Display (HMD)**: A computer graphics display that is worn on the head of the user, so that the displayed graphics are continuously in front of the eyes of the user. HMDs may use Cathode Ray Tube (CRT) or Liquid Crystal Display (LCD) technology, and usually incorporate optical lenses to widen the

displayed image and move it farther from the user's eyes. Many HMDs include headphones for audio, and most are used in conjunction with trackers.

- **Spatially Immersive Display (SID)**: A computer graphics display which surrounds the user on more than one side. SIDs are usually implemented with rear-projection screens. Common SID types include the CAVE™ (Cruz-Neira, Sandin, and DeFanti, 1993) and dome displays. SIDs do not require the user to wear any headgear, except for stereo viewing glasses if stereoscopic graphics are used.
- **Tracker**: A device that measures 3D position, and sometimes orientation, relative to some known source. Common tracker types are electromagnetic, optical, ultrasonic, gyroscopic, and mechanical linkage (Meyer & Applewhite, 1992).

1.3 Problem Statement

How can we begin to analyze interaction techniques for immersive virtual environments? There are a multitude of tasks which one might conceivably want to perform within a VE, and most of them are application-specific. However, we can reduce the space of the problem by recognizing that there are a few basic interaction "building blocks" that most complex VE interactions are composed of. Such an approach is similar to that proposed by Foley for interaction in a 2D graphical user interface (Foley, 1979).

If, then, we can identify these universal tasks, understand them, and evaluate techniques for them, we will have come a long way towards understanding the usability and interaction requirements for immersive VE applications. From our experience with VE applications and discussion with other researchers, we have identified four task categories: *travel*, *selection*, *manipulation*, and *system control*.

Travel, or viewpoint motion control, refers to a task in which the user interactively positions and orients her viewpoint within the environment. Since head tracking generally takes care of viewpoint orientation, we are mainly concerned with viewpoint translation: moving from place to place in the virtual world. Selection is a task that involves the picking of one or more virtual objects for some purpose. Manipulation refers to the modification of the attributes of virtual objects, such as position, orientation, scale, shape, color, or texture. Selection and manipulation tasks are often paired together, although selection may be used for other purposes (e.g. denoting a virtual object whose color is to be changed). Finally, system control encompasses other commands that the user gives to accomplish work within the application (e.g. delete the selected object, save the current location, load a new model). We will not consider system control separately in this work.

For each of these universal interaction *tasks*, there are many proposed interaction *techniques*. For example, one could accomplish a selection technique in a very indirect way, by choosing an entry from a list of selectable objects. Alternately, one could use a direct technique, where the user moves his (tracked) virtual hand so that it touches the virtual object to be selected. Each of these interaction techniques has advantages and disadvantages, and the choice of a certain technique may depend on many parameters.

In general, we feel that interaction techniques for immersive VEs have been designed and developed in an ad hoc fashion, often because a new application had unusual requirements or constraints that forced the development of a new technique. With few exceptions, ITs were not designed with regard to any explicit framework, or evaluated quantitatively against other techniques. Currently, then, we have a large collection of ITs for VEs, but little in-depth understanding of their characteristics or analysis of their relative performance.

The goals of this research, then, are four-fold:

1. To develop formal characterizations of the universal interaction tasks and formal categorizations or taxonomies of interaction techniques for those tasks,
2. to use these characterizations to design new techniques for each of the universal tasks,
3. to develop and utilize quantitative experimental analyses for the purpose of comparing the performance of interaction techniques for the universal tasks, and
4. to show the validity of the formal frameworks and evaluations by applying experimental results to a real-world VE application which involves all of the universal interaction tasks.

1.4 Scope of the Research

A complete and thorough understanding of VE interaction and user interfaces is not a realizable goal at this point in the maturity of the research area. Therefore, in this work we will focus on specific pieces of the overall problem with high levels of importance and benefit to the VE community.

First, this thesis focuses on low-level interaction techniques – small methods that are used to carry out a single user task. We feel that VE interaction must be understood at this level before we can begin to discuss complete VE user interface metaphors. This is similar to the situation in 2D user interfaces when graphical UIs first became popular. The first step was to develop ITs that performed well and were easily understandable, such as push buttons, pull-down menus, windows, and sliders. Only when this was complete could these elements be combined to form a usable interface. This does not mean that we are neglecting the context in which interaction is performed; on the contrary this context is explicitly included in our design and evaluation framework. We simply desire to understand the components of a usable VE interface before proposing complete interfaces.

Second, this thesis assumes that the goal of interaction is a high level of performance. This may seem overly restrictive, but we take a broad definition of performance which includes not only time for task completion and accuracy, but also more qualitative measures such as ease of use, user comfort, and even the level of presence. Using this definition, almost any application can specify its interaction requirements in terms of performance metrics. However, there are cases in which the goal of a VE application is only loosely based on these performance metrics, such as a VE which simply attempts to replicate interaction in the real world (a naturalistic metaphor). Techniques such as these will not be considered in our design and evaluation.

Third, we choose to consider ITs for a small number of very common and important VE user tasks. Certainly, many interactive VEs contain tasks other than travel, selection, and manipulation, but these three seem to be the most universal and important to understand initially. Furthermore, many more complex interaction tasks are actually composed, at least in part, of these three tasks. Thus, we aim to identify techniques which produce high levels of performance on these generic tasks, so that these techniques can then be applied to the more specific tasks in an application. We do not claim that a general technique will always have better performance than one designed specifically for the task at hand (in fact, this may rarely be the case), but it is impractical to design a new interaction technique for each task in each application. At some level, interaction needs to be more general or even standardized.

Fourth, we are restricting our study to those techniques which are useful in immersive VEs. This choice is purely a function of our interests, and we make no claim that immersive VEs are better than other types of three-dimensional environments. We do, however, claim that immersive VEs are useful for certain tasks, domains, and applications because of their unique properties of immersion, immediacy, whole-body input, etc. Also, the general principles derived from this work should be applicable to many types of systems, and not only immersive VEs.

Fifth, we focus on single-user systems only. A large body of research into multi-user, collaborative VEs is emerging, and these have their own sets of issues related to interaction. Again, however, we feel that we must know more about the simple case in which only one user interacts with the environment before moving on to more complex multi-user VEs.

Finally, this work is restricted to a small number of physical input and output devices that are in common use. For display, all of our studies will use a head-mounted display (HMD), and simple, non-spatialized audio. We will not consider localized sound, haptics, olfactory feedback, or other non-standard forms of output. On the input side, we restrict our study to combinations of six degree of freedom trackers and simple button devices. No specialized input devices will be used or designed in this work. However, some of our experiments and applications will make use of passive physical props. These are non-instrumented physical objects that add realism, constraints, or other additional information to the virtual environment. For the most part, however, the techniques we discuss will differ only in their software implementation, not in the devices they use.

These decisions were not made arbitrarily. Rather, we are seeking to understand a simple subset of interaction techniques for VEs. This subset consists of techniques that can be implemented easily by anyone with a standard VE configuration. In many cases, it may be useful to go beyond these boundaries (for example, to build a new input device that matches a certain task), but the techniques we are studying are generally applicable to a wide range of possible applications.

1.5 Hypotheses

Our work covers a large territory in the overall field of VE interaction. However, there are three broad hypotheses that we have attempted to demonstrate in all phases of this research.

1. Intuition alone is not sufficient for the development of useful and usable (well-performing) interaction designs for VE applications.
2. Formal evaluation of VE interaction techniques will lead to specific and easily applied guidelines for the development of VE user interfaces.
3. The use of our formal methodology for the design and evaluation of VE interaction techniques will cause a measurable increase in the performance and usability of a real VE application to which evaluation results are applied.

We will refer to these hypotheses often throughout this thesis.

1.6 Contributions

This research makes a number of contributions to the fields of virtual environments, three-dimensional interaction, and HCI:

1. Our understanding of 3D interaction techniques has been extended from an intuitive feel for a technique's performance (often incorrect) to empirical measurements of performance and a formal understanding of the relationships between techniques.
2. The taxonomies and other parts of the design and evaluation framework provide a common ground for discussion and research in a more detailed and systematic fashion than simple lists of techniques or metaphors.
3. The combination of empirical results and formal frameworks provides the opportunity to create predictive models of technique performance.
4. The design and evaluation methodology can be reused to create and assess techniques for other VE interaction tasks.
5. The evaluation testbeds themselves can be reused to assess new interaction techniques for the tasks of travel, selection, and manipulation and compare their performance to previously tested techniques.
6. An indirect result of this research is a virtual environment application for environmental design education that has been shown to be both effective in its domain and to exhibit high levels of usability.
7. Finally, and perhaps most importantly, our experience in designing and evaluating VE interaction techniques has led to general principles and specific guidelines and recommendations (sections 1.8, 7.1) that can be used by application developers when creating highly interactive VEs.

1.7 Summary of This Work

In this chapter, we have introduced the subject of interaction techniques for VEs, motivated the need for research in this area, and defined the terms we will use, the scope of the work, our hypotheses, and our contributions.

Chapter two will present a detailed look at previous work that has influenced or informed the current research. This includes research into interaction in 2D interfaces, the evaluation of virtual environments, low-level perceptual and cognitive psychology work, and current three-dimensional user interfaces and interaction techniques.

Chapter three presents our design and evaluation methodology, with all of its component parts. This formal and systematic methodology is the abstract basis for the specific research that will be presented in later sections.

Chapter four applies this methodology to the task of travel, or user viewpoint movement control. We present descriptions of current travel techniques, taxonomies of techniques, and the results from five experiments comparing techniques for various tasks. We also discuss a travel testbed evaluation and its results.

In Chapter five, the methodology is applied to object selection and manipulation. Again, we discuss techniques from the literature, a taxonomy of techniques, and results of our evaluation of techniques. A testbed evaluation is also performed, and its results are presented in detail.

Chapter six describes a real-world VE application which is highly interactive. We discuss the initial two phases of interaction design for this application and the usability problems we encountered. We then describe the changes we made to the system based on the results of our evaluation, and the usability improvements that resulted.

Finally, we conclude in chapter seven with a discussion of the main contributions of this research and possibilities for future work in this area. In particular, this chapter contains detailed explanations of the guidelines and principles that have emerged from this research, so it will be of particular interest to application developers and interaction designers.

1.8 Summary of Recommendations

Our extensive design and evaluation of VE interaction techniques has led to a set of general principles and guidelines. Since these will likely be the most important legacy of this research, we list these recommendations here, and present a detailed exposition of them in chapter seven. The guidelines are divided into four categories: general principles for VE interaction, and guidelines for the design of travel, selection, and manipulation techniques.

1.8.1 Generic VE Interaction Guidelines

1. Do not assume that natural techniques will be the most intuitive or that they will have the best performance.
2. Provide redundant interaction techniques for a single task.

1.8.2 Guidelines for the Design of Travel Techniques

1. Make simple travel tasks simple by using target-specification techniques.
2. Avoid the use of teleportation; instead, provide smooth transitional motion between locations.
3. If steering techniques are used, train users in strategies to acquire survey knowledge. Use target-specification or route-planning techniques if spatial orientation is required but training is not possible.
4. Constrain the user's travel to two dimensions if possible to reduce cognitive load.
5. Use non-head-coupled techniques for efficiency in relative motion tasks. If relative motion is not important, use gaze-directed steering to reduce cognitive load.

1.8.3 Guidelines for the Design of Selection Techniques

1. Use ray-casting techniques if speed of remote selection is a requirement.
2. Ensure that the chosen selection technique integrates well with the manipulation technique to be used.
3. If possible, design the environment to maximize the perceived size of objects.

1.8.4 Guidelines for the Design of Manipulation Techniques

1. Reduce the number of degrees of freedom to be manipulated if the application allows it.
2. Provide general or application-specific constraints or manipulation aids.

3. Allow direct manipulation with the virtual hand instead of using a tool.
4. Avoid repeated, frequent scaling of the user or environment.
5. Use indirect depth manipulation for increased efficiency and accuracy.

CHAPTER II

INTERACTION IN VIRTUAL ENVIRONMENTS

The research presented here has roots in several diverse fields, and builds on many previous results. In this chapter, we will briefly discuss prior work in related disciplines that has an overall bearing on this work. This includes concepts from the field of human-computer interaction, types of user interface evaluation, work in three-dimensional UIs and interaction, related work in the areas of perceptual and cognitive psychology, and previous efforts to evaluate components of immersive virtual environments. This general background will be presented here, but we will reserve discussion of research related to the particular tasks of viewpoint motion control, selection, and manipulation to the appropriate chapters devoted to those subjects.

2.1 Human-Computer Interaction Concepts

As we have noted, there exists a large body of work in the field of human-computer interaction that informs the current research. Many of the specific results and guidelines that are offered by HCI practitioners (e.g. Hix and Hartson, 1993) do not apply directly to immersive VEs, because of the difficulties of interaction in three dimensions, the difference in input and output devices, slower system responsiveness, and so on. However, these specific recommendations can often be generalized to principles that apply in any type of human-computer interface.

One set of general principles, or heuristics, were given by Nielsen (1993). He claimed that a small set of heuristics could account for a large percentage of the usability problems in any interactive system, given a sufficient number of experts to study the system. These heuristics are quite general (e.g. "speak the user's language"), and so they apply to any human-computer interface. However, this generality also causes the heuristics to be difficult to apply practically. In our research, we are searching for specific recommendations for virtual environment interfaces.

Some of the best known principles were described by Norman in the classic work entitled *The Design of Everyday Things* (Norman, 1990). These principles which apply to user interfaces are taken from a discussion of everyday artifacts that we use in our homes, schools, and offices. Since many virtual environments purport to represent a semi-realistic world, it is perhaps even more important that interaction in VEs follow these guidelines (Bowman and Hodges, 1995). Norman identifies four characteristics of usable artifacts: affordances, constraints, good mappings, and feedback. Affordances refer to the properties of an object that inform the user of its purpose and the way it can be used. Constraints are limitations on the use of an object that guide users into proper actions. Good mappings mean that the conceptual model, or metaphor, on which an object is based is easily understood in the specific task domain of the object. Finally, feedback is the indication given by an artifact of the state of its operation or usage, to help the user understand what has happened so that the next action can be planned and carried out.

The idea of mappings proposed by Norman is related to previous work on the use of mental models (a user's understanding of the operation of an artifact) and metaphor (using the understanding of a known concept or object to explain the workings of an artifact) (Gentner and Stevens, 1983). The use of metaphor is an important strategy for UIs since we can explain to someone how to use a software application in terms of something he already understands. The risk is that an inappropriate metaphor will mislead or confuse the user, or that a forced metaphor, while understandable, may degrade user performance.

In traditional 2D user interfaces, there are two major categories of general metaphors. The *conversational metaphor* proposes a dialogue between the user and the computer in the form of a conversation. That is, the user issues a command and the system responds. This metaphor was largely used in command-line interfaces, such as a UNIX shell, but still exists in today's graphical UIs in the form of menu commands, dialog boxes, and so on. The other dominant metaphor is the *simulated world metaphor*, which represents the constructs of a computer application as objects with predictable behaviors in a mini-world. A common example is the desktop metaphor for personal computers, in which programs and data are represented as files which can be placed in folders, file cabinets, trash cans, and so forth, similar to the way paper documents are organized in an office. Since VEs are seen as virtual worlds, most use a very strong simulated world metaphor for almost all tasks. However, conversational elements may also have a place for certain actions in VEs.

Another important HCI concept relevant to this research is the notion of *task analysis*. Task analysis breaks down a task into its component parts, and formalizes the steps that must be taken to complete a task. This explicit characterization leads to a more detailed understanding of the task, and also to a more structured method for understanding various strategies applied to the task. When applied to UIs, task analysis can provide a framework for the design of ITs, as well as reveal reasons for the successes and failures of current approaches. We will use task analysis heavily in the design and evaluation of ITs for VE tasks.

There is a strong tradition in HCI and Human Factors research of formalizing models of human performance. Methods such as GOMS (Card, Moran, and Newell, 1983) and the Keystroke-level model (Card, Moran, and Newell, 1980) attempt to model human performance for a certain computer task by counting the numbers of low-level actions that must take place for the task to be completed. These low-level parts (based on a task analysis) may be explicit user motions, such as key presses, or cognitive processes that the user must carry out. By assigning time values to each of these low-level components, these models may also predict human speed for interfaces that have not yet been implemented or tested. Although our analysis will not attempt to model user action in such a fine-grained manner, we will follow the spirit of this earlier work.

Finally, the HCI literature has provided us with a number of techniques for UI evaluation. These methods represent a wide range in terms of cost, numbers of users needed, formality, and types of results. One of the most simple techniques is guideline-based evaluation (Nielsen and Molich, 1992), which is an informal analysis based on known principles such as those discussed above. This requires only that an expert or group of experts think about and/or use the interface briefly, and can often identify serious problems at an early development stage. The cognitive walkthrough technique (Polson et al, 1992) is similar in that only UI experts are needed to carry it out, but it attempts to be slightly more formal by requiring the evaluators to follow a strict process and answer specific questions about each task within the interface. One of the most common evaluation methods is the usability study (Williges, 1984). Here, several users perform prescribed tasks with the UI, and are observed for task completion time, errors, and other issues. This method is slightly more time-consuming and expensive, but can identify important problems because of the fact that actual users participate. To obtain results that are even more applicable to the real world, some have also performed observations of users in the field (Holtzblatt and Jones, 1993), although it is questionable whether users work in a normal way while being observed. Finally, UI researchers can perform formal experiments in the scientific tradition (Eberts, 1994), which have specific hypotheses, are tightly controlled, and use statistical analysis to obtain results. These are the most expensive and time-consuming studies, and are usually used to obtain basic knowledge about an interface or technique that is quite different from that which has gone before it. Since our research falls into this category, we will make use of formal experimentation in our evaluation of ITs for virtual environments.

2.2 Three-Dimensional User Interfaces

User interface research has only recently begun to seriously consider truly three-dimensional applications and the added difficulties that they present. Common personal computer software is still almost exclusively 2D, except in a few niche applications. However, it is becoming increasingly important that 3D UIs are analyzed, understood, and designed well, as more 3D applications become mainstream. These applications include 3D CAD, architectural design, animation, visualization, and even entertainment. In all of these cases, the fact that information is displayed and manipulated in three dimensions provides a new challenge for UI designers. Some of the problems have been identified and categorized (Herndon, van Dam, and Gleicher, 1994, Hinckley et al, 1994), but there are few general principles or solutions for these difficulties.

For desktop 3D applications, the limitations and inherent 2D nature of common input devices, such as the mouse, pose a major challenge. In these cases, the two degrees of freedom (DOFs) of the input device must be mapped onto three, or in some cases six (three translational and three rotational), dimensions. For this reason, a good deal of research has gone into the design and analysis of input devices specifically for 3D applications (MacKenzie, 1995). One of the most common devices is the tracker (Meyer and Applewhite, 1992), which is a 6 DOF digitizer – that is, it is a sampling device which continuously outputs six scalar values representing position and orientation. Other devices like the Spaceball (TM) (Spacetec IMC, 1998) and the Sidewinder (TM) (Microsoft, 1998) are self-centering devices which sense displacement and rotation in all six dimensions. Other designs have focused on modifying the mouse, such as the “Rockin’ Mouse” (Balakrishnan et al, 1997).

Analysis of input devices has been an important research topic in recent years. Card, Mackinlay, and Robertson provided a formal framework for design and evaluation of both 2D and 3D devices (Card, Mackinlay, and Robertson, 1990). Other studies have focused on the experimental comparison of two or more of these devices. For example, Zhai and Milgram (1993) compared the tracker to the spaceball for an object placement task. One problem with many of these studies is their implicit assumption that the input device and the interaction technique are inextricably linked. That is, an input device determines the IT that must be used with it. We recognize the importance of a well-designed input device in a usable system, but claim that ITs can be evaluated separately. All of our experiments, then, will use a tracker for 6D input, but a multitude of different ITs will be studied.

Another area that has seen many research efforts is the standardization of 3D interfaces, analogous to the ubiquitous desktop metaphor in 2D. It has been argued that such standardization is necessary before VEs can become accepted tools in the real world. Several research efforts have attempted to provide a single interface metaphor that can allow usability and productivity for a wide range of VEs (e.g. Wloka and Greenfield, 1995, Rygol et al, 1995). We would claim, however, that standardization is not necessarily beneficial for VE interfaces at their current level of maturity. Rather, we will focus on optimizing the interaction for specific tasks in particular domains.

Recently, the field of *two-handed interaction* in three dimensions has been researched extensively. Two-handed interfaces are a new paradigm for 3D input that attempt to take advantage of the human ability to use both hands simultaneously to provide more intuitive, comfortable, and productive applications. Hinckley's work in this area is quite instructive (Hinckley et al, 1997). Using previous work in the analysis of two-handed tasks such as handwriting, he showed the validity of several principles for two-handed interfaces. These include the ideas that the hands should work complementarily, not necessarily in parallel, that the non-dominant hand provides a frame of reference within which the dominant hand works, and that the non-dominant hand is good at large, coarse-grained manipulation, while the dominant hand excels at fine-grained work. These principles have been applied to several non-immersive 3D applications (e.g. Goble et al, 1995, Mapes and Moshell, 1995), with encouraging results. We feel that the use of two-handed interfaces in immersive VEs is quite promising, and therefore will include two-handed techniques in our design and evaluation.

2.3 Perceptual and Cognitive Psychology Concepts

Since our research focuses on human performance when interacting with VEs, it is only natural that we should use the results of prior work investigating human capabilities in general. Much of this information comes from the fields of perceptual and cognitive psychology. Perceptual psychology studies the ways humans perceive their environment through the senses, while cognitive psychology focuses on the mental aspects – how humans reason, learn, remember, etc.

Since most immersive VEs are highly visual, it is quite important that we understand human visual perception. In particular, depth perception is crucial, since we are attempting to represent a 3D environment on 2D displays. Research has identified many visual cues that humans use to determine depth, and divide them into monocular vs. binocular (using one or two eyes), and static vs. dynamic (available from a single image or requiring motion) cues (Bruce and Green, 1990). Most depth cues are static and monocular, including linear perspective, texture gradient, relative height, and aerial perspective. Motion parallax, referring to the understanding of depth gained from head or eye motion, is a dynamic monocular cue. Stereopsis – the depth effect due to the fact that our two eyes receive two slightly different images of the world – is characterized as static binocular. Finally, there are oculomotor depth cues, which, unlike the others, do not depend on the images received at the retinae. These cues rely on information from the muscles which cause the eyes to focus (accommodation) and rotate (convergence). We cannot achieve a perfect representation of depth in current VEs, because the actual images all appear on a screen at a single depth, and therefore the oculomotor cues – cues based on the convergence angle and accommodation of the eyes – are in conflict with other depth cues.

Stereo in particular is widely believed to be a very important depth cue that enhances immersive VEs. However, it is quite difficult to achieve a proper stereo effect, as it requires care in calibration, measurement, and rendering of the stereo pair (Davis and Hodges, 1995). Many studies have been performed comparing human performance in stereoscopic, monocular, and biocular (the same image presented to both eyes) viewing situations, and the general consensus is that stereo improves presence and can improve performance (Barfield, Hendrix, and Bystrom, 1997, Hendrix and Barfield, 1996). On the other hand, some studies have found that the addition of other cues to a non-stereo display may produce performance that is as good or better than performance with a stereoscopic display (e.g. Nemire, 1996). Because of technological limitations, our studies will use biocular displays, but will include many additional depth and feedback cues to aid performance.

Wickens has presented a good summary of the application of cognitive psychology to VEs (Wickens and Baker, 1995). An important concept from cognitive psychology that relates to the current work is the model human processor (Card, Moran, and Newell, 1986). This describes the cognitive process that people go through between perception and action. It is important to the study of interaction techniques because cognitive processing can have a significant effect on performance, including task completion time, number of errors, and ease of use. A major goal of IT designers is the creation of techniques which use few cognitive resources, and may become automatic in some sense, so that cognitive power may remain focused on the actual task at hand. One particularly important concept is the limitation on working memory described by Miller (1956). He reported that working memory can hold only seven plus or minus two "chunks" of information at a time. If more information needs to be recalled, previous chunks may be displaced or interfered with. Interfaces should be designed so that this limited space can be used for domain-specific information.

Finally, perceptual and cognitive psychology have shed light on individual differences in the ability of humans. One such line of work that relates to the current discussion is the study of spatial ability (McGee, 1979). Humans vary in their ability to reason spatially, especially in three dimensions. Studies such as the classic mental rotation experiments (Cooper and Shepherd, 1978) have demonstrated these differences. A user's spatial ability can have a significant effect on their performance in 3D interaction tasks. Therefore, we must be sure to consider individual differences when designing and evaluating ITs. Designers should attempt to create techniques which perform robustly for users with a wide range of spatial abilities. In evaluation, we must take care not to attribute a performance difference to the difference in techniques when it is actually caused by a user-specific characteristic, such as spatial ability.

2.4 Evaluation of Immersive Virtual Environments

Although virtual environments have been in existence for over thirty years, it has only been in the last few years that researchers have really begun to perform analysis and evaluation of technology, techniques, and applications of VEs. As stated earlier, this type of research is necessary if VEs are to become useful in the real world. In this section, then, we will review some of the work that has been done to quantify the effectiveness of VEs and human performance within them.

One question that should be asked at the outset is, “What evidence is there that immersive VEs are better than other types of computer applications for ANY tasks?” Researchers have addressed this issue in both a general sense and in specific applications. For example, Pausch et al (1993) performed a study comparing human performance using a head-mounted display with and without head tracking, and reported that head-tracking had a significant effect in improving results. In the application domain, Hodges et al (1995) have shown an immersive VE can produce results for psychological therapy which are similar or equivalent to those achieved when a physical environment is used.

Another issue that has intrigued researchers is the measurement of presence, or immersion. Barfield has attempted in several studies (Barfield et al, 1995) to relate the level of presence to task performance. Slater’s work (Slater, Usoh, and Steed, 1994, 1995) has examined the effects of various display modalities, interaction techniques, and system algorithms on the reported level of presence. One problem with this type of research is the lack of a standard definition of presence and an appropriate measurement technique. Most studies have used qualitative measures (e.g. interviews or questionnaires), although some have attempted to relate other values to the sense of presence.

Another area of current research is the effect of various low-level system characteristics on performance in immersive VEs. Besides the studies addressing display type mentioned earlier, there have been experiments on the effect of mean frame rate (Richard et al, 1995), variance of frame rate (Watson et al, 1997), and level of visual detail (Watson, Walker, and Hodges, 1995). These experiments have generally used a standard task, such as visual search or pick and place, and compared users’ speed and accuracy under the various experimental conditions. Such studies are similar in format to those we will present in this work, although our main independent variables are higher-level entities (interaction techniques). Based on this body of work, our studies will attempt to provide a “near best case” system environment, with a high average frame rate, low frame rate variance, and high visual detail in the entire display, so that our results will not be confounded by these variables.

Finally, recent research has attempted to apply common HCI design and assessment techniques to VEs. The most common example of this is the summative usability study, in which users do a structured set of tasks within a complete system or prototype system in order to reveal usability problems that can be solved in the next design iteration. It is becoming more common for VE developers to perform usability studies to verify the effectiveness of their designs (e.g. Bowman, Hodges, & Bolter, 1998, Arns, Cook, & Cruz-Neira, 1999). The concept of usability engineering includes guidelines and evaluation throughout the design cycle of a system, and this model has begun to see use for VEs as well (Hix et al, 1999).

There has been little work in the evaluation of specific interaction techniques for immersive VEs, although this may be changing. We will forgo a discussion of this body of work for now. Rather, in each of the chapters on a specific interaction task, we will review the relevant research on IT design and evaluation for that task.

CHAPTER III

DESIGN AND EVALUATION CONCEPTS

We wish to perform our design and evaluation of interaction techniques for immersive virtual environments in a principled, systematic fashion (see e.g. Price, Baecker, and Small, 1993, Plaisant, Carr, and Shneiderman, 1995). Formal frameworks provide us not only with a greater understanding of the advantages and disadvantages of current techniques, but also with better opportunities to create robust and well-performing new techniques, based on the knowledge gained through evaluation. Therefore, this research will follow several important design and evaluation concepts, elucidated in the following sections.

3.1 Taxonomy and Categorization

The first step in creating a formal framework for design and evaluation is to establish a *taxonomy* of interaction techniques for each of the universal interaction tasks (note on the word ‘taxonomy’: we will employ both of its accepted meanings: “the science of classification,” and “a specific classification”). Taxonomies partition the tasks into separable subtasks, each of which represents a decision that must be made by the designer of a technique. In this sense, a taxonomy is the product of a careful task analysis. For each of the lowest level subtasks, technique components (parts of an interaction technique that complete that subtask) may be listed. Figure 2.1 presents a simple generalized taxonomy, including two levels of subtasks, and several technique components. Taxonomies for the tasks of travel (sections 4.3.1 and 4.6.1) and selection/manipulation (section 5.4.1) are presented later in the thesis.

The taxonomies must come from a deep and thorough understanding of the interaction task and the techniques that have been proposed for it. Therefore, some initial informal evaluation of techniques and/or design of new techniques for the task is almost always required before a useful taxonomy can be constructed (section 3.4).

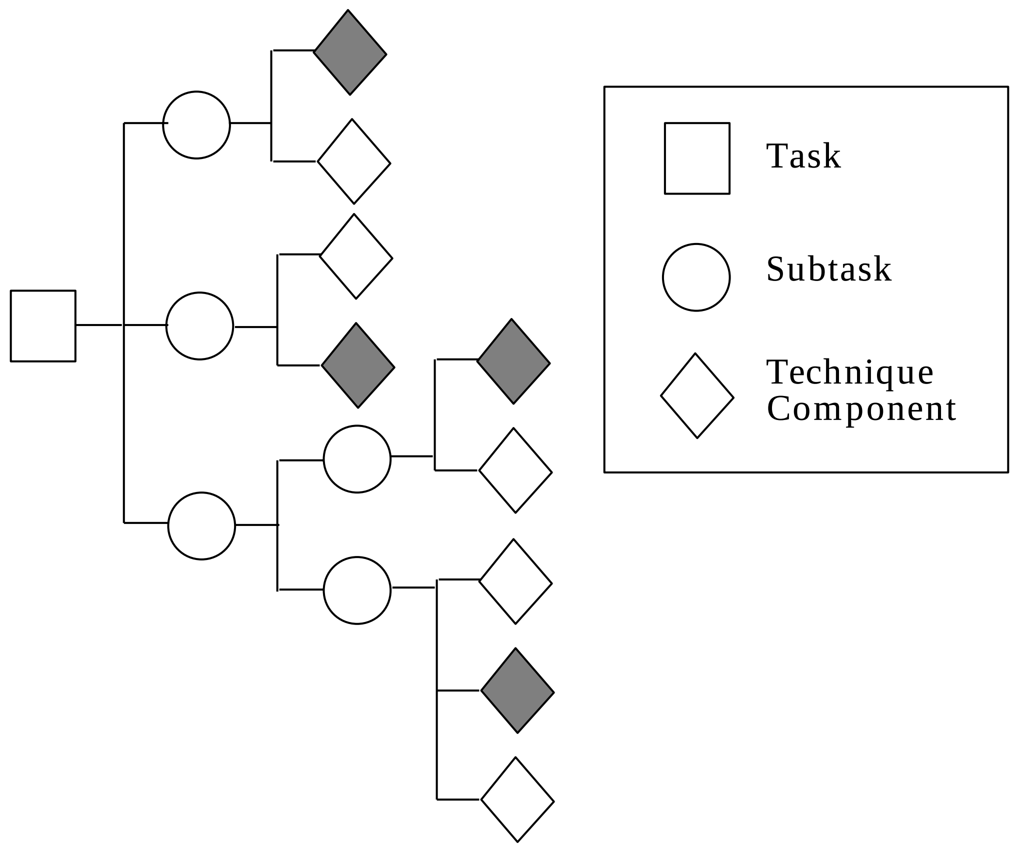


Figure 2.1 General Taxonomy Format

Let us consider a simple example. Suppose the interaction task is to change the color of a virtual object (of course, this task could also be considered as a combination of universal interaction tasks: select an object, select a color, and give the “change color” command). A taxonomy for this task would include several task components. Selecting an object whose color is to change, choosing the color, and applying the color are components which are directly task-related. On the other hand, we might also include components such as the color model used or the feedback given to the user, which would not be applicable for this task in the physical world, but which are important considerations for an IT.

Ideally, the taxonomies we establish for the universal tasks need to be complete and general. Any IT that can be conceived for the task should fit within the taxonomy, and should not contain components that are not addressed by the taxonomy. Thus, the components will necessarily be abstract. The taxonomy will also include several possible choices for each of the components, but we do not necessarily expect that each possible choice will be included. For example, in the object coloring task, a taxonomy might list touching the virtual object, giving a voice command, or choosing an item in a menu as choices for the color application component. However, this does not preclude a technique which applies the color by some other means, such as pointing at the object.

Moreover, we do not claim that any given taxonomy represents the "correct" partitioning of the task. Different users have different conceptions of the subtasks that are carried out to complete a task. Rather, we see our taxonomies as practical tools that we use as a framework for design and evaluation (see below). Therefore, we are concerned only with the utility of a taxonomy for these tasks, and not its "correctness." In fact, we discuss two possible taxonomies for the task of travel, both of which have been useful in determining different aspects of performance. Rules and guidelines have been set forth for creating proper taxonomies (Fleishman & Quaintance, 1984), but we felt that the structure of these taxonomies did not lend itself as well to design and evaluation as the simple task analysis.

One way to verify the generality of the taxonomies we create is through the process of *categorization*. If existing techniques for the task fit well into the taxonomy, we can be more sure of its completeness. Categorization also serves as an aid to evaluation of techniques. Fitting technique components into a taxonomy makes explicit their fundamental differences, and we can determine the effect of choices in a more fine-grained manner. Returning to our example, we might perform an experiment comparing many different techniques for coloring virtual objects. Without categorization, the only conclusions we could draw would be that certain techniques were better than others. Using categorization, however, we might find that the choice of object selection techniques had little effect on performance, and that color application was the most important component in determining overall task time.

3.2 Guided Design

Taxonomy and categorization are good ways to understand the low-level makeup of ITs, and to formalize the differences between them, but once they are in place, they can also be used in the design process. We can think of a taxonomy not only as a characterization, but also as a design space. In other words, a taxonomy informs or guides the design of new ITs for the task, rather than relying on a sudden burst of insight (hypothesis 1).

Since a taxonomy breaks the task down into separable subtasks, we can consider a wide range of designs quite quickly, simply by trying different combinations of components for each of the subtasks. For example, the shaded components in figure 2.1 represent a possible complete interaction technique. There is no guarantee that a given combination will make sense as a complete IT, but the systematic nature of the taxonomy makes it easy to generate designs and to reject inappropriate combinations.

Categorization may also lead to new design ideas. Placing existing techniques into a design space allows us to see the "holes" that are left behind – combinations of components that have not yet been attempted. One or more of the holes may contain a novel, useful technique for the task at hand. This process can be extremely useful when the number of subtasks is small enough and the choices for each of the subtasks are clear enough to allow a graphical representation of the design space, as this makes the untried designs quite clear (Card, Mackinlay, and Robertson, 1990).

3.3 Performance Measures

The overall goal of this research is to obtain information about human performance in common VE interaction tasks – but what is performance? As computer scientists, we tend to focus almost exclusively on speed, or time for task completion. Speed is easy to measure, is a quantitative determination, and is almost always the primary consideration when evaluating a new processor design, peripheral, or algorithm. Clearly, efficiency is important in the evaluation of ITs as well, but we feel there are also many other response variables to be considered.

Another performance measure that might be important is accuracy, which is similar to speed in that it is simple to measure and is quantitative. But in human-computer interaction, we also want to consider more abstract performance values, such as ease of use, ease of learning, and user comfort. For virtual environments in particular, presence might be a valuable measure. The choice of interaction technique could conceivably affect all of these, and they should not be discounted.

We should remember that the reason we wish to find good ITs is so that our applications will be more usable, and that VE applications have many different requirements. In many applications, speed and accuracy are not the main concerns, and therefore these should not always be the only response variables in our evaluations.

Also, more than any other computing paradigm, virtual environments involve the user – his senses and body – in the task. Thus, it is essential that we focus on user-centric performance measures. If an IT does not make good use of the skills of the human being, or if it causes fatigue or discomfort, it will not provide overall usability despite its performance in other areas. In this work, then, we will evaluate based on multiple performance measures that cover a wide range of application and user requirements.

3.4 Range of Evaluation Methods

Research in HCI has introduced a wide range of interface evaluation techniques, as discussed earlier. Evaluators have a choice regarding the statistical validity of their tests, the number of users involved, the time and effort required, and the results they wish to achieve. In this research, we feel that many of these techniques are appropriate for various stages of evaluation.

Initially, we come to look at these interaction tasks and techniques with very little concrete information, except our experience with them in applications, and in a few cases the published evaluations of others. Our first goal is to establish a taxonomy and perform categorization, but this is difficult given limited information. Therefore, in many cases it is appropriate to perform some informal evaluation at the beginning to gain a base of understanding of both the task and techniques. This may take the form of a guideline-based evaluation, where one or more usability experts try the techniques and note obvious problems and successes. In many cases, since there are few guidelines or experts in this field to draw from, an informal user study would be useful, in which a few users try out the techniques on some representative tasks, and their general performance and comments are noted. Finally, if the techniques have already been implemented as part of an application, a usability study with some quantitative measures may provide some good information.

Once we are familiar with the task and some techniques, we can create an initial taxonomy and formal framework for evaluation. Within this framework, more formal experimentation can be performed. These experiments are likely to be quantitative, statistically valid, and low-level (meaning that the test does not involve a full application, but only a tightly-controlled system with low-level interaction tasks). In order to further our understanding, these experiments should focus on specific technique components and performance measures, so that it can be determined what the important variables are. From these results, we can refine our taxonomy and evaluation framework, and prepare for testbed evaluation, which is described in the next section.

All of these types of evaluation lead to both specific results and practical guidelines (hypothesis 2) that apply to VE interfaces.

3.5 Testbed Evaluation

The experimental methods and other evaluation tools discussed above can be quite useful for gaining an initial understanding of interaction tasks and techniques, and for measuring the performance of various techniques in specific interaction scenarios. However, there are some problems associated with using these types of tests alone.

First, while results from informal evaluations can be enlightening, they do not involve any quantitative information about the performance of interaction techniques. Without statistical analysis, key features or problems in a technique may not be seen. Performance may also be dependent on the application or other implementation issues when usability studies are performed.

On the other hand, formal experimentation usually focuses very tightly on specific technique components and aspects of the interaction task. An experiment may give us the information that technique X performs better than technique Y in situation Z, but it is often difficult to generalize to a more meaningful result. Techniques are not tested fully on all relevant aspects of an interaction task, and generally only one or two performance measures are used.

Finally, in most cases, traditional evaluation takes place only once and cannot truly be recreated later. Thus, when new techniques are proposed, it is difficult to compare their performance against those that have already been tested.

Therefore, we propose the use of *testbed evaluation* as the final stage in our analysis of interaction techniques for universal VE interaction tasks. This method addresses the issues discussed above through the creation of testbeds – environments and tasks that involve all of the important aspects of a task, that test each component of a technique, that consider outside influences (factors other than the interaction technique) on performance, and that have multiple performance measures.

As an example, consider a proving ground for automobiles. In this special environment, cars are tested in cornering, braking, acceleration, and other tasks, over multiple types of terrain, and in various weather conditions. Task completion time is not the only performance variable considered. Rather, many quantitative and qualitative results are tabulated, such as accuracy, distance, passenger comfort, and the “feel” of the steering.

The VEPAB project (Lampton et al, 1994) was one research effort aimed at producing a testbed for VEs, including techniques for viewpoint motion control. It included several travel tasks that could be used to compare techniques. However, this testbed was not based on a formal understanding of the tasks or techniques involved.

In this work, we have created a series of testbeds for the universal VE interaction tasks of viewpoint motion control, selection, and manipulation. Together, these testbeds make up VR-SUITE – the Virtual Reality Standard User Interaction Testbed Environment.

The testbeds will allow us to analyze many different ITs in a wide range of situations, and with multiple performance measures. Testbeds are based on the formalized task and technique framework discussed earlier, so that the results are more generalizable. Finally, the environments and tasks are standardized, so that new techniques can be run through the appropriate testbed, given scores, and compared with other techniques that were previously tested.

3.6 Models of Human Performance

Testbed evaluation provides us with a good and general technique for comparing interaction techniques designed for a given task, but this is not the ultimate goal of this research. Rather, we want to be able to design interaction techniques and applications that are more usable and cause users to be more productive. In this light, knowing that a certain technique outperforms another in the tasks required by our application is not good enough, because the best technique may not have been thought of yet! What we really desire, then, is a quantitative model of task performance that lets us determine whether we have reached near-optimal performance, and if not, how we can come closer to it.

If our testbeds were simply representative sets of tasks and environments that seemed intuitively to test techniques fully, it would be difficult or impossible to generalize the results into a performance model, and any model that was created would be quite coarse-grained. However, since the testbeds are grounded in a formal framework that splits tasks, techniques and other factors into fine-grained components, we can create models based on these components which should generalize to produce models that predict the performance of even techniques that were not tested.

We believe there are many benefits of using testbed evaluation combined with formal frameworks to produce models of human performance on the various interaction tasks. However, performance modeling is outside the scope of the current research, and we have left it as future work (chapter 7).

3.7 Application of Results

Testbed evaluation produces a set of results that characterize the performance of an interaction technique for the specified task. Performance is given in terms of multiple performance metrics, with respect to various levels of outside factors. These results become part of a performance database for the interaction task, with more information being added to the database each time a new technique is run through the testbed.

The last step in our methodology is to apply the performance results to VE applications, with the goal of making them more useful and usable. In order to choose interaction techniques for applications appropriately, we must understand the interaction requirements of the application. We cannot simply declare one best technique, because the technique that is best for one application will not be optimal for another application with different requirements. For example, a VE training system will require a travel technique that maximizes the user's spatial awareness, but this application will not require a travel technique that maximizes point-to-point speed. On the other hand, in a battle planning system, speed of travel may be the most important requirement.

Therefore, applications need to specify their interaction requirements before the correct ITs can be chosen. This specification will be done in terms of the performance metrics which we have already defined as part of our formal framework. Once the requirements are in place, we can use the performance results from testbed evaluation to recommend ITs that meet those requirements. These ITs, having been formally verified, should increase the performance levels (including usability) of the application (hypothesis 3).

3.8 Summary of Methodology

Figure 2.2 summarizes the basic design and evaluation methodology we will use for our research on interaction techniques for immersive virtual environments, including each of the components discussed in the previous sections. It should be noted that this process may be slightly different in individual cases, but our design, evaluation, and application will generally follow a procedure similar to this.

For each universal interaction task, the process begins with informal evaluation techniques: observation, user studies, and/or usability evaluations. These should lead to an understanding of the task and the space of possible techniques, which allows us to create a taxonomy and to categorize existing and proposed ITs, and may also inspire the creation of new techniques. We can also list outside factors influencing performance and performance measures at this time. Once this formal framework is in place, we can perform more formal experiments, involving specific task and technique components and performance measures. These results, along with our design framework, may lead to the design and implementation of novel techniques for the task. Also, experimentation may cause some reworking of the initial taxonomy. When the formal framework is judged complete, we can move to the final analysis step: testbed evaluation. Use of the testbed with a range of techniques and performance measures produces a dataset of results for the given task, which can then be used to make an informed choice of ITs for the target application(s), given their performance requirements.

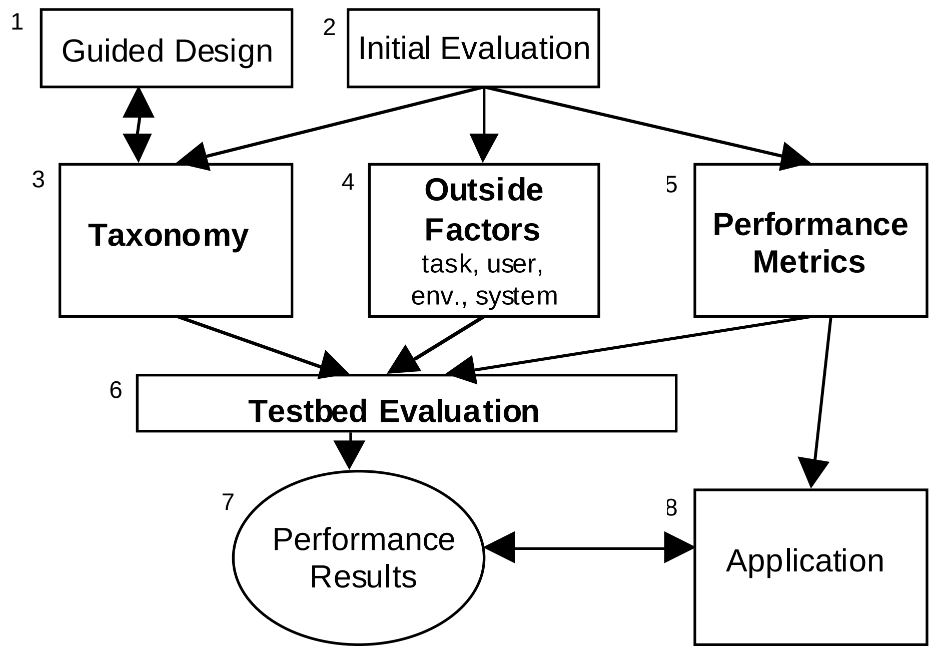


Figure 2.2 Flowchart of Design and Evaluation Methodology

CHAPTER IV

TRAVEL

4.1 Introduction and Definitions

Travel, or *Viewpoint Motion Control*, is one of the most basic and universal interactions found in virtual environment applications. We define travel as the control of the user's viewpoint motion in the three-dimensional environment. This is distinguished from *wayfinding*, which is the cognitive process of determining a path based on visual cues, knowledge of the environment, and aids such as maps or compasses. Together, travel and wayfinding make up the overall interaction called *navigation*. In our work, then, we are studying the techniques which allow a user to move from place to place in a VE, and not the displays or other aids which help the user to find her way.

Travel is almost certainly the most common interaction in VE applications, apart from simple head motion. In most VE systems, the user must be able to move effectively about the environment in order to obtain different views of the scene and to establish a sense of presence within the 3D space. Therefore, it is essential that travel techniques be well-designed and well-understood if VE applications are to succeed. In most cases, travel is not an end unto itself; rather, it is simply used to move the user into a position where he can perform some other, more important task. Because of this, the travel technique should be easy to use, cognitively simple, and unobtrusive. It is not obvious whether a given technique meets these criteria, so formal evaluation and analysis are important.

In this chapter, then, we will explore interaction techniques for viewpoint motion control in immersive VEs, beginning with prior work in the area. Next, an initial evaluation framework and four experiments will be presented. These experiments analyzed some common techniques and had important results, but fell short in other areas. This led to the development of an alternate framework. Another experiment is discussed which shows the relative advantages of the expanded approach. Finally, this chapter will discuss the testbed evaluation we performed for the task of viewpoint motion control.

4.2 Related Work

A number of researchers have addressed issues related to navigation and travel both in immersive virtual environments and in general 3D computer interaction tasks. It has been asserted (Herndon et al, 1994) that studying and understanding human navigation and motion control (e.g. Schieser, 1986, Warren & Wertheim, 1990) is of great importance for understanding how to build effective virtual environment travel interfaces. Although we do not directly address the cognitive issues surrounding virtual environment navigation, this area has been the subject of some prior investigation (e.g. Wickens, 1995). Wayfinding issues have been the subject of studies by Darken and Sibert (1996a, 1996b). Also, a system has been proposed (Ingram & Benford, 1995) which attempts to replicate the classic urban wayfinding cues identified in “The Image of the City” (Lynch, 1960).

Various metaphors for viewpoint motion and control in 3D environments have also been proposed. Ware et al. (1988, 1990, 1996) identify the “flying,” “eyeball-in-hand,” and “scene-in-hand” metaphors for virtual camera control. As an extension of the scene-in-hand metaphor, Pausch et al. (1995) make use of a “World-in-Miniature” representation as a device for navigation and locomotion in immersive virtual environments. Another interesting metaphor uses head motion to control the position of the viewpoint (Kheddar, Chellali, and Coiffet, 1995, Koller, Mine, and Hudson, 1996).

Numerous implementations and studies of non-immersive 3D travel techniques have been described. Strommen compares three different mouse-based interfaces for children to control point-of-view navigation (Strommen, 1994). Mackinlay et al. describe a general method for rapid, controlled movement through a 3D environment (Mackinlay, Card, and Robertson, 1990), and a similar technique is used immersively in the Cosmic Explorer application (Song and Norman, 1993). Ware and Slipp assessed the usability of different velocity control interfaces for viewpoint control in 3D graphical environments (Ware and Slipp, 1991).

Mine (1995) offers an overview of motion specification interaction techniques. He and others (Robinett & Holloway, 1992) also discuss issues concerning their implementation in immersive virtual environments. Several user studies concerning immersive travel techniques have been reported in the literature, such as those comparing different travel modes and metaphors for specific virtual environment applications (e.g. Chung, 1992, Mercurio et al., 1990). Physical motion techniques have also been studied (e.g. Iwata and Fujii, 1996), including an evaluation of the effect of a physical walking technique on the sense of presence (Slater, Usoh, and Steed, 1995).

4.3 Original Evaluation Framework

4.3.1 Categorization of Techniques

Given techniques for travel in immersive virtual environments, one could perform many experiments involving those techniques and come to some understanding of their effect on performance in certain applications. However, it is not entirely clear what determines the “performance” of a travel technique. Moreover, it would be difficult or impossible to determine which components of the techniques were significant in improving or lessening performance, and results from one application or task would not necessarily transfer to another. For this reason, we have devised a more formalized framework within which to evaluate virtual travel techniques. Stanney (1995) proposes that a taxonomy of interaction techniques is needed for “imposing order on the complex interactions between user, task, and system phenomena.” The evaluation framework presented here includes such a taxonomy and an emphasis on outside factors which can influence user performance.

In order to understand travel techniques and their effects more deeply, we need to categorize them and break them down into their lower-level components. Toward this end, we have developed a taxonomy of immersive travel techniques, which is presented in

Figure 4.1. The taxonomy splits a technique into three components, which apply regardless of the type of travel being done (exploration, search, maneuvering, etc.).

Direction/Target Selection refers to the method by which the direction or object of travel is specified. Depending on whether control of direction is continuous or not, the user may either "steer" (choose a direction), or simply choose a target object. Gaze-directed steering, in which the user moves in the direction she is looking, and pointing, where the user points in the direction she wants to go, are two popular steering techniques. This section also lists techniques for discrete selection of a target object.

Velocity/Acceleration Selection techniques allow the user to vary the speed of travel. Many VE applications dispense with this entirely, and use a constant travel velocity. However, several techniques have been proposed, including continuous gestures to select velocity, the use of props such as foot pedals, or adaptive system-controlled speed.

The final component of a travel technique is the *Conditions of Input*. This refers to the input required by the system in order to begin, continue, and end travel. The user may be in constant motion, in which case no input may be required. Alternately, the system may require continuous input to determine the user's state, or simple inputs at the beginning and/or end of a movement. Again, this component may be under system control.

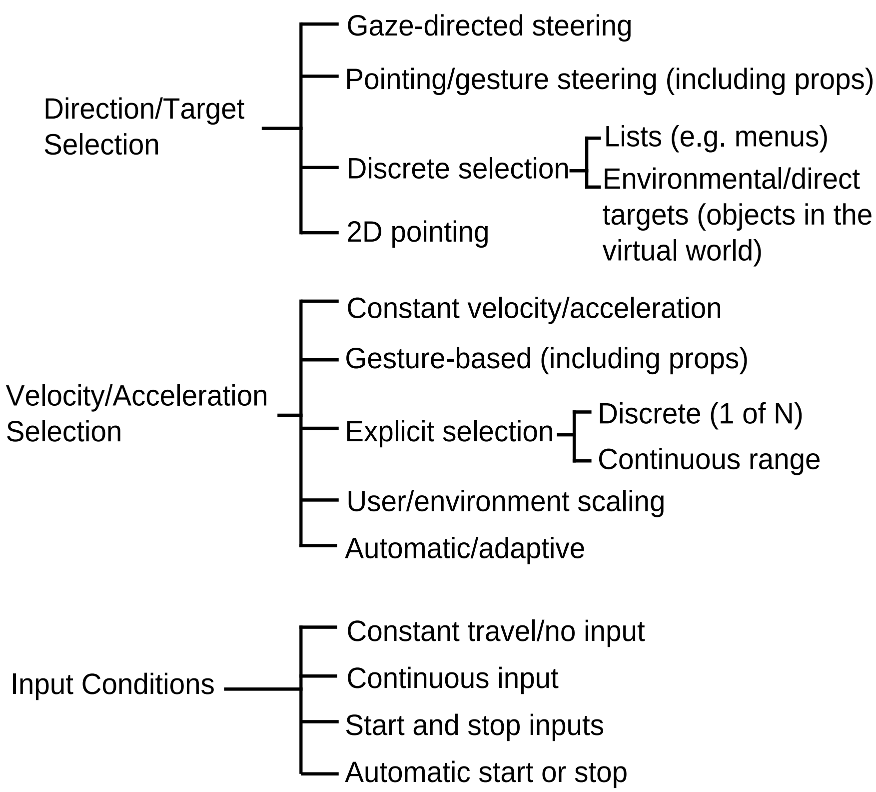


Figure 4.1 Taxonomy of Travel Techniques for Immersive Virtual Environments

We do not claim that this taxonomy is complete, since many new techniques for controlling user motion are being designed. However, most current techniques fit into the taxonomy, at least at a high level. More importantly, by breaking a technique into three components, we can study them separately, and gain a greater understanding of differences in performance. A technique which is performing poorly may be improved by changing only one of the components, but this might not be recognized unless techniques are divided into their constituent elements.

This taxonomy also encourages the guided design of new techniques. By choosing a component (and an implementation of that component) from each section of the taxonomy, a travel technique may be created from its parts, and useful new combinations may come to light. Not all components will work with all others, but there are many opportunities for interesting designs.

For example, one might combine environmental target selection with gesture-based velocity selection, explicit start inputs, and explicit or automatic stop inputs. This would produce a technique that would allow a user to travel along a path from the current position to a specified object, using a high velocity on the less interesting parts and a slower speed at places of interest. The user could stop moving at any point along the path, or be stopped automatically when the target object was reached. Such a technique might be a natural fit for an immersive "tour" application, where there are certain known places that users wish to visit, but designers also desire that movement be under some degree of user control.

We limit the scope of our design and evaluation to travel techniques implementing virtual movement. That is, we will not consider techniques which use physical motions such as walking in place or walking on a treadmill. Such techniques may be quite natural and useful, but are not generally applicable to VE applications, especially when three-dimensional motion is needed.

4.3.2 Performance Measures

There are few categories of virtual environment applications that are currently in use for productive, consistent work, but the requirements of these applications for travel techniques cover a wide range. Further, there are many new applications of VEs being researched, which also may require travel techniques with different characteristics. It is therefore impractical to evaluate travel techniques directly within each new application. Instead, we propose a more general methodology, involving a mapping from travel techniques to a set of performance metrics. These are measurable characteristics of the performance of a technique. With this indirect mapping, application designers can specify desired levels of various metrics, and then choose a technique which best fits those requirements.

Our list of performance metrics for immersive travel techniques includes:

1. *Speed* (efficient task completion)
2. *Accuracy* (proximity to the desired target)
3. *Spatial Awareness* (the user's knowledge of his position and orientation within the environment during and after travel)
4. *Ease of Learning* (the ability of a novice user to use the technique)
5. *Ease of Use* (the complexity or cognitive load of the technique from the user's point of view)
6. *Information Gathering* (the user's ability to actively obtain information from the environment during travel)

7. *Presence* (the user's sense of immersion or "being within" the environment due to travel)
8. *User Comfort* (lack of physical discomfort, including simulator sickness (e.g. Hettinger and Riccio, 1992))

Again, this list may not be complete, but it is a good starting point for quantifying the effectiveness and performance of virtual travel techniques. In particular, we emphasize that speed and accuracy are not the only characteristics of a good travel technique, and in many applications are not the most important. For example, the designer of an architectural walkthrough application might be most interested in high levels of spatial awareness, information gathering, and presence. By doing experiments that relate travel technique components to performance metrics, we can identify techniques that meet those needs, and the results of the experiments will also be generalizable and reusable by designers of other applications.

Some of the metrics, such as speed and accuracy, are simple to measure quantitatively. Others, however, are difficult to measure due to their inherent subjective nature. To quantify these metrics, standard questionnaires for factors such as ease of use (e.g. Chin, Diehl, & Norman, 1988), presence (e.g. Slater, 1995), and simulator sickness (e.g. Kennedy et al., 1993) should be part of the experimental method.

4.4 Initial Experiments

Using this framework, we designed and ran three initial experiments on common VE travel techniques (These experiments are described in more detail in Bowman, Koller, & Hodges, 1997). We wanted to show that generalizable results could be obtained without knowing the target application. These experiments produced useful data which is applicable in a variety of situations.

4.4.1 Spatial Awareness Experiment

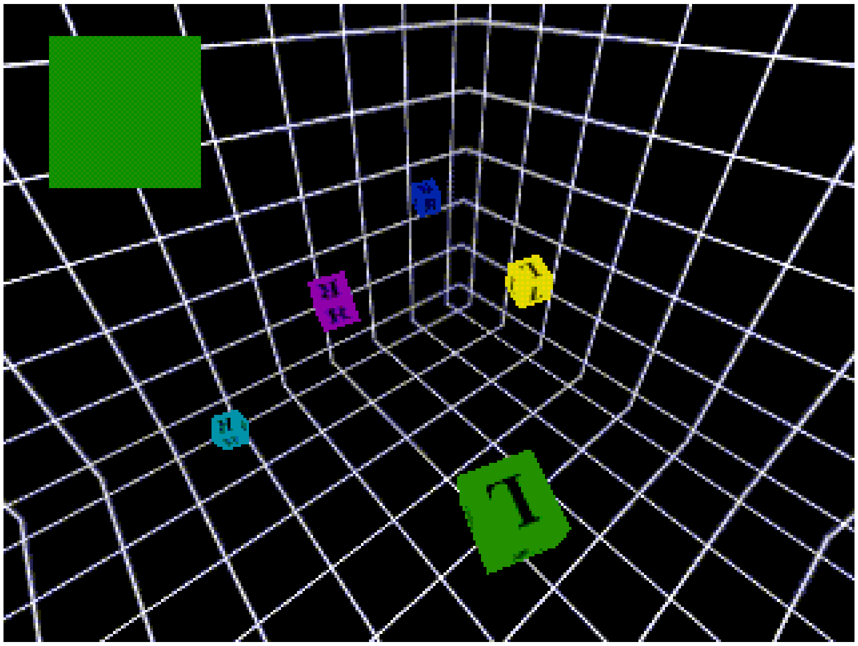


Figure 4.2 Environment for the Spatial Awareness Experiment

Our first experiment focused on one of the more abstract performance metrics: spatial awareness. We were interested in how immersive travel techniques would affect the user's awareness of the three-dimensional environment around him. Specifically, we tested how various velocity and acceleration schemes altered the user's level of spatial awareness.

The virtual environment for this experiment consisted of a set of cubes of contrasting colors, as seen in Figure 4.2. Users learned the locations of the cubes within the space, from both stationary and moving positions. In an experimental trial, the user was taken from the starting location to a new location, then shown a colored stimulus, matching the color of one of the cubes. We measured the user's spatial awareness by the time required to find the cube of that color. The subject proved she had found the correct cube by pressing either the left or right mouse button depending on the letter (“L” or “R”) printed on the cube.

We contrasted four different velocity/acceleration techniques, each of which was system-controlled. The first two techniques used a constant velocity, one quite slow, the other relatively fast. We also implemented and tested a “slow-in, slow-out” technique, in which travel starts and ends slowly, with acceleration and deceleration in between. Finally, we tested an infinite velocity (also called “jumping” or “teleportation”) technique, where users are taken immediately to the target location.

The results of the experiment showed that the level of spatial awareness was significantly decreased with the use of a jumping technique ($p < 0.01$). In fact, users were generally reduced to a simple search of the space after jumping from one location to another. This is a significant result, since many application designers might be tempted to use teleportation because of its speed and accuracy. The experiment shows that this is unwise unless some degree of user disorientation is acceptable in the target application. Surprisingly, none of the other techniques showed significant differences in performance: even up to relatively large velocities, users could maintain spatial awareness.

4.4.2 Absolute Motion Experiment

In the second experiment, we wanted to obtain some basic information about the speed and accuracy of two common steering techniques: gaze-directed steering, in which the direction of motion is determined by the user's gaze, and pointing, in which the user's hand orientation determines the direction of travel. Even though speed and accuracy are not always the most important considerations in a travel technique, they are still widely desirable. Once a target has been chosen, it is usually unacceptable to the user to move there slowly or imprecisely. We chose to compare gaze-directed steering with pointing because they seem to be quite different in their focus: gaze-directed steering is simple but constraining, while pointing is expressive but also more complex.

The experimental task was quite simple. Users traveled using one of the techniques from a starting location to a target sphere. We varied the size of the sphere and the distance to the sphere. We hypothesized that gaze-directed steering might produce greater speed and accuracy than pointing, because of its simplicity and the relative stability of the head compared to the hand.

Although gaze-directed steering did produce slightly better times for this task, we found that there was no statistically significant difference between the two techniques. Users were able to travel very close to the optimal straight-line path between the starting and target locations whether gaze-directed steering or pointing was used. This was useful information given the advantages of pointing shown by our next experiment.

4.4.3 Relative Motion Experiment

Rather than moving directly to an object in the environment, in this experiment users were required to move to a point relative to an object in the 3D space. This task is commonly used in applications such as architectural walkthrough. For example, suppose the user wishes to obtain a head-on view of a building so that it fills his field of view. There is no specific target object; rather, the user is moving relative to the building. In this experiment, the target was located on a line defined by a three-dimensional pointer, at a known distance from the tip of the pointer. Figure 4.3 shows the pointer and the target, although the target was not visible during experimental trials.

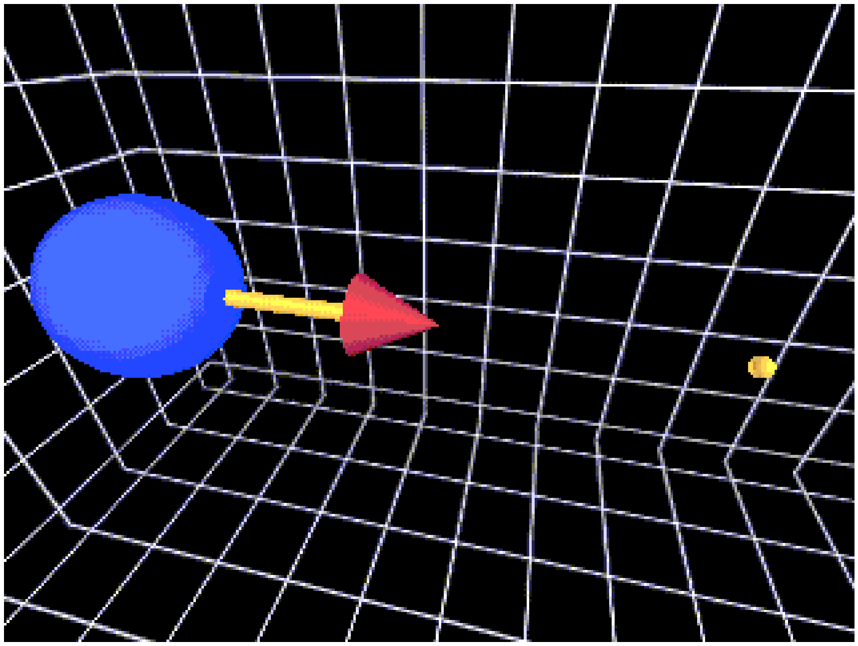


Figure 4.3 The Relative Motion Experiment Environment

Again, we measured speed and accuracy for the gaze-directed steering and pointing techniques. With this task, however, we highlighted the main difference between the two techniques: that gaze-directed steering requires the user to be looking in the direction of motion, while pointing allows gaze and travel to be in different directions. Thus, users of the pointing technique could look at the pointer to judge their travel to the target location, while gaze-directed steering required users to look at the pointer, then look in the estimated target direction to travel, then look back to check their progress, and so on.

Indeed, the experiment showed that the pointing technique was significantly faster for the relative motion task ($p < 0.025$). When combined with results from the absolute motion experiment, we can conclude that pointing is a good general-purpose technique where speed and accuracy are important performance measures.

4.5 Expanded Evaluation Framework

Although our initial set of experiments produced significant results in evaluations of some common VE travel techniques, we also noted that we were not able to capture a complete picture of the techniques from simple experimental designs. The problem was that our experiments studied the effects of a single factor only (travel technique), and did not consider other factors that might have an important effect on performance.

This is illustrated well by the absolute and relative motion experiments. Though they tested the same techniques (gaze-directed steering and pointing) and measured the same performance variables (speed and accuracy), they produced quite different results. In the case of absolute motion, the two techniques performed equally, but for a relative motion task, pointing showed more speed and accuracy. There was, therefore, an interaction between technique and task. This illustrates the fact that a technique cannot in general be considered in isolation from the task for which it is to be used.

Similarly, characteristics of the environment may affect the performance of a travel technique. Consider the absolute motion experiment. In the environment that we used, there was only a single object (the target), visible at all times, with a straight-line path between it and the user. In this environment, gaze-directed steering and pointing produced the same results. However, if the environment had been full of distracter objects and obstacles that the user had to avoid to reach the target, the two techniques might have exhibited significantly different performance characteristics. Techniques cannot be considered in isolation from the environments in which they are to be used.

For these reasons, we felt it necessary to expand our evaluation framework to include the multitude of other factors that can affect performance of virtual travel techniques. Rather than attempting to discern these dependencies in an ad hoc fashion for each experiment that is run, our expanded framework formalizes the notion that many variables contribute to the performance metrics. By explicitly including these variables in the framework, we can more easily choose what factors to control in an experimental setting, and choose values wisely for those variables which will be held constant. The expanded framework includes variables related to task, environment, user, and system characteristics.

4.5.1 Task Characteristics

For immersive travel, there are many factors related to the task that could conceivably affect performance. Some of these characteristics come directly from a consideration of the performance values that we wish to measure. Some of the task characteristics that we consider are:

- Distance to be traveled
- Amount of curvature or number of turns in the path
- Visibility of target
- Number of degrees of freedom of motion required
- Accuracy required
- Complexity of the task; cognitive load induced on the user
- Information required of the user

For example, we could distinguish between the absolute and relative motion tasks described above using the visibility characteristic. The target is invisible in the relative motion task, meaning that other objects in the environment must be used to determine the location of the target.

4.5.2 Environment Characteristics

As we have noted, the environment in which the user travels can also have an effect on performance. The same task in different environments may produce strikingly different results on one or more of the performance measurements. We have identified characteristics such as:

- Visibility within the environment
- Number of obstacles or distracters
- Activity or motion within the environment
- Size of the environment
- Level of visual detail and fidelity
- Homogeneity (amount of variation) in the environment
- Structure
- Alignment with the standard axes

Varying one or more of these environment variables may have allowed us to see some significant differences between the gaze-directed steering and pointing techniques in the absolute motion experiment. For example, adding more distracter objects or greater activity in the environment may have caused the more cognitively simple gaze-directed steering technique to perform better.

4.5.3 User Characteristics

It is also important to consider the differences in users of VE applications when evaluating performance. This can be a significant factor in the performance of various techniques, because the designers of techniques often assume something implicitly about users. Work in the field of user modeling (Kobsa & Wahlster, 1989) is quite relevant to this part of our framework. We are considering, among others, the following user characteristics:

- Age
- Gender
- Visual acuity
- Height
- Reach
- Ability to fuse stereo images
- Experience with VEs
- Experience with computers
- Technical / non-technical background
- Spatial ability

The importance of taking user characteristics into account became quite evident during a study we performed comparing various techniques for selecting and manipulating virtual objects (Bowman & Hodges, 1997). Our implementation of one technique (Poupyrev et al., 1996) mapped the user's physical arm extension to a more lengthy virtual arm extension, so that the number of objects that could be selected depended on the user's reach.. In the user study, most people liked this technique, but a few of our users had very short arms, and could not reach many of the objects at all. This caused them to become quite frustrated with this technique and to prefer other techniques that did not rely on physical arm length.

4.5.4 System Characteristics

Finally, we have extended our framework to include aspects of the hardware or software used to realize the virtual environment application. It is quite possible that design decisions made by system developers or hardware designers may affect the performance of techniques for virtual travel. However, just because these factors are not always under the control of the technique designer does not mean that they should not be considered in the design. For best performance, designers may need to create techniques that perform in a robust manner under a wide variety of system conditions. The system characteristics we have identified include:

- Rendering technique
- Lighting model
- Frame rate
- Latency
- Display characteristics (stereo/mono, field of view, resolution, brightness, etc.)
- Collision detection
- Virtual body representation

These factors can cause differences in the usefulness of many interaction techniques. Studies on the effects of varying frame rate and latency for various tasks have been performed (e.g. Ware & Balakrishnan, 1994), but there is still much work to be done.

4.5.5 Information Gathering Experiment

In order to validate our evaluation methodology, we designed and ran a new experiment within our expanded framework. We hoped to isolate some important and general results, and to show the usefulness of considering a larger number of experimental variables simultaneously.

Our focus was the effect of various steering techniques on the performance metric of information gathering. Information gathering is an important goal in many situations, and it is especially applicable to immersive virtual environments. Many of the major categories of VE applications, such as architectural walkthrough (e.g. Brooks, 1992), information visualization (e.g. Ingram & Benford, 1995, Bolter et al., 1995), simulation and training (e.g. Tate, Sibert, & King, 1997), and education (e.g. Allison et al., 1997), have a strong informational component. If the user is not able, for whatever reason, to focus on and remember important information, then the utility of the VE application is questionable.

There are many possible reasons why a user might not be able to gather as much information as is desirable, but a major factor is cognitive load. A famous result from cognitive psychology (Miller, 1956) shows the severe limitations on the capacity of working memory. When other influences force the person to use part of his working memory or other cognitive resources, information may be lost, or displaced (Baddeley, 1983). We wondered whether travel techniques induced cognitive load, and could therefore affect the amount of information that could be recalled by the user.

We chose to focus on the direction selection portion of the taxonomy, and to again study gaze-directed steering and pointing techniques. We also added a third technique, torso-directed steering, in which a tracker is attached to the user’s torso, so that she travels in the direction her body is facing. We felt that these three represented a useful cross-section of commonly used techniques, and that there were some interesting tradeoffs among them.

For example, both pointing and torso-directed steering have the advantage that the user can look in one direction and move in another. This could be important when gathering widely scattered information. However, these techniques are also cognitively more difficult than gaze-directed steering, in which head orientation is the only thing the user must control. Torso-directed steering might be more natural (since it simulates the way we walk) and thus produce less cognitive load than pointing, but it also has the disadvantage that it can only be used to move in a horizontal plane, as the torso cannot comfortably be pointed up or down. We were quite interested to see how these tradeoffs affected a user's ability to gather information.

Looking at our expanded framework, however, we felt that there were several other factors that could influence performance on this task. Therefore, we also chose one environment characteristic and one system characteristic to vary along with the travel technique. First, we felt that the complexity of the path through the environment might be quite important in the cognitive load induced upon a user. We captured this complexity characteristic in the *dimensionality* of the path. That is, some paths would be one-dimensional: straight and horizontal; others would be two-dimensional: still horizontal, but with turns; and still others would be three-dimensional: having turns and also vertical components.

Second, we hypothesized that the presence or absence of a collision detection feature might affect information gathering. If a user is focusing on information and not on the path he is traveling, he may move through a wall or other object. The effort required to move back through the object and back onto the desired path may use cognitive resources and displace information. With collision detection available, the user is kept near to the path, and is free to gather information without paying as much attention to the direction of motion. On the other hand, the use of collision detection may violate the mental model of the user, if the user has been told that he will keep moving as long as a button is pressed, for example. This also may induce cognitive loading. Therefore, we were interested to see how the use of collision detection would affect performance.

4.5.5.1 Method

To measure the user's ability to gather information, we decided to use a memory task. Subjects traveled through corridors, using one of the three steering techniques. Corridors were used so that the user would have only a single, directed path through the environment, with no choices as to which path to take. The experiment used one-, two-, and three-dimensional corridors, 3x3 meters in size, made up of straight segments, and employing only 90 degree turns. An outside view of a three-dimensional corridor is seen in figure 4.4. Signs, each containing a single word, were located on the walls, ceilings, and floors of the corridors, as seen in figure 4.5. The words used were common, short, non-proper nouns and were randomly scattered through the corridor. Each corridor contained 12 signs. Subjects were instructed to minimize the amount of time spent in the corridor (the maximum time was 60 seconds, but a trial also ended if the subject reached the end of the corridor), and also maximize the number of words and locations of words that they could remember.

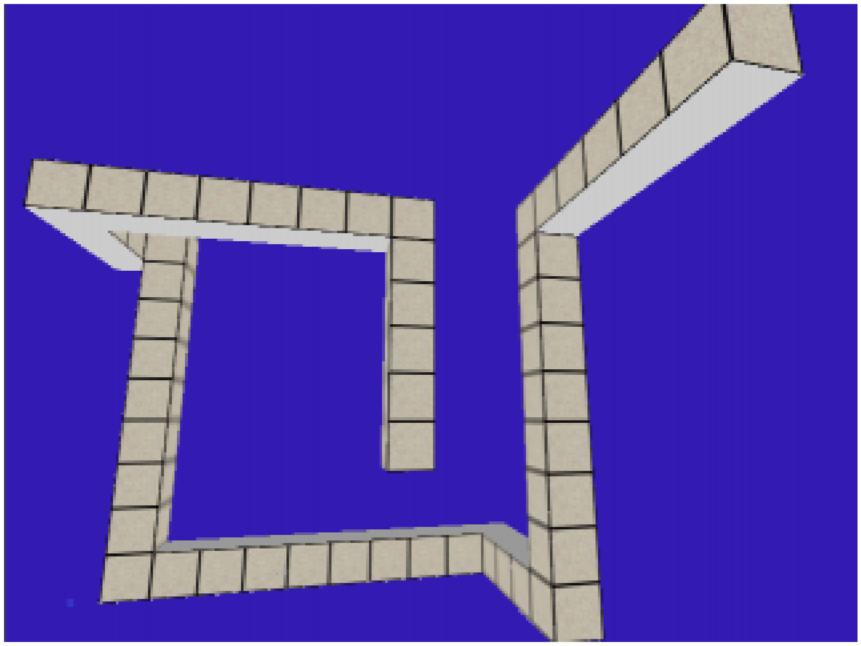


Figure 4.4 Outside View of a Three-Dimensional Corridor

Thus, we presented subjects with a very difficult, memory-overloading task. It has been shown that the limit of working memory is generally seven plus or minus two chunks of information (Miller, 1956), and we were presenting 12 words and associated sign locations to the subject. Even if subjects could store both the word and location as a single chunk, and even if some words could be chunked together semantically or in some other way, the amount of information should still fill working memory. Therefore, if cognitive load is induced because of the travel technique, the dimensionality of the corridor, or the presence or absence of collision detection, we should observe that the amount of remembered information should decrease.

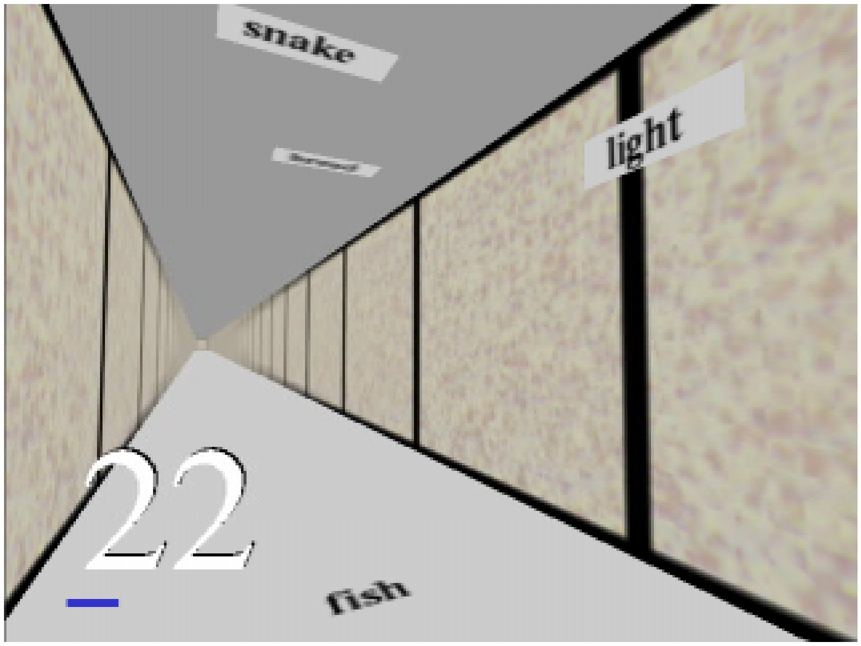


Figure 4.5 Interior of a Corridor from the Information Gathering Experiment

In order to demonstrate their memory of the corridor, subjects indicated words and locations on a paper map of the corridor immediately after each trial. An example map is shown in figure 4.6. Subjects indicated the position of the sign along the corridor, the surface on which the sign was seen, and the word printed on the sign. If words were remembered without locations, or vice-versa, these could also be listed on the map.

For each of the steering techniques, the other two components of a complete travel technique were held constant. Velocity was 3.0 meters per second while traveling; subjects began travel by pressing and holding a button, and stopped by releasing the button.

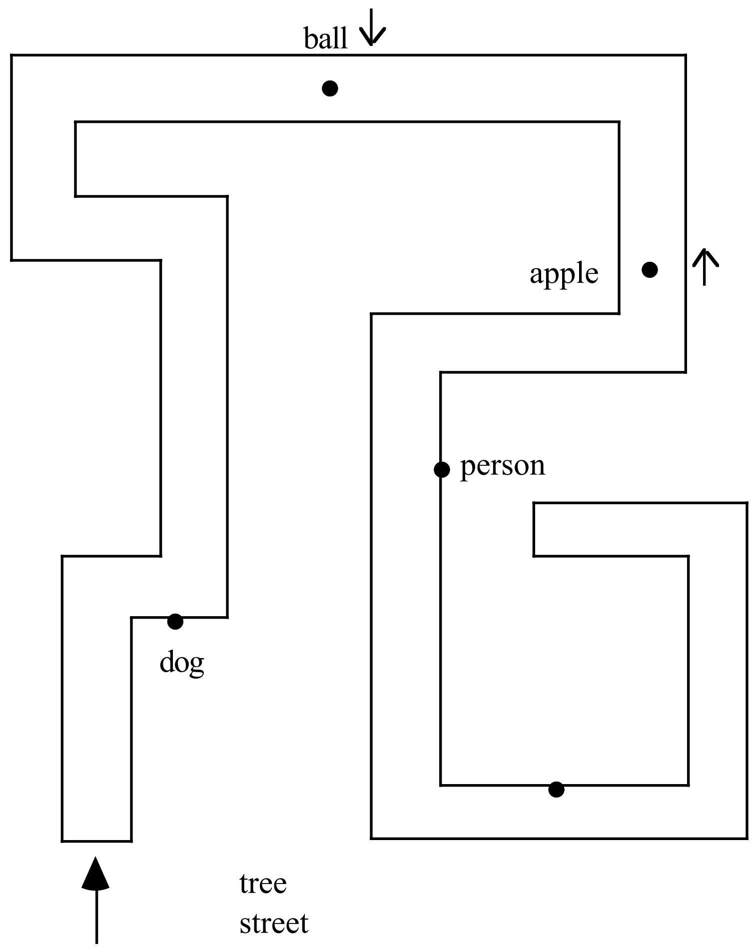


Figure 4.6 Example Completed Corridor Map with Four Word/Location Pairs, One Unpaired Location, and Two Unpaired Words

Each subject completed 16 trials: six each with the gaze-directed steering and pointing techniques, and four with the torso-directed steering technique. Within each technique, there were two trials of each dimensionality (the torso-directed technique can only be used in one- and two-dimensional environments), where one of the trials used collision detection and the other trial did not. Thus, each combination of the three variables (steering technique, dimensionality, and collision detection) was encountered once by each participant. Each subject traveled through each corridor exactly once, and the order of the corridors was different for each subject. To be less confusing for the subject, trials using a given technique were grouped together; however, we counterbalanced the order in which the techniques were seen. To eliminate effects of learning the techniques, subjects spent time in a “practice room” before each set of trials, where they practiced the use of the next steering technique.

Twenty-six student volunteers (twenty-three males and three females) participated in the study. Two subjects quit the experiment before completion due to dizziness or nausea induced by the VE system. Each subject completed a pre-session questionnaire (Appendix A) in which we gathered demographic data such as age, gender, eyesight, technical background, computer knowledge, and experience with immersive VEs. Subjects wore a Virtual Research VR4 head-mounted display (HMD), and were tracked using Polhemus Fastrak or Isotrak II electromagnetic trackers. Input was given to the system with a three-button joystick. The system maintained a constant rate of 30 frames per second.

4.5.5.2 Results

The experiment measured various response variables related to the information gathering task. We measured the time spent in each corridor, the number of word/location pairs the subject got exactly right, and several variations of partially correct words and locations. Since we had instructed subjects to maximize several things simultaneously, we desired a single response variable that would encompass all of these values. The formula used for this overall score was: $1/3\ (60\text{-}t) + 3a + 2\ (b+c+d) + e + f + g$, where t=seconds spent in the corridor, a=number of word/location pairs exactly correct, b, c, and d represent responses that have two of three aspects (word, position, and surface) correct, and e, f, and g are responses where only one of the aspects are correct. This formula gives higher weight to the most correct responses, and rewards moving quickly through the corridor.

Using this metric as our response variable, we performed a three-factor analysis of variance (ANOVA). Results were quite clear: the dimensionality of the corridor was extremely significant in affecting the score ($p < 0.01$), but travel technique and collision detection did not have a significant effect. Further analysis using Duncan's test for comparison of means showed that the average score for each dimensionality was significantly different than the averages for the other two dimensionalities. Table 4.1 presents the average scores for each condition.

Table 4.1 Average Values of Overall Score for Each Tested Treatment Combination in the Information Gathering Experiment; Higher Scores are Better

	1-Dimensional		2-Dimensional		3-Dimensional	
	Collision Off	Collision On	Collision Off	Collision On	Collision Off	Collision On
Gaze-directed	16.90	16.51	11.85	11.21	10.21	9.57
Pointing	15.57	16.68	10.36	10.85	9.33	9.38
Torso-directed	15.50	15.92	10.63	12.15	N/A	N/A

We also performed further analysis of the data in order to find other relationships between our three independent variables and performance of the information gathering task. First, we wondered whether any learning was occurring during the trials themselves. We plotted learning curves for each of the orderings of techniques (necessary since the number of trials depended on the technique), and found no significant improvement in score over time for any of the orderings, implying that neither the use of the technique nor the task strategy changed much as the trials progressed.

Second, we also performed a three-factor ANOVA with total time per trial as the response variable, in order to see which variables had an effect on the speed with which users moved through the corridors. The results here were synonymous with the previous

ANOVA: dimensionality was the only significant factor ($p < 0.01$). Thus, as the dimensionality of the path increased, time spent in the corridor increased. Most subjects finished the one-dimensional corridors quickly, while two- and three-dimensional corridors often took the entire 60 seconds.

Finally, we examined the demographic data collected in the questionnaire for any trends relating this information to performance of the information gathering task. There was a fairly even split between those who had never experienced immersive VEs (16 subjects) and those who had used a VE system previously (10 subjects). Among those who completed the experiment, the more experienced participants had a slightly higher average score per trial (13.2 vs. 11.5). This is not a statistically significant result, but may show that users with even a single experience using a VE application were more focused on the task and not distracted by the technology itself or the feel of the system.

4.5.5.3 Discussion

The results of the information gathering experiment were somewhat surprising, as we had expected that different steering techniques would produce different levels of cognitive load, and thus significantly affect overall scores. We found, though, that path dimensionality was the only significant variable, and that it dominated the results. Our intution regarding the techniques was not sufficient to predict the results (hypothesis 1) However, this does not mean that we learned nothing about the nature of the travel techniques in question.

On the contrary, we noted many important characteristics of the various techniques that help us to explain the lack of significant differences from the experiment. First of all, as we noted previously in our absolute motion experiment (Bowman, Koller, & Hodges, 1997), novice users tend to emulate gaze-directed steering with pointing (by keeping their hands pointed in the direction of their gaze) unless there are large rewards for doing otherwise (as in the relative motion experiment). We saw this again in the current experiment, and also noted the same characteristic with the torso-directed steering technique. This fact quite possibly led to the lack of significant differences between the techniques. We hypothesize that users more familiar with the techniques would be able to use them more advantageously (e.g. look to the side as you move forward with the pointing or torso-directed steering techniques). Given enough expert users of the techniques, it would be interesting to include the experience level of users as another independent variable.

Also, as we stated at the beginning of this section, each technique contains certain tradeoffs. Intuitively, gaze-directed steering should produce the least cognitive load of the three techniques. However, it also provides fewer affordances for information gathering (one must stop moving in order to look to the side for information). The opposite is true of pointing: it should be more cognitively complex but should better afford information collection. Since we have only one measure of information gathering ability, these tradeoffs may have balanced out, producing no visible differences between techniques. In order to further examine these tradeoffs, we would need experimental tasks that test the limits of both sides.

This experiment also showed the usefulness of our evaluation framework. Before the experiment began, it was not clear what factors would lead to significant performance differences. However, because of our expanded framework, we were able to identify three different factors which we felt could be important in an information gathering task. Had we considered only travel techniques in isolation, this experiment might not have revealed any significant results. Because we varied several factors, however, we were able to identify a characteristic with an extremely significant effect on performance.

We found no statistically significant information about the effects of the use of the collision detection feature. However, several subjects did comment to the experimenter that they felt that it was easier to move through the space and perform the task when this feature was enabled. This in itself should encourage designers to include this characteristic in their systems.

Finally, we observed that our subjects had several different strategies for performing the experimental task. Some focused on time, and raced through the environment as quickly as possible, memorizing a few words and locations along the way. Others were much more deliberate, stopping at each sign or cluster of signs to try to commit them to memory. Still others developed hybrid schemes. Subjects also differed in what they attempted to remember. Some consistently recalled the first three or four words and locations (the primacy effect), while others focused on the last things they saw in the corridor (the recency effect). A third group simply wrote down as many words as they could, then tried to match them to locations on the map.

All of these dissimilar strategies may have affected our ability to get significant results. We could have imposed a strategy on the user by instructing them explicitly to perform the task a particular way, and perhaps seen less variability. On the other hand, users will also have differing methods in real applications, and we should be searching for interaction techniques which perform in a robust manner under a variety of strategies. In this sense, it is correct to allow the user flexibility in determining the most appropriate tactic for the task at hand. Formal evaluation of the effects of user strategy proved important in a later experiment.

4.6 Alternate Evaluation Framework

4.6.1 Taxonomy

The expanded framework for design and evaluation of travel techniques provides a great deal of power in explaining performance differences. However, we do not feel that our initial taxonomy of techniques is as complete and general as it should be. One of the reasons for this is the “force-fit” of techniques for quite different travel tasks into the same category. Therefore, we have developed an alternate taxonomy (figure 4.7) based on a very simple task analysis.

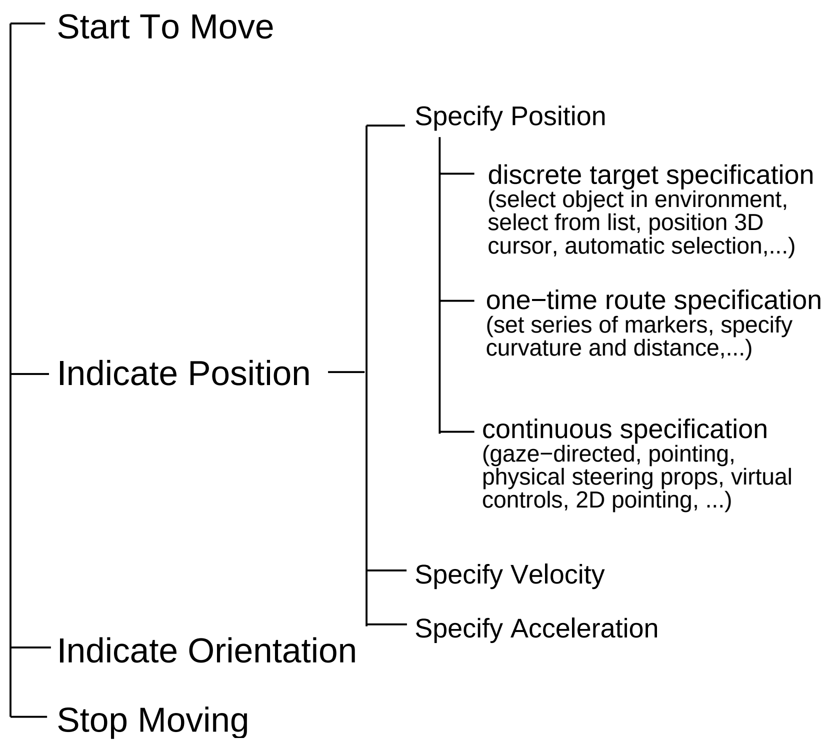


Figure 4.7 Alternate Taxonomy for Travel Techniques with Detail on Position Indication Subtask

We recognize that the task of viewpoint motion control consists mainly of two parts: setting viewpoint position and setting viewpoint orientation. Within each of these parts, there are two quite different methods one might employ. One might specify the destination or target position/orientation. This is a discrete, one-time selection. On the other hand, one might not have a target in mind at all, and might wish to simply specify a continuous trajectory for the position and/or orientation of the viewpoint. In between these two extremes are techniques we call "one-time route specification," which allow the user to specify not only the endpoints of the motion, but also intermediate points or an entire path. The path specification is done prior to any actual movement of the viewpoint.

By separating these strategies in our task analysis, we can more closely and accurately fit the various technique components from our original taxonomy. This ensures that we compare techniques with tasks for which they are suitable, and that the design of new techniques using the framework will follow a logical progression.

4.6.2 Guided Design

This new taxonomy inspired the design of two new interaction techniques for travel, showing the usefulness of the concept of guided design (designing based on a taxonomy and not intuition alone).

First, we noticed in our review of published travel techniques that almost all of them fall into the continuous steering category, with a few discrete target selection techniques as well. However, the one-time route specification category was not represented. Therefore, we developed a simple route-planning technique which we tested experimentally against the other two metaphors (section 4.6.3). With our technique, the user holds a three-dimensional scale model of the environment, and places markers in this model using a stylus (figure 4.8). When a button is pressed, the system does a simple interpolation and takes the user along a piecewise linear path connecting these markers in the full-scale environment . Such a technique has the potential to be a good compromise between the amount of user control over travel and the amount of cognitive load placed on the user while moving, since all route specification is done prior to movement.

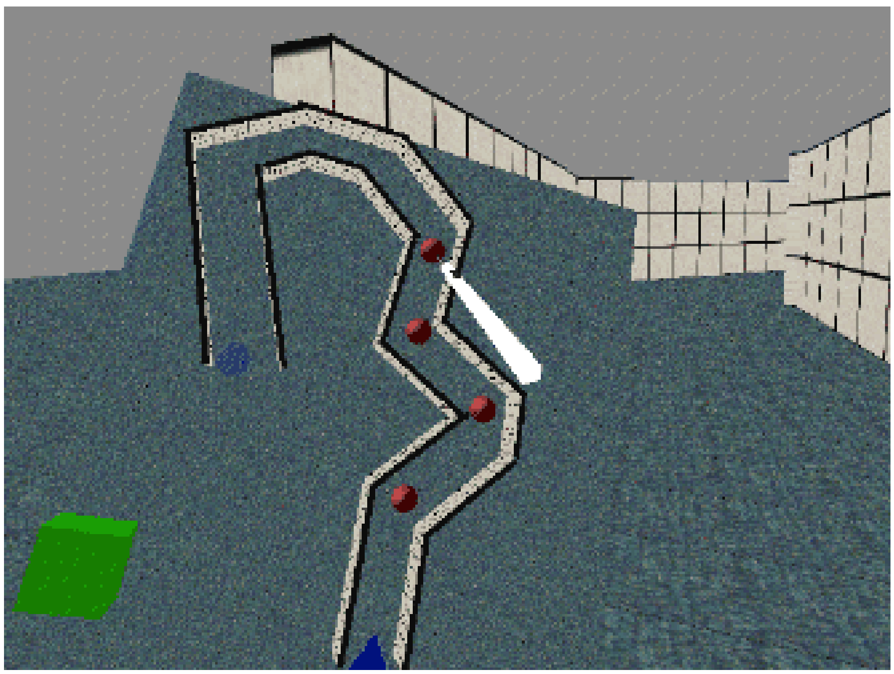


Figure 4.8 Route-planning Technique Using Virtual Map and Stylus

Second, the taxonomy shed new light on the continuous steering metaphor. Most of the existing techniques let the user specify a direction, and potentially a velocity (as in our earlier taxonomy). However, the subtask in the new taxonomy is “specify position,” which implies that viewpoint motion can also be thought of as a manipulation task. It is a simple step from this realization to a large set of potential travel techniques which are based on object manipulation techniques (see chapter 5). This is not a completely new idea (Ware & Osborne, 1990, Pierce et al, 1997), but is made more precise because of the taxonomy. We implemented a travel technique based on the HOMER object manipulation technique (section 5.3.3), in which the selected object is used as a pivot point for viewpoint movement.

4.6.3 Spatial Orientation Experiment

The new taxonomy also lent itself to a new evaluation of techniques. Using one technique from each of the three metaphors discussed above (steering, discrete target selection, route planning), we conducted a test to see which one would produce the highest levels of spatial orientation in users during and after travel. Spatial orientation refers to the knowledge that people have of their own location and orientation (direction) within a space.

Chance et al (1998) found that using a physical (walking) translation technique produced better spatial orientation, although the absolute error measurements they report were still relatively high. As in Chance et al, we chose to use a maze traversal task followed by a pointing task to measure spatial orientation. The mazes were actually corridors – they presented no choice points – and each contained three easily recognizable objects (Figure 4.9). At the end of each corridor, the subject was "virtually blindfolded" (the corridor and objects disappeared from view), and asked to point in the direction of one of these objects. The response variable was the angular error, in degrees, for this pointing task.

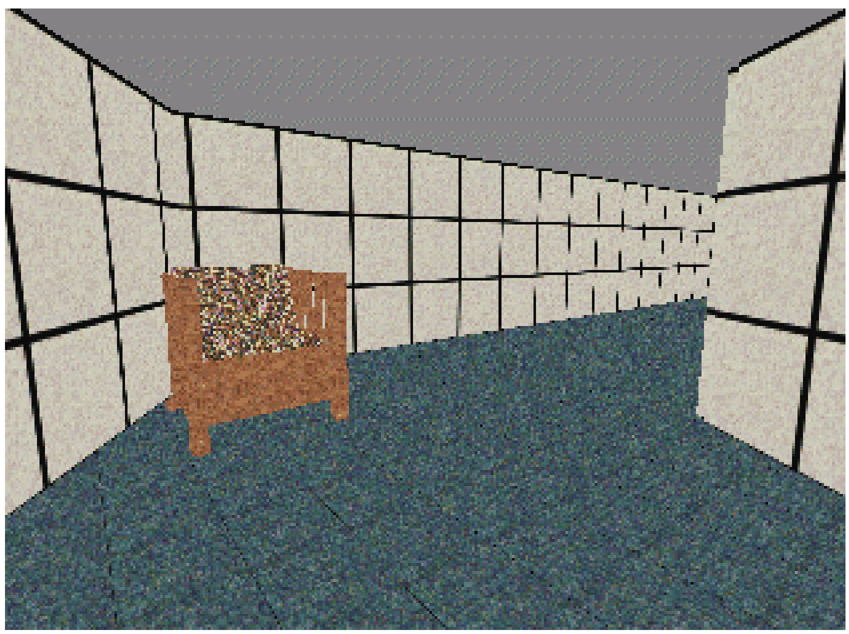


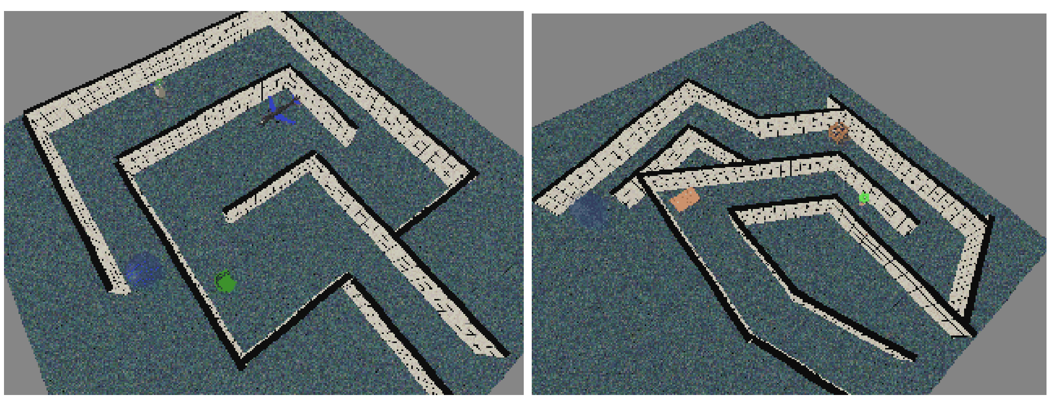
Figure 4.9 Inside View of a Corridor with a Target Object

The experiment, then, required users to pay close attention to the environment through which they were moving. The task might be performed using only route knowledge, along with the positions of the objects along the route, but survey knowledge of the corridor would make the task much easier. In order to maximize subjects' chances to acquire knowledge about the environment, we did not place any time restrictions on the corridor traversal, but rather allowed subjects to stop at any point and take as much time as needed.

The travel techniques were chosen as representatives of the three types of position-specification techniques from our new taxonomy. First, the *system-automated* technique gave users no control over their path. The system simply moved the subjects from the beginning of the corridor to its end on a route approximately in the center of the corridor at all times. The *pointing* technique allows users to continuously specify the direction of motion. Users point in the desired direction of motion. Finally, we chose the *route-planning* technique, in which users set a path before moving, and then are moved along that path by the system.

These three techniques represent different levels of user vs. system control of motion, and we hoped to discover which metaphor produces the highest level of spatial orientation. The extreme techniques are analogous to a driving example: the pointing technique lets the user "drive," while the system-automated technique simply makes the user a "passenger." The route-planning technique represents a compromise between the two.

We also included other variables that could potentially affect spatial orientation. Two factors relate to the complexity of the environment. As in the information-gathering experiment, we varied the dimensionality of the corridor (two or three dimensions). Two-dimensional corridors replicate the experience of moving through building hallways, while three-dimensional corridors also require ascending and descending. Second, some corridors had only ninety-degree (right angle) turns, while others turned at arbitrary angles. See Figure 4.10 for examples of the four corridor types. Finally, we examined conditions in which a three-dimensional map of the corridor was given to subjects before traversal versus trials with no map available*.



* We initially included a fourth factor: the presence or absence of a velocity control feature with which the user could speed up or slow down his rate of travel. Pilot testing, however, indicated that this factor was insignificant for the task, and it was dropped from the experiment.

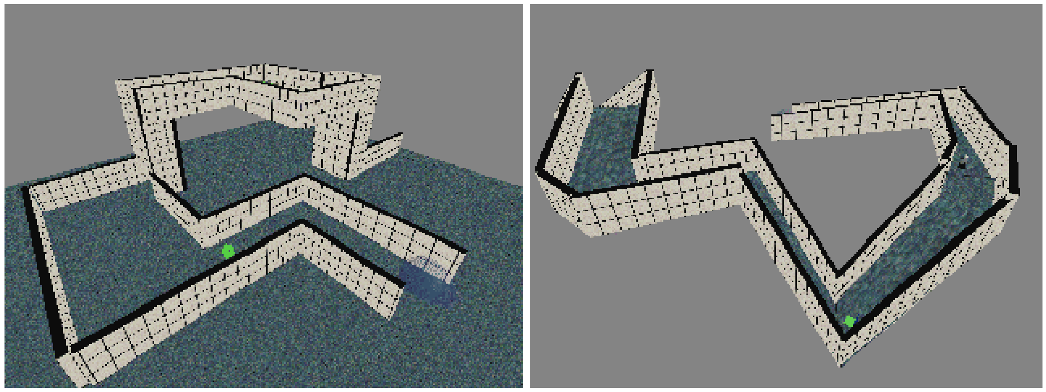


Figure 4.10 Views of Four Corridor Types used in the Spatial Orientation Experiment: Top left: 2D, right angles, top right: 2D, non-right angles, bottom left: 3D, right angles, bottom right: 3D, non-right angles

4.6.3.1 Method

The subjects for the experiment were 29 university students (23 males, 6 females), ranging in age from 18 to 24, with a mean age of 21.14. Eleven subjects reported some previous experience with immersive VEs. Subjects received extra credit in psychology or computer science classes for their participation. Three additional subjects did not complete the experiment due to simulator sickness.

Each subject completed a demographic questionnaire (Appendix A) before taking a standardized spatial ability test (the "cube comparison test" from the Educational Testing Service). This test measures 3D visualization and rotation skills, which are crucial to the experimental task. Before beginning the experiment, subjects were also shown a simple virtual world containing 24 common objects (chairs, tables, lamps, etc.) that would be used as stimuli during the experiment. This served the dual purpose of acquainting users with the head-mounted display (HMD) and tracking system, and introducing them to the objects they would need to know later.

The HMD used was a Virtual Research VR4, with a biocular display (same image to both eyes). The Polhemus Fastrak tracking system tracked the subject's head and two hands. Experimental software was built using the Simple Virtual Environment (SVE) library (Kessler, Kooper, & Hodges, 1998) and ran on a Silicon Graphics (SGI) Indigo2 MaxImpact at a near-constant frame rate of 25 frames per second.

Subjects also completed a set of preliminary VE tests designed to provide a benchmark for their ability to point to object locations in virtual space. In both the benchmark and main experimental tasks, pointing was accomplished using a tracked stylus, which is simply a tracker receiver embedded inside a pen. Users see a virtual representation of the stylus in the VE (see Figure 4.8) that moves in sync with the physical stylus so that the direction of pointing can be visualized. The stylus button is used to record answers. Two other receivers are used – one for head tracking and the other in the user's non-dominant hand where the 3D corridor maps may be viewed.

The benchmark tasks first measured the subjects' ability to point to visible objects in a sparse virtual world (each environment contained a "home" object which users looked at to begin a trial, and three target objects). Users pointed the stylus at one of the targets in response to an aural cue played through headphones. The second set of benchmark tasks required subjects to first study object positions and then turn away. When the stylus button was pressed, the objects disappeared and the subject would be asked to point in the direction of one of the objects. This more closely mimicked the main experimental task, which would require users to point blindly in the direction of a previously seen object. For both types of tasks, we presented trials in which all objects were on the same horizontal plane as the user, and trials in which objects might be anywhere in the 3D space surrounding the user. Subjects completed five trials of each type, for a total of 20 trials.

The main experiment compared the three travel techniques, the two corridor dimensionalities (2D & 3D), the two turn conditions (right angles vs. non-right angles), and the two map conditions (present or absent). All of these were within-subjects variables. There are only 20 valid combinations of these variables, as the route-planning technique uses a 3D map on every trial as a fundamental component of specifying a path through the corridor.

Subjects completed one trial for each valid treatment combination, with all trials using the same travel technique grouped together. Before each group, subjects were allowed to use the travel technique in a practice corridor as long as they wished. The order of travel techniques was counterbalanced between subjects. Within a set of trials using the same travel technique, the order of treatment combinations was randomized.

Corridors were chosen from a set of 16 (four corridors for each combination of the dimension and angle variables). Three objects were placed in each corridor at one of several pre-defined locations within the corridor. Subjects might encounter the same corridor layout more than once during the experiment, but never during the same travel technique group, and never with the same objects or object positions.

On trials where a map was present, the user was allowed to study and manipulate the map using a tracker in his non-dominant hand. Subjects were given as much time as they desired to study the map before movement started. When the subject began moving, the map disappeared.

The pointing technique was implemented using the tracked stylus. Users pressed the button to begin moving, then pointed the stylus in the desired direction of travel. In the system-automated technique, the user simply pressed the button to begin moving down the pre-defined path through the corridor.

For the route-planning technique, subjects used the stylus to place markers on the 3D map of the corridor to define a path (Figure 4.8). The path began at the corridor entrance, then moved in a straight line to the first marker, from there to the second, and so on. The last segment of the path took users from the location of the last marker to the end of the corridor. Subjects began motion by clicking the stylus button while touching a green box on the edge of the map.

While using any of the techniques, users could click the button while moving to stop, then click again to start moving. No collision detection was provided, so that subjects could travel through corridor walls. On two-dimensional corridors, users were constrained to a constant height above the floor, but they were allowed to move anywhere in virtual space while in three-dimensional corridors. Subjects always had complete control of their head orientation and gaze direction, and could look in any direction while moving or stopped.

At the end of each corridor, the visual representation of the corridor was removed, and subjects were presented with an audio stimulus instructing them to point in the direction of one of the objects seen in that corridor. Users estimated the direction by pointing the stylus and recorded their answer by pressing the button. We measured the angular error between the direction pointed and the actual direction to the object. Secondary response variables included time spent in each corridor, the number of times the user stopped in each corridor, and the strategies used to manage the spatial orientation task. We recorded subjects' strategies by observation only. This aspect of the experiment, though exploratory, proved quite interesting, and is discussed in detail below.

4.6.3.2 Results

In this section we present results of statistical analyses on the experimental data. The following section will explain and expand upon each of these results. Analysis of the experiment was split into two full factorial designs, since not all combinations of all factors were valid. Analysis 1 considered two techniques (pointing and system-automated), two dimensionalities, two turn conditions, and two map conditions, for a total of 16 treatment combinations. Analysis 2 considered all three techniques, two dimensionalities, and two turn conditions, with the 3D map always present, for a total of 12 combinations. The map variable could not be used in analysis 2 since a map was available on all trials with the route-planning technique.

We performed a repeated-measures analysis of variance for both of these experimental designs, on both the main dependent variable (angular pointing error) and the secondary dependent variable (time spent in each corridor). Results of both analyses for the error metric are summarized in Figure 4.11. Analysis 1 showed a significant main effect for dimension (mean 2D error: 32.47, mean 3D error: 38.62, $p < 0.005$), and a marginally significant main effect for the map variable (mean map absent error: 33.29, mean map present error: 37.80, $p < 0.075$). No significant differences between travel techniques or the turn conditions were found. Analysis 2 showed a marginally significant main effect for dimension (mean 2D error: 36.012, mean 3D error: 41.254, $p < 0.1$), but no other significant effects.

We performed the same analyses on the amount of time spent by subjects in each corridor. Both analysis 1 and analysis 2 showed significant main effects for technique ($p < 0.075$), dimension ($p < 0.005$), and angle condition ($p < 0.001$). Subjects spent longer amounts of time while using the pointing technique, when in 3D corridors, and when in corridors with right angles only. These results are summarized in Figure 4.12.

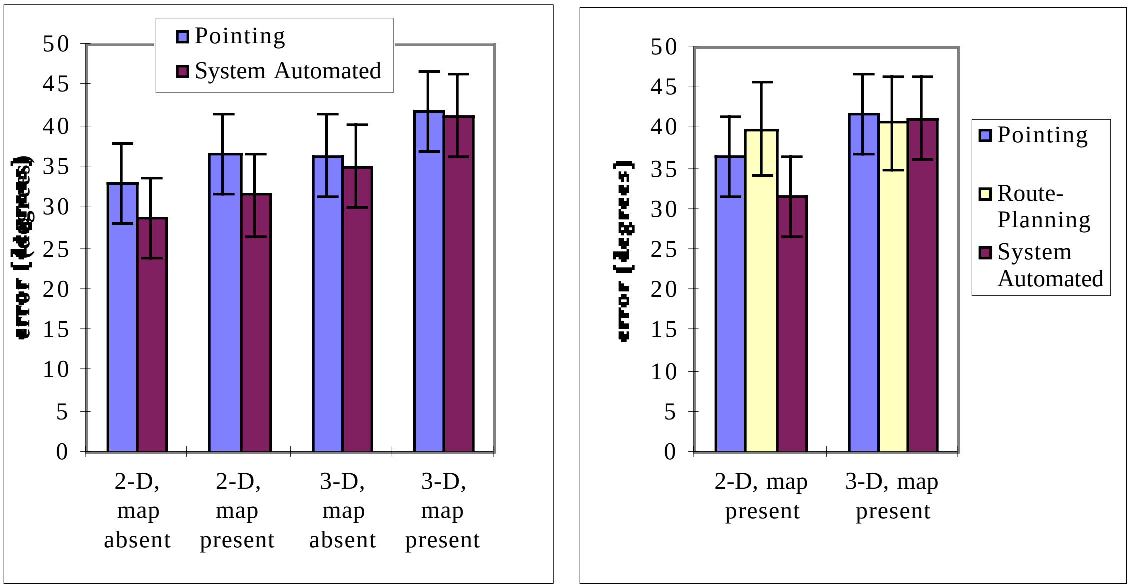


Figure 4.11 Mean Error in Various Treatment Combinations for Analysis 1 (left) and Analysis 2 (right)

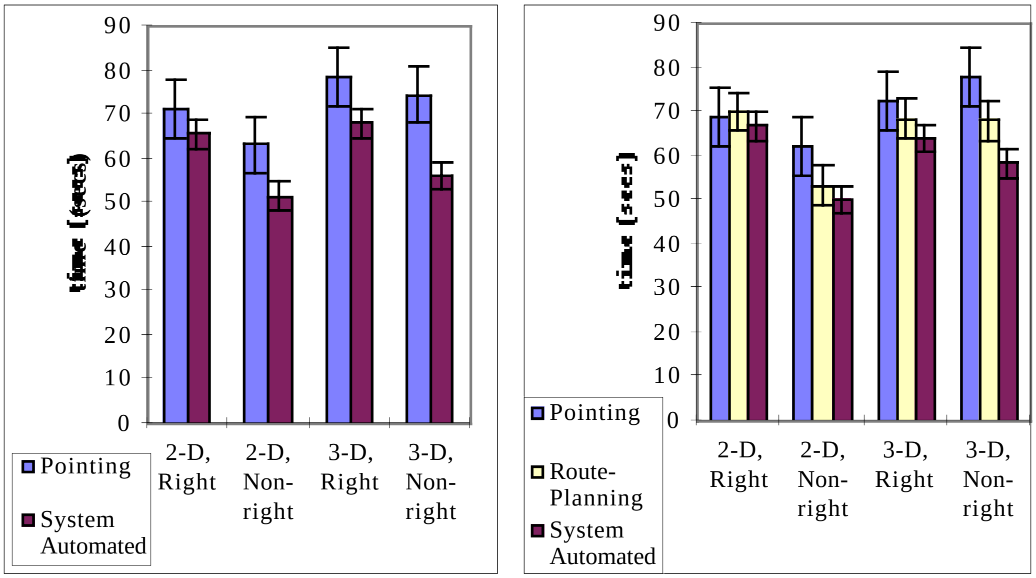


Figure 4.12 Mean Time in Various Treatment Combinations for Analysis 1 (left), and Analysis 2 (right)

We also included user strategies in our analysis of the experimental results. Our subjects were quite creative in their methods for minimizing error in the pointing task. We gave no suggestions to subjects about how to approach the task. We told them only the capabilities of the various travel techniques and that they should do whatever they felt necessary to point accurately at the target objects. We observed six main strategies during the experiment:

- *Stop & Look:* The simplest strategy, in which users simply stop moving at various points within the corridor and turn to look in other directions than the direction of motion.
- *Proprioceptive Pointing*: Users physically point in the directions of objects they have already seen, to give themselves a proprioceptive cue for later recall.
- *Backing in*: Users turn around just before the end of the corridor and move backwards to the end, ensuring that they see the corridor just before it disappears. This is possible with all techniques, but is quite difficult to do properly using the pointing technique.
- *Path retracing*: Users go back along the path they have just traversed through the corridor, both to remind themselves of what they have seen, and to see the corridor from another viewpoint. This strategy cannot be used with the system-automated technique, and requires careful thought to be used with the route-planning technique.

- *Moving through walls*: Users do not follow the corridor exactly, but instead move through corridor walls in order to better understand the relationships between adjacent passageways. A “lawnmower” strategy, in which the user simply travels along parallel lines through the space, is one example. Again, the system-automated technique does not allow this strategy.
- *3D overview*: Users fly up above the corridor to obtain a single view of the complete corridor and the objects within it, which might encourage a survey representation. This strategy is only available on 3D corridors using the pointing or route-planning techniques.

Table 4.2 shows the number of subjects using a strategy with a particular travel technique. One subject did not use any of the six strategies; another used every available strategy (six each for pointing and route-planning, three for system-automated). Subjects averaged 2.5 strategies for the pointing technique, 1.6 for the system-automated technique, and 2.2 for the route-planning technique.

For our analysis of the relation of strategy to error, we defined three between-subjects variables corresponding to the “level” of strategy sophistication for each subject on each technique. For pointing, this was a value between zero and three indicating the number of technique-specific strategies used (3D overview, moving through walls, path retracing). For the system-automated technique, the level was either zero or one indicating whether or not the subject used the backing in strategy. The route-planning strategy level ranged from zero to three indicating the use of the 3D overview, moving through walls, and/or backing in strategies.

Table 4.2 Number of Subjects Observed Using Common Strategies for Each Travel Technique

	Stop & look	Prop. pointing	Backing in	Path retracing	Through walls	3D overview
Pointing	22	12	5	10	7	15
System-automated	22	10	13	0	0	0
Route-planning	21	10	8	2	7	15

With the strategy level variables included in analysis 2, we found a large number of significant interactions indicating that the use of technique-appropriate strategies made a difference in the user’s spatial orientation (error metric). For example, we found a significant ($p < 0.05$) interaction between technique, dimension, and the pointing technique strategy level. Subjects who had a pointing technique strategy level of zero had better scores with the system-automated technique than with the pointing technique, and did equally well on 2D and 3D corridors. Subjects with a pointing technique strategy level of one or two had approximately equal scores using all three travel techniques. Subjects with a pointing technique strategy level of three had better scores using the pointing technique than other techniques, and performed better on 3D corridors than 2D. This interaction suggests that strategy sophistication is significant in determining user performance.

We also analyzed the demographic and spatial ability information that we collected. The spatial ability test has a maximum score of 42, and our subjects averaged 25.862, with scores ranging from 4 to 41. This average is higher than reported means sampled from the general population and from college students. Regression analysis showed that spatial ability score was a significant predictor of average error in the experiment, and average error on each technique, corridor complexity, and map condition. Subjects with higher spatial ability performed better on the pointing task. Previous VE experience did not significantly predict these values. We also found that males performed significantly better than females, but are reluctant to draw conclusions from this due to our low number of female subjects. These results are consistent, however, with prior work. For example, Waller et al (1998) also found a significant effect due to gender in their experiment.

Finally, we analyzed the benchmark tests run on each subject before the main experiment. Both benchmark variables (visibility of objects and location of objects) produced significant differences ($p < 0.001$). Subjects averaged 12.8 degrees of error when objects were visible versus 26.7 degrees when object locations had to be remembered. Trials in which all objects were on the same horizontal plane had an average error of 14.6 degrees, while trials in which objects could appear anywhere had an average error of 24.9 degrees. Regression indicated that the error on trials with visible objects in the same horizontal plane was a significant predictor of error in the main experiment.

4.6.3.3 Discussion

The results presented above confirm that the spatial orientation of a user traveling through an immersive virtual environment depends on the complex interactions of many factors. None of the variables we studied proved solely responsible for the subjects' performance; rather, they all contributed in subtle ways. User strategy played an unexpected role in determining performance. In this section we will revisit and explain each of the major results.

The analyses of the angular error response variable showed that the dimension of the corridor was a significant effect – that subjects performed better on 2D corridors than 3D. Such a result is to be expected since the added complexity of the third dimension makes the corridor layout more difficult to comprehend and remember. Such 3D corridors are not familiar to subjects, but 2D corridors are seen in everyday life. This result also replicates our earlier finding in the information gathering experiment.

Interestingly, we found no significant differences between the performance of the three travel techniques. The system-automated technique produced the lowest average error, but the differences were not statistically significant. The overall mean error was 37.2 degrees for all conditions, which is lower than the mean error for the *best* technique (physical motion) reported in Chance et al (1998). This indicates that virtual travel techniques may indeed allow maintenance of good spatial orientation, although the error values are not directly comparable due to differences in the experiments.

We also found no main effect of the angle condition variable, though we expected that corridors with right angles only would be less complex and therefore produce lower error. However, it is also possible that the non-right angles served as more unique landmarks for subjects, allowing them to visualize their position in a corridor more effectively. These characteristics may have balanced each other so that we saw no significant effect from the angle condition.

In analysis 1 (considering technique, dimension, angle condition, and map), we also found a marginally significant main effect for the map variable, showing that subjects performed better when they were not given a map of the corridor before traveling through it. This result seems counterintuitive, since one would think a map would allow the user to form a better mental representation of the corridor layout. This result could be explained in three ways. First, subjects may have felt that the map gave them an advantage, and therefore did not concentrate as deeply when traveling through the corridor. The map did not show the locations of the three target objects, so subjects needed to integrate the object information with the corridor representation while traveling. Second, the map itself may have been another source of cognitive load, distracting subjects from the task rather than aiding them in it. Finally, we noted several subjects who did not make use of the map or gave it only a cursory glance before beginning through the corridor.

The time response variable also provided some useful information. Subjects using the pointing technique spent significantly longer in each corridor than they did using the other two travel techniques. However, this longer amount of time did not result in any performance gains for the pointing task, indicating that the pointing technique is more complex and requires more effort on the part of the user to maintain spatial orientation. 3D corridors were also shown to require more user time, again proving the difficulty of understanding such corridors. Even with the extra time spent in these corridors, the error was still significantly higher than in 2D corridors. Finally, we showed that subjects spent more time in corridors containing only right angles than in those with non-right angles. This is easily explained due to the fact that the corridors with right angles only were on average 20 meters longer than non-right angle corridors. When times are normalized by corridor distance, time in right-angle corridors is actually slightly lower.

The most interesting results are those pertaining to the strategies subjects used to maximize their performance on the pointing task. They are too numerous to go through one by one, but the example given above illustrates the importance of strategy. Those subjects who used no sophisticated strategies with the pointing technique (such as 3D overview or moving through walls) had better scores using the system-automated technique. However, those subjects with a high level of sophistication for pointing technique strategies actually performed better with pointing than the other two techniques. Furthermore, these subjects reversed the effect of the corridor's dimension by performing better on 3D corridors than 2D.

This gets at the heart of the contrast between the active pointing technique and the passive system-automated technique. Subjects who use the pointing technique naively, to take them directly through the corridor, will experience more cognitive load and thus will perform better with the system-automated technique, where the distraction of choosing a path is absent. On the other hand, subjects who take advantage of the unique characteristics of the pointing technique (the ability to move through walls, the ability to fly in three dimensions, the ability to retrace one's path) give themselves more and better opportunities to comprehend the layout of the space and thus will perform better with the pointing technique. Better performance on 3D corridors for these sophisticated users is explained by the fact that subjects were constrained to a constant height above the floor on 2D corridors, and therefore could not use the powerful 3D overview strategy.

Other significant interactions indicated the importance of the user's strategy when using the other two travel techniques, as well. It is overly simplistic, then, to say that one interaction technique outperforms another, although this may be the case in some situations. In general, though, it is more correct to say that one interaction technique *affords* the user more *opportunities* for high performance levels. Whether or not the user takes advantage of those opportunities is a major factor in determining user performance. For the travel techniques we studied here, the pointing and route-planning techniques give users more control, meaning more opportunities to understand corridor layout and object placement. It is more difficult, in general, to use the sophisticated strategies with the route-planning technique because the entire path must be specified in advance – the user cannot decide halfway along the path that she would like to go somewhere else. Therefore, in cases where subjects were highly sophisticated in all three techniques, performance should be highest using the pointing technique. Indeed, among the two most sophisticated groups of subjects, the average error for pointing was lower (20.32 degrees) than mean error values for the route-planning (21.76 degrees) and system-automated (25.71 degrees) techniques, although these subjects did extremely well using all three techniques.

These results provide two important guidelines (hypothesis 2) for developers of interaction techniques for VE applications. First, the techniques should provide affordances for high performance on an application's main tasks. Second, the users must be trained to take advantage of the opportunities – to use strategies that will help them achieve the desired performance levels. For tasks where spatial orientation is especially important, it appears that a travel technique giving users complete control over their position, such as pointing, can produce high performance levels given that appropriate strategies are used. If it is not possible or practical to train users in those strategies, it may be more beneficial to use a passive travel technique inducing lower cognitive load.

4.7 Travel Testbed

After all of these preliminary experiments, we implemented and ran a testbed evaluation for the task of travel. This testbed is based both on our alternate taxonomy and the formal framework presented earlier, including the lists of outside factors and multiple performance metrics. The testbed is designed to allow experimentation with any travel technique on a wide variety of travel tasks. However, we implemented only two search tasks that were especially relevant to our target application.

These tasks are simple and general, being found in a wide variety of VE applications. Darken (1995) characterizes the two as *naïve search* and *primed search*. Naïve search involves travel to a target whose location within the environment is not known ahead of time. Primed search involves travel to a target which has been visited before – if the user has developed a good cognitive map of the space and is spatially oriented, he should be able to return to the target. We would also like to test *exploration*, in which the user is simply moving about with no specific target, but it would be very difficult to quantify performance on such an open-ended task.

4.7.1 Method

We created a medium-sized environment (one in which there are hidden areas from any viewpoint, and in which travel from one side to the other takes a significant amount of time). The size and type of the environment could be variable if this was deemed an important outside factor on performance, but we left it constant in our implementation. We also built several types of obstacles which could be placed randomly in the environment. These included fences, sheds, and trees (figure 4.13).

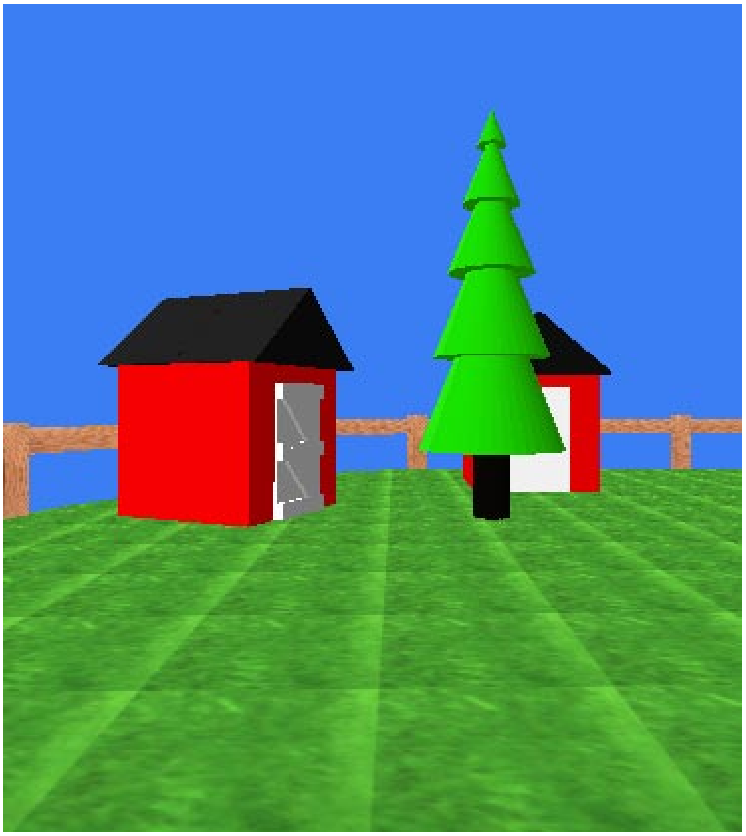


Figure 4.13 Example Obstacles from the Travel Testbed Experimental Environment

Targets for the search tasks were flags mounted on poles. The targets were numbered one to four in each instance of the environment, and each number was associated with a flag color so that the user would be able to identify the targets from a distance. Each target also had a circle painted on the ground around it, indicating the distance within which the user would have to approach to complete the search task (figure 4.14). There were two sizes of this circle: a large one (ten meter radius) corresponding to low required accuracy, and a small one (five meter radius) corresponding to high required accuracy.

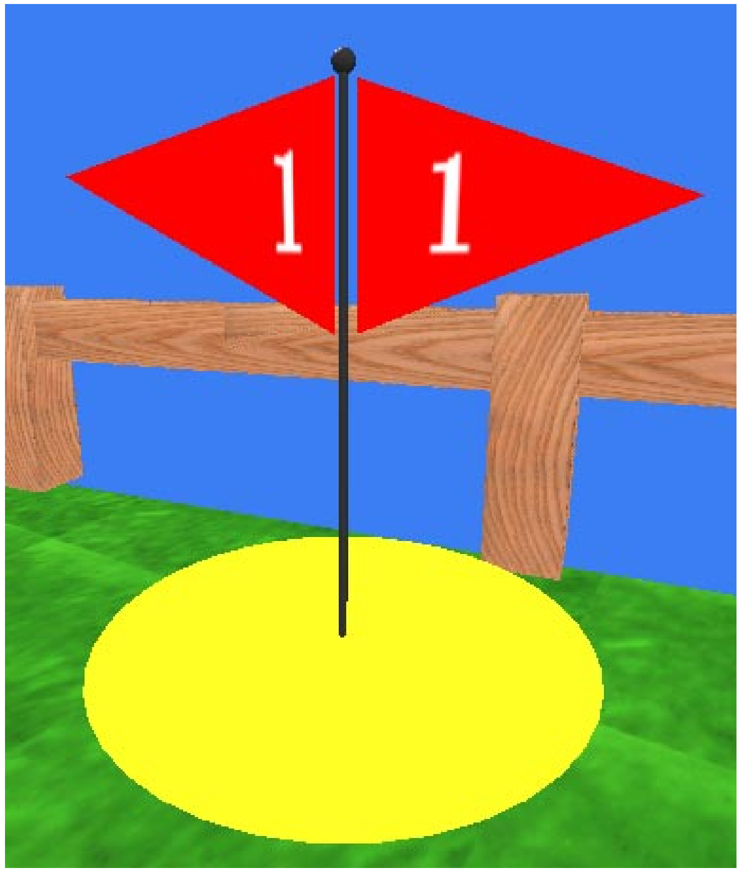


Figure 4.14 Target Object from the Travel Testbed Experimental Environment Including Flag and Required Accuracy Radius

Each subject completed 24 trials – eight trials in each of three instances of the environment. Each environment instance had the same spatial layout, but different numbers and positions of obstacles, and different target positions. In each environment instance, the user first completed four naïve search trials, in which the four targets (numbered one to four) were to be found in order. Before each trial, the flag number and color were presented to the user. During this phase, targets only appeared as they were needed. That is, during the first trial only the first target was visible, during the second trial, targets one and two were visible, and so on. This was to ensure that subjects would not see a target before its trial, thus changing a naïve search to a primed search. The first trial began at a predefined location, and subsequent trials began at the location of the previous target. In each of these trials, the target was not visible from the starting location, and the required accuracy radius was at its low level.

In the second phase, subjects completed four primed search trials where they returned to each of the four targets once, not in numerical order. Again, the flag number/color stimulus was presented to users before each trial. During these trials, all targets were present in the environment at all times, since the subjects had already visited each target and these were therefore primed search trials. Two factors were varied (within-subjects) during these trials. First, we varied whether the target could be seen from the starting position of the trial (visible/invisible). Second, we varied the required accuracy using the radii around each target. Each of these variables had two levels, and therefore there were four possible combinations, and one trial of each of these combinations during each environment instance.

Seven travel techniques were implemented and used in this experiment. Travel technique was a between-subjects variable. Three were steering techniques: pointing, gaze-directed, and torso-directed, as described in the information gathering experiment.

We also implemented two target-specification techniques. In the ray-casting technique, the user pointed a virtual light ray at an object to select it, and then was moved by the system from the current location to that object. The second target-specification technique involved dragging an icon on a two-dimensional map held in the non-dominant hand. The map shows the layout of the environment and an icon indicating the user's position within the environment (figure 4.15, top). Using a stylus, the user can drag this icon to a new location. When the icon is released the user is flown smoothly from the current location to the corresponding new location in the environment. Both the stylus and the map have both physical and virtual representations (figure 4.15, bottom). This technique was one of the travel metaphors used in our target application at the time (section 6.3.2). With both the ray-casting and map techniques, the user could press a button during movement to stop at the current location.

Finally, we studied two manipulation-based travel techniques, as described in section 4.6.2: one based on the HOMER technique (section 5.3.3) and another on the Go-Go technique (section 5.3.1). With the HOMER technique, the user selects an object using ray-casting, then uses hand movements to move the viewpoint around that object. The Go-Go technique uses a non-linear mapping to allow the user to stretch his virtual hand far away from his body. The user clicks a button to "grab the air" at the current location of the virtual hand, and then uses hand motions to move the viewpoint around the environment.

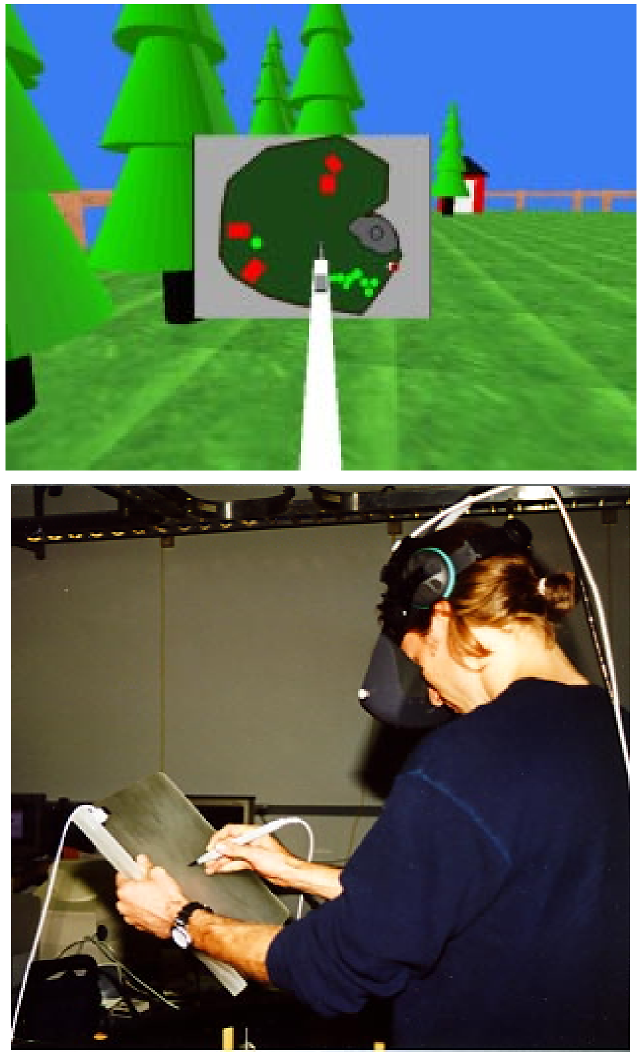


Figure 4.15 Virtual (top) and Physical (bottom) Views of the Map Dragging Travel Technique

For each subject, we measured the total time taken to complete each trial (broken into two parts: the time between the onset of the stimulus and the beginning of movement, and the actual time spent moving). We assumed that this first time would correspond to the time spent in mental processing (perception of the stimulus and environment, and cognitive effort to remember where a target was last seen in the primed search task). This is not entirely accurate, as wayfinding activities undoubtedly continue after a subject's travel has begun. Therefore, the absolute measurements here are not meaningful, but the relative differences between techniques may give some indication of the amount of perceptual/cognitive processing necessary to move to a certain location or in a certain direction using a technique. We have labeled this measure "think time" in the analyses to follow. We also obtained subjective user comfort ratings in the areas of arm strain, hand strain, dizziness, and nausea. After each environment instance, the subjects gave a rating for each of these factors on a ten-point scale (Appendix B). Each subject also took a standardized test of spatial ability (the ETS cube comparison test). Finally, we gathered demographic information about our subjects, including age, gender, handedness, technical ability, and VE experience (questionnaire in Appendix A).

Forty-four undergraduate students were recruited from the department of Psychology subject pool to participate in the experiment. Four subjects did not complete the experiment due to sickness or discomfort, and two subjects did not complete the experiment due to computer problems. Thus, 38 subjects (32 males, 6 females, mean age 19.7) completed the evaluation, meaning that each technique was used by at least five subjects.

4.7.2 Results

In this section, we will present the most relevant results from the travel testbed experiment. For complete tables of results, see Appendix C.

We performed a one-way analysis of variance (ANOVA) on the results for the naïve search task, with travel technique as a between-subjects variable. Table 4.3 gives the results for the naïve search task for each technique.

Table 4.3 Mean Completion Times (seconds) for Naïve Search Task (Standard Deviation in Parentheses)

Technique	Think Time	Travel Time	Total Time
Gaze-directed	2.16 (1.10)	18.28 (4.63)	20.44 (5.24)
Pointing	2.20 (0.92)	22.33 (6.98)	24.53 (7.88)
Torso-directed	2.77 (0.63)	27.00 (6.27)	29.77 (6.49)
HOMER	4.20 (1.00)	37.66 (5.65)	41.86 (6.31)
Map dragging	29.54 (11.58)	52.39 (13.11)	81.93 (18.61)
Ray-casting	1.86 (0.48)	34.95 (8.89)	36.81 (8.43)
Go-Go	3.29 (2.43)	21.48 (6.86)	24.77 (8.73)

For each of the three time measures (think time, travel time, and total time), the travel technique used had a statistically significant effect ($p < 0.001$). We also performed post-hoc comparisons of techniques (LSD and Bonferroni), and found that for the think time measure the map dragging technique was significantly slower than all other techniques. This makes intuitive sense, since the map technique is based on the target-specification metaphor, where movement must be planned before it is carried out. The ray-casting technique also has this property, but selection of a single object is much faster than dragging an icon through an entire route. With the other techniques, movement could begin immediately. However, because the difference is so large, we feel that there may be another factor at work here. The map technique requires users to mentally rotate the map so that it can be related to the larger environment. This mental rotation induces cognitive load on the user, which may cause them to be unsure of the proper direction of movement. The increased cognitive load can be seen directly in increased thinking time.

In the travel time measure, using the same post-hoc tests, we found that the pointing and gaze-directed steering techniques and the Go-Go technique were significantly faster than HOMER, ray-casting, and map dragging. The torso-directed steering technique was significantly faster than HOMER and map dragging. In general, then, steering techniques performed well at this task because of their directness and simplicity. Users could look at the environment, determine where they wanted to search next, and then go there with little or no thought required. The torso-directed technique performs slightly worse, as we found in the information gathering experiment. We believe this is purely a function of mechanics. The user of the torso-directed technique must move his feet and whole body to change directions, while the other steering techniques require only movements of the head or hand. It is also interesting that the Go-Go technique performed well here, but HOMER did not, since they are both manipulation-based travel techniques. The difference seems to be that HOMER requires an object to move about, while the Go-Go technique allows the user to simply grab the air and pull himself forward. Again, the map dragging technique performed poorly. It is simply not suited for exploration and naïve search, because it assumes the user has a distinct target in mind.

Total time results for the naïve search task were almost identical to the results for the travel time measure, since most of the time was spent moving. Again, pointing, gaze-directed steering, and to a lesser degree, Go-Go, performed significantly better than other techniques.

For the primed search task, we performed a multi-variate analysis of variance (MANOVA), with technique as a between-subjects variable and visibility (two levels) and required accuracy (two levels) as within-subjects variables. Travel times were normalized relative to the distance between the starting point and the target (this was not necessary for the naïve search task since subjects in that task had no knowledge of the location of the target and thus did not move in straight lines). Table 4.4 presents a summary of results for this task. We do not list results for the two levels of required accuracy independently, because this factor was not significant in any of our analyses. Results for think time mirrored the naïve search task. Again, technique was significant ($p < 0.001$), with the map dragging technique significantly slower in post-hoc comparisons (LSD and Bonferroni) than all other techniques, for the same reasons given above. Neither of the within-subjects factors was significant in predicting think time.

Technique was also significant for the travel time measure ($p < 0.001$). Here, post-hoc tests showed that pointing and gaze-directed steering were significantly faster than HOMER, ray-casting, and the map technique. Again, these techniques allow the user to form a direct mapping between the desired direction of motion and the action that needs to be taken (look or point in that direction). The map technique performed badly, but it was only significantly worse than gaze-directed steering, pointing, and Go-Go. We had expected that the map would be useful for the primed search, since it allows users to specify the location of the target and not the direction from the current location to the target. However, this assumes that the user understands the layout of the space, and that the technique is precise enough to let the user move exactly to the target. In the experiment, the size of the target was not large enough, even in the low required accuracy condition, to allow precise behavior with the map technique. We observed users moving directly to the area of the target, but then making small adjustments in order to move within the required range of the target. However, the best results with the map did occur in trials with low required accuracy and a target not visible from the starting location. We also found that visibility of the target from the starting location was significant here ($p < 0.001$). Trials in which the target was visible averaged 12 seconds, as opposed to 23 seconds for trials in which the target was hidden.

Total time for the primed search task produced similar results. Again, technique was significant ($p < 0.001$), with the gaze-directed steering and pointing techniques performing best, according to the post-hoc comparisons. Visibility also significantly affected total time ($p < 0.001$). Another technique that we expected to perform well in the primed search task was ray-casting, since it allows the user to move directly to a target. This should especially hold in cases where the target is initially visible. We believe these results were not found due to our implementation of targets as flags. The flagpoles were very thin, and thus impossible to select at any distance. The flags themselves were larger, but due to the size of the environment might appear very small from the starting location. Thus, users of the ray-casting technique often had to select an intermediate target in order to get close enough to select the flag.

Table 4.4 Mean Completion Times (seconds) for Primed Search Task, with Targets not Within View from Start Location (Invisible) or In View from Start Location (Visible)

*(*normalized times – seconds per 100 meters)*

Technique	Invisible think time / Visible think time	Invisible travel time* / Visible travel time*	Invisible total time* / Visible total time*
Gaze-directed	1.69	10.52	12.21
	1.49	4.70	6.18
Pointing	2.30	10.20	12.49
	2.03	5.61	7.63
Torso-directed	2.95	22.87	25.82
	1.40	5.81	7.21
HOMER	3.85	26.34	30.19
	2.67	13.81	16.48
Map dragging	20.58	25.07	45.65
	14.01	18.97	32.98
Ray-casting	2.09	29.69	31.78
	1.92	13.72	15.64
Go-Go	2.66	17.55	20.21
	1.72	7.36	9.09

We also performed an analysis that compared the two types of tasks. For this analysis, technique was again a between-subjects variable, while task was a within-subjects factor. We only considered the trials in which the target was initially visible and the required accuracy was low, since these were the conditions in all of the naïve search trials. For the travel time measure, we found that task was significant ($p < 0.001$), with the naïve search taking 30 seconds on average vs. 23 seconds for the primed search. For the think time measure, task was not significant, but we did find a significant interaction between task and technique ($p < 0.025$). This interaction is due to the fact that the amount of think time for the map technique drops significantly for primed search trials (figure 4.16 – error bars have been omitted in the figure for readability), while think time for the other techniques remains approximately the same. This indicates that subjects had learned the layout of the space and were more confident in the map dragging task because they knew the area in which the target lay. For each of the significant results reported above, the observed statistical power was 0.987 or greater, with alpha = 0.05.

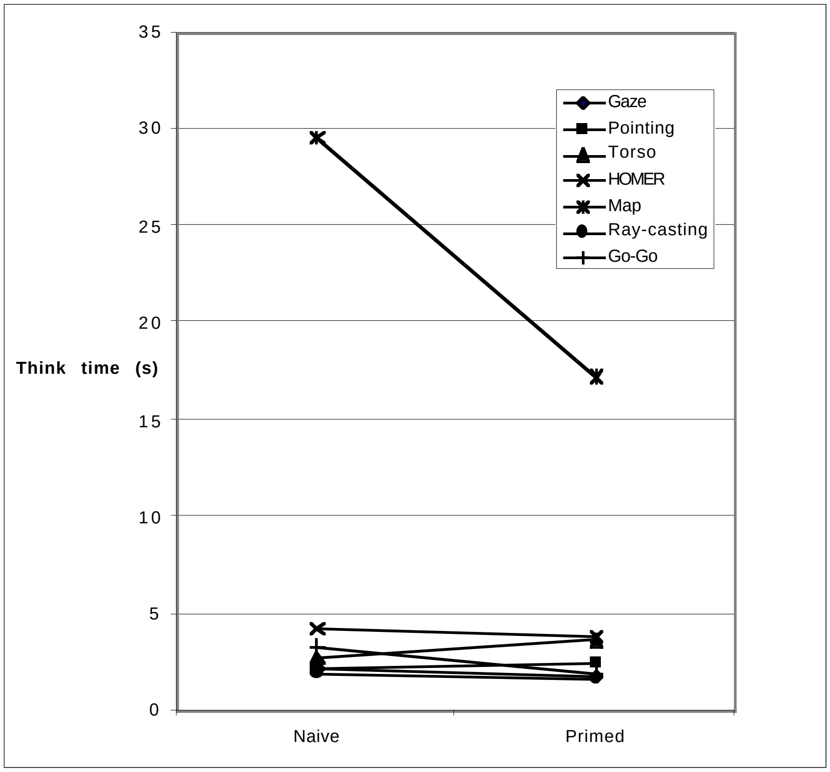


Figure 4.16 Interaction Between Task Type and Technique for Think Time on Search Tasks

Our evaluation showed that if the most important performance measure is speed of task completion, that steering techniques are the best choice. Users also seem to prefer these techniques over others. Of the steering techniques, pointing is clearly the most versatile and flexible, given our earlier results comparing it to gaze-directed steering and torso-directed steering. The Go-Go technique also performed well in this study with respect to speed. However, upon analysis of our comfort rating measures, we found that Go-Go produced arm-strain, dizziness, and nausea in some users. These problems were also seen with the HOMER technique, suggesting that viewpoint movement using hand-based manipulation may be discomforting to users because it is so different from the normal methods of movement. Gaze-directed steering also produced some significant discomfort (mainly dizziness), likely because it requires rapid and repeated head movements. The visual scene lags behind head movements due to tracker latency, so these could be the cause of discomfort. Of the seven techniques, only pointing and ray-casting produced no significantly high discomfort levels.

We also analyzed the demographic data and found no correlation between task performance and age, gender, VE experience, or spatial ability. These appear to be tasks whose speed is largely determined by the physical interaction technique used rather than individual differences.

As discussed above, the map technique was the most disappointing technique in this study. It seems to be well-suited for low precision, goal-directed travel. We believe that this technique would have performed better if the required accuracy had been lower on certain trials. It would probably also benefit from the use of a “view-up” map as opposed to a standard “north-up” map (Darken & Cevik, 1999). Performance on the primed-search would likely increase because of its egocentric nature. However, we have other reasons for using a north-up map, including the fact that it is a fixed frame of reference within a dynamic environment, and thus may facilitate learning of the spatial layout more quickly (Wickens & Baker, 1995). The map technique is also useful for other tasks, such as object manipulation (see chapter six), and so we do not believe that this technique should be removed from consideration as a result of its performance in this evaluation.

Finally, we also noted a reoccurrence of the theme of user strategies in this evaluation. No collision detection was implemented in the experimental environments, so users could move through objects if desired. In certain cases, this was highly advantageous, for example when the flag was just on the other side of a large fence. We noted that subjects using this strategy performed better on the primed search task, because they could take a straight-line path to the target. We also observed that certain techniques afford this strategy more than others. Steering techniques in general do not afford this, as they more closely mimic natural movement. Subjects using steering techniques generally went around obstacles. More unnatural techniques such as map dragging, Go-Go, HOMER, and ray-casting seem to suggest to the user that this environment does not work in the same manner as the physical world, and that therefore moving through objects is allowed. This represents another benefit of so-called “magic” techniques.

4.8 Summary

In this chapter, we have presented the results of our design and evaluation of viewpoint motion control, or travel, techniques for immersive VEs. Because of its pervasiveness, it is essential that we understand this task and the space of techniques for it. To this end, we have presented a formal framework for the design and evaluation of travel techniques, including two alternate taxonomies, performance metrics, and outside factors that could influence performance. Within this framework, we have designed new techniques and evaluated a wide range of techniques in six experiments. These evaluations, in particular the testbed experiment, have produced guidelines and empirical results that will allow application developers to choose appropriate travel techniques. We present such a practical application of these results in chapter six.

We learned several important lessons from our evaluation of travel techniques. From the relative motion experiment, we learned that techniques that do not couple the user’s head and the direction of motion are more efficient for relative motion tasks. The spatial awareness experiment showed that teleportation can cause disorientation in users. The information gathering experiment indicated that path complexity affects a user’s ability to obtain information from an environment. The spatial orientation experiment showed that users’ strategies, in conjunction with the affordances of the travel technique, can affect spatial orientation performance. Finally, the testbed evaluation indicated that steering techniques are generally efficient for search tasks.

CHAPTER V

SELECTION AND MANIPULATION

5.1 Introduction and Definitions

Once a VE user has been given the ability to move about the 3D space effectively, via a viewpoint motion control technique, the next step is to interact with the environment in some way. In a mechanical design application, this might mean positioning various parts so that they fit together. A training system for rescue workers might require the user to open doors, move obstacles, or make use of tools. A virtual science lab could allow the user to build molecules from components or position electrical charges. All of these interactions fall under the broad heading of selection and manipulation.

Selection involves the specification of one or more virtual objects by the user for some purpose. The purpose might be to specify the object of a command (e.g. delete the selected object), to invoke a command (e.g. selecting a menu item), to change the system state (e.g. selecting a toggle switch that controls a rendering parameter), or to choose a new tool (e.g. selecting a tool that creates cubes). Often, however, selection is performed to set up *manipulation*, that is, setting the position and/or orientation of a virtual object. Obviously, unless the user is constantly manipulating a single object, she must first select the object she wishes to manipulate.

Since many VE developers believe that the best way for the user to interact with a VE is the most natural way (a position we do not hold), many VE systems utilize a naive *natural mapping* for selection and manipulation. The natural mapping simply maps the scale and location of the user's physical hand directly to the scale and location of a virtual hand, so that when the virtual hand touches an object in the VE, it may be selected, and selected objects are manipulated by attaching them to the virtual hand – in other words, the user simply reaches out and grabs an object to select or manipulate it. This basic metaphor has been extended so that users can have fingertip control of virtual objects (Kijima and Hirose, 1996).

The natural mapping does have the advantage that it is quite intuitive for almost all users, since it replicates the physical world. However, this metaphor is simply not powerful enough for most VE applications. First, the objects that may be selected are only those within a physical arm's reach of the user, and once an object is selected, it may only be manipulated within that relatively small space. This may not be a problem if the work environment is only the size of a tabletop, but makes manipulation in larger environments difficult. To allow selection of faraway objects or large-scale movement of objects, a travel technique must be used in conjunction with the natural mapping.

Secondly, manipulation of large objects is problematic with the natural mapping. In the physical world, the objects that we can manipulate in our hands are limited to a certain size, but there are no such restrictions in the virtual world. Imagine a city planning application where the user wished to reposition a skyscraper. If the user was within an arm's length of the building, it would inevitably obscure the user's view, so that precise placement would be impossible.

When careful consideration is taken, it should be obvious that a real-world technique would be inadequate for selection and manipulation tasks in VEs, since the tasks we wish to perform go beyond our real-world capabilities. In the same way, a travel technique based on physical walking will be completely inadequate if the application requires travel on a global scale. The power of VEs is not to duplicate the physical world, but to extend the abilities of the user to allow him to perform tasks not possible in the physical world. For these reasons, we will consider in this chapter techniques for selection and manipulation that go beyond the natural mapping. In particular, the techniques will allow selection of objects at a distance, and manipulation within a large space.

5.2 Related Work

5.2.1 Interaction Metaphors

A variety of interaction techniques have been proposed and implemented which address the problem of selecting and/or manipulating objects within a virtual space. Among techniques which can select and manipulate faraway objects, most techniques fall into three categories: arm-extension, ray-casting, and image plane techniques.

Arm-extension techniques address the problem of the user's limited reach directly – they allow the user to extend her virtual hand much farther than her physical hand, so that faraway objects can be "touched." An advantage of such techniques is that manipulation can still be done via hand motion, as in the natural mapping. However, selection of objects that are very far away or small may be difficult, because the hand must be positioned precisely. Such techniques differ in the way that the virtual arm is extended. Some map the physical hand motion onto virtual hand motion using a mapping function (Poupyrev et al, 1996). Others use more indirect means to extend and retract the virtual arm (Bowman and Hodges, 1997). Still others employ more arcane mapping functions, such as from physical hand position to virtual hand velocity (Bowman and Hodges, 1997).

Ray-casting techniques select faraway objects by extending an idea from the 2D desktop metaphor. Just as one positions the pointer over an icon on the desktop to select it, so in three-dimensions one can point a virtual light ray into the scene to intersect and select a virtual object (Mine, 1995). Generally, the direction of the light ray is specified by the orientation of the user's hand (e.g. the ray emanates from the user's outstretched index finger), so that selection becomes a simple task of pointing at the desired object. The common manipulation scheme is to attach the object to the light ray at the point of intersection, but this makes manipulation unwieldy (Bowman, 1996), so other manipulation schemes may be desired.

Image plane techniques (Pierce et al, 1997) are a combination of 2D and 3D interaction. Selection of objects is done, as the name suggests, in the viewplane, so that the dimension of depth into the scene is not considered. For example, in one technique the user selects an object by partially occluding it with his virtual hand. That is, the virtual hand covers the desired object in the displayed image. Actually, this is a ray-casting technique, since one can consider it to use a ray emanating from the user's eyepoint and going through the virtual hand position to select an object, but we list these techniques separately, preferring that the term "ray-casting" be used for pointing techniques where the ray emanates from the virtual hand. Again, selection is simple for these techniques, but manipulation of objects once they are selected is an open question. Pierce et al's implementation (1997) scales the user so that the virtual hand actually touches the selected object, at which point natural hand movements can be used to manipulate the object. When the object is released, the user is scaled back to normal size.

Finally, there are certain techniques which do not fit into any of these categories. Rather, they try to maintain the intuitiveness of the natural mapping while overcoming its inherent limitations by employing the natural mapping in a manner not consistent with the physical world (perhaps we could call these "unnatural mappings"). One of the most obvious of these techniques is to employ a scaling factor (make the user larger or the world smaller) so that the user can reach any object with the virtual hand. Mine, Brooks, and Sequin (1997) use scaling together with a framework that allows the user to exploit his proprioceptive sense for navigation and manipulation. This can be a powerful metaphor, but may also have side effects for viewing the effects of changes – since the scale of the user and world are different, a small motion by the user results in a large motion in the world. Another idea employing scaling is to have two copies of the world, one large and one small. In the World in Miniature (WIM) technique (Stoakley, Conway, and Pausch, 1995), the user manipulates small objects in a "dollhouse" world held in the hand, and the corresponding full-size objects move accordingly. This has been extended in the recent "voodoo dolls" technique (Pierce, Stearns, & Pausch, 1999), in which the user creates his own miniature parts of the environment (dolls), and may use two hands to manipulate these doll objects relative to one another.

We also note that a good deal of work has been done in the area of aiding the user to position objects correctly. Most of these methods use some type of constraints to reduce the number of degrees of freedom that must be controlled by the user, or to reduce the required precision on the part of the user. For example, one can constrain an object to move only in one dimension (Bowman and Hodges, 1995), model an object's collisions with other parts of the world (Kitamura, Yee, and Kishino, 1996) or place some intelligence in the object so that it naturally seeks to be aligned correctly with the world and other objects (Bukowski and Sequin, 1995).

5.2.2 Evaluation of Techniques

There has been little work in the evaluation of selection and manipulation techniques for immersive VEs, but some studies have been reported in the areas of 3D selection and manipulation. Ware (Ware and Jessome, 1988, Ware and Balakrishnan, 1994) has carried out several investigations into the use of a tracked hand or input device for object placement in 3D environments. Also, Zhai and Milgram (1993) compared different input devices in a principled manner based on a proposed taxonomy of manipulation in 3D space.

One piece of work in immersive VEs deserves special mention. Poupyrev (1997) has implemented a "testbed" for the evaluation of selection and manipulation schemes, which incorporates our goals of systematic evaluation and multiple performance measurements. Unlike our proposed testbed, however, this work is more of a tool for those who would wish to perform experiments to compare various techniques. The user of the system can design and implement experiments quickly based on a text description of the interaction techniques, outside factors, and performance measurements. Our testbed, on the other hand, is a more generalized set of experiments that attempts to model all of the important variables and measurements.

5.3 Initial Evaluation and Design

Our first work in this area was inspired by a talk given at SIGGRAPH '96 on a new interaction technique for virtual object manipulation: the Go-Go technique (Poupyrev et al, 1996). The technique seemed intuitive and easy-to-use, and it promised to have wide application. However, no indications of performance were given, and no studies compared this technique with the many others that had been proposed for the same task. The technique had novelty and elegance, but we felt that this was not enough to proclaim it a cure-all. It needed to be tested and understood.

Therefore, we produced our own implementation of the Go-Go technique and several others and evaluated them with a simple user study (Bowman and Hodges, 1997). Our goal was to understand the characteristics of the task and the techniques, in an attempt to discover what makes a technique "good" for virtual object manipulation.

5.3.1 Techniques Considered

The techniques we studied fell into two categories: arm-extension and ray-casting. As we have noted, arm-extension techniques, including Go-Go, allow the user to select faraway objects by providing a mechanism by which the virtual arm may be made much longer than the physical arm. Users can then manipulate the objects directly with their hand, in a natural manner. Ray-casting techniques (Mine, 1995), on the other hand, use a pointing metaphor. A virtual light ray extends from the user's hand, and objects are selected by intersecting them with the light ray. The object is attached to the light ray for manipulation.

Within each of these categories, we investigated several variants. For arm-extension techniques, the main differentiator was the mapping technique used to determine the length of the virtual arm. The mapping function for the Go-Go technique, shown in figure 5.1, has two parts. When the user's physical hand is within a threshold distance D from the body, there is a one-to-one relationship between physical and virtual arm length. However, outside this threshold, the virtual arm length follows a non-linear function relative to the distance of the physical arm from the user's body.

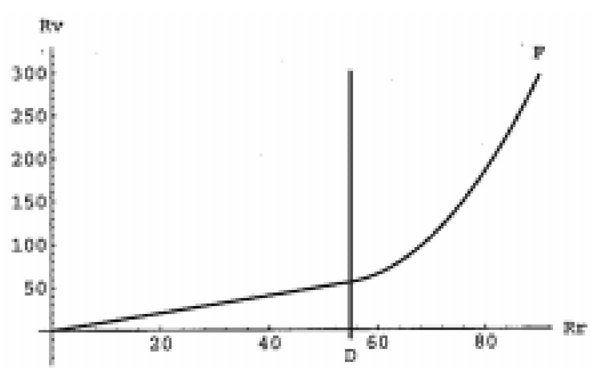


Figure 5.1 Mapping Function for the Go-Go Technique: R_r=Physical Hand Distance, R_v=Virtual Hand Distance. Reproduced from (Poupyrev et al, 1996)

We also looked at two other mapping functions. One is similar to Go-Go, except that there is no area of one-to-one growth – the virtual arm grows according to the non-linear function at every position ("fast Go-Go"). This allows the user's reach to extend to a greater, though still bounded, distance.

Second, we explored the possibility of mapping physical hand position to virtual hand *velocity*, in a technique we called "stretch Go-Go." This was done by defining three concentric regions of space about the user. When the physical hand is within the medium-range region, the virtual arm length is constant. If the physical arm is stretched far from the body, into the outer region, the virtual arm grows at a constant rate. Similarly, with the physical hand in the inner region, the virtual arm shrinks at a constant rate. This has the advantage that the user can reach any object, no matter its distance. To help the user visualize the mechanism, we provided a graphical gauge showing the three regions and the user's current hand position (figure 5.2).

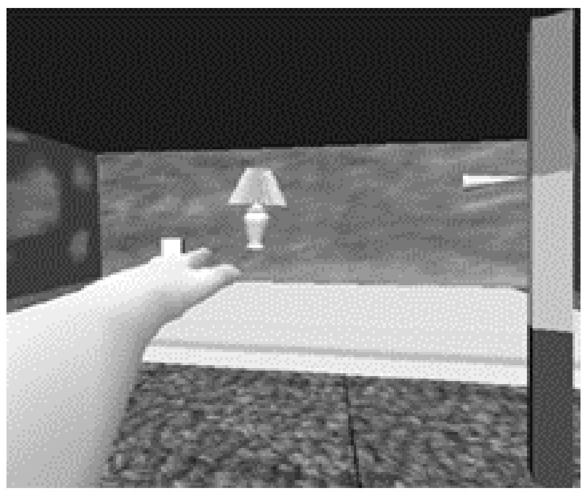


Figure 5.2 Stretch Go-Go Technique, with Gauge

Finally, we considered a technique that does not use a mapping function at all, but rather specifies the virtual arm length in a more indirect manner. This technique simply uses two mouse buttons to grow or shrink the virtual arm at a constant rate. Again, this technique has unlimited reach, although it may lack the intuitive characteristics of techniques where the arm is stretched out to make it longer.

We also included two ray-casting techniques in our survey. Both techniques use the same virtual light ray idea for object selection, and both manipulate the object by attaching it to the light ray. The techniques differ in their expressive power. With the basic ray-casting technique, there is no way to change the distance of the object from the user – the object must move along a sphere centered at the user whose radius is the object's original distance from the user. Thus, in the second of these techniques, we added a "reeling" feature, which allows the user to move the object closer or farther away along the light ray, similar to reeling a fishing line in or out.

5.3.2 User Study

Armed with these six techniques (four arm-extension and two ray-casting), we conducted a simple user study to assess their performance and applicability. Eleven student volunteers (two females and nine males) participated in the study. The equipment used included a Virtual Research VR4 head-mounted display, Polhemus Fastrak trackers, an SGI Indigo2 Max Impact, and a custom-built 3-button joystick. Users were immersed in a virtual room containing several pieces of furniture and given several minutes to practice and use each of the six techniques.

We did not collect any quantitative data in this study, but instead observed the performance and errors of the users, and collected their comments about the relative merits of each of the interaction techniques. This information led to a much more thorough understanding of the tasks of selection and manipulation, and of the techniques themselves.

None of the six techniques proved adequate for selection and manipulation of faraway objects. The favorite techniques were Go-Go and the indirect arm-extension technique, but problems were noted with each of these as well. There were difficulties with precision of selection, precision of manipulation, speed of use, user comfort, and expressiveness of the technique. We made three general observations about the tasks and techniques, which can be expressed as guidelines (hypothesis 2).

First, naturalism is not always a necessary component of an effective technique. Users almost unanimously found Go-Go to be the most natural technique, but many evaluators preferred other techniques. Indirect stretching was more effective for several subjects because it offered more precise control of the hand location, and less physical work on the part of the user. Several users also liked ray-casting with reeling because of the lack of physical effort required: they could support their arm and simply point with their wrists and press joystick buttons. This goes against common intuition regarding VE interaction: the most natural technique is not always the best in terms of performance or preference. This indicates that more formal methods are necessary to determine appropriate ITs (hypothesis 1).

Second, physical aspects of users were important in their evaluation of the techniques. For example, those users with shorter arms were less likely to prefer the go-go technique because their reach was more limited. Also, all of the arm-extension techniques depend on the specification of a point at the center of the user's torso. The virtual hand in these techniques is kept on a line defined by this torso point and the location of the physical hand. Although we defined this point relative to the user's head position, the height of the user made a difference. If the torso point is not approximated well, the hand will appear lower or higher than it should be, and grabbing and manipulation will be more difficult. In short, techniques that are dependent on the user will require user modeling in order to be most effective.

Our most important finding, however, was that grabbing and manipulation must be considered separately for overall usability. Although only two of our users preferred a ray-casting technique overall, almost every user commented that it was easier to grab an object using ray-casting than with any of the arm-extension techniques. This result agreed with our earlier observations on the use of ray-casting in VE applications (Bowman, 1996, Bowman, Hodges, and Bolter, 1998). It requires no arm stretching and less precision on the part of the user: one simply points the ray and releases the button. With the arm-extension techniques, one must place the hand within the object, which can be quite difficult at a great distance or when a small physical motion maps to a large translation of the virtual hand.

On the other hand, no users preferred ray-casting techniques for object manipulation, as arbitrary rotations of an object are practically impossible using these techniques. With an arm-extension technique, objects can be rotated in their own coordinate system, and their position can be controlled easily as well. None of the current techniques, then, were universally acclaimed, because none of them were easy to use and efficient throughout the entire interaction: grabbing, manipulating, and releasing the object.

5.3.3 HOMER Technique

In response to these results, it was clear that a hybrid technique combining the best features of both the arm-extension and ray-casting metaphors could provide gains in efficiency, accuracy, and usability. This simple observation led to the implementation of the HOMER (Hand-centered Object Manipulation Extending Ray-casting) family of techniques. These techniques simply use the better-performing metaphor for each part of the task: ray-casting for object selection and in-hand object manipulation.

The basic technique works like this (see figure 5.3): the user activates the virtual light ray and intersects the desired object with it by pointing, just as in the ray-casting technique. Upon releasing the button, the virtual hand immediately moves to the center of the selected object, so that manipulation can be performed directly with the hand, and so that any rotation can be achieved. When the drop command is given, the virtual hand returns to the location of the physical hand.

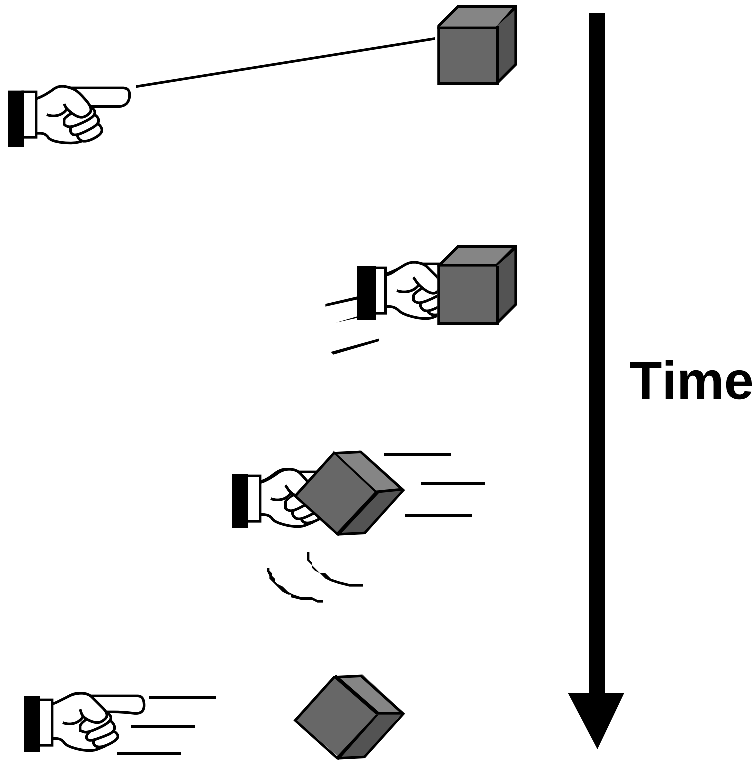


Figure 5.3 Time Sequence of the HOMER Technique

The HOMER techniques exhibit both ease of selection and ease of manipulation, since they use well-performing technique components for both of these tasks. There is one issue that must be addressed, however, to make the HOMER techniques completely expressive (that is, to ensure that they allow a user to place an object at any position and orientation). This is the question, again, of object distance from the user. In the basic HOMER technique, hand motions are mapped one-to-one onto the object, so there is no way the object could be placed twice as far away from the user, or brought very near for inspection. Thus, we need a mechanism for controlling object depth once the object has been selected.

We provide two such mechanisms, one direct and one indirect. The indirect HOMER technique simply uses the “reeling” feature discussed earlier, where two mouse buttons are used to move the object nearer or farther away. This provides complete expressiveness, but may be slow or cumbersome. The direct HOMER technique uses a linear mapping function to control object depth. A linear function was chosen because it is more predictable and easier to control than a non-linear function, no matter the distance from the user. The virtual object moves N meters in or out for every one meter of physical hand motion in or out, where N is the ratio between the original object-to-user distance and the original hand-to-user distance. Therefore, if the user moves his physical hand twice as far away from his body, the object will move to twice its original distance from the body as well.

This technique also allows the user to have direct control of the mapping function, since it depends on the distance between the user's physical hand and her body at the time of selection. If a large N is needed, the user can place her hand very close to her body, but if more control is desired, the hand can be positioned farther away.

5.4 Formal Evaluation Framework

5.4.1 Categorization of Techniques

The initial user study provided us with a good understanding of the tasks of selection and manipulation, and of the space of possible techniques for realizing these tasks. Our original categorization of techniques into arm-extension, ray-casting, image-plane, and "other" techniques is useful at a high level, but there may be large performance differences within a category. Therefore, this categorization does not allow us to make generalizations such as, "arm-extension techniques provide greater accuracy of placement," since this depends on the implementation of the arm-extension technique.

Therefore, we have re-categorized ITs for selection and manipulation based on a more formal task analysis, as we did for travel techniques. This taxonomy is shown in figure 5.4.

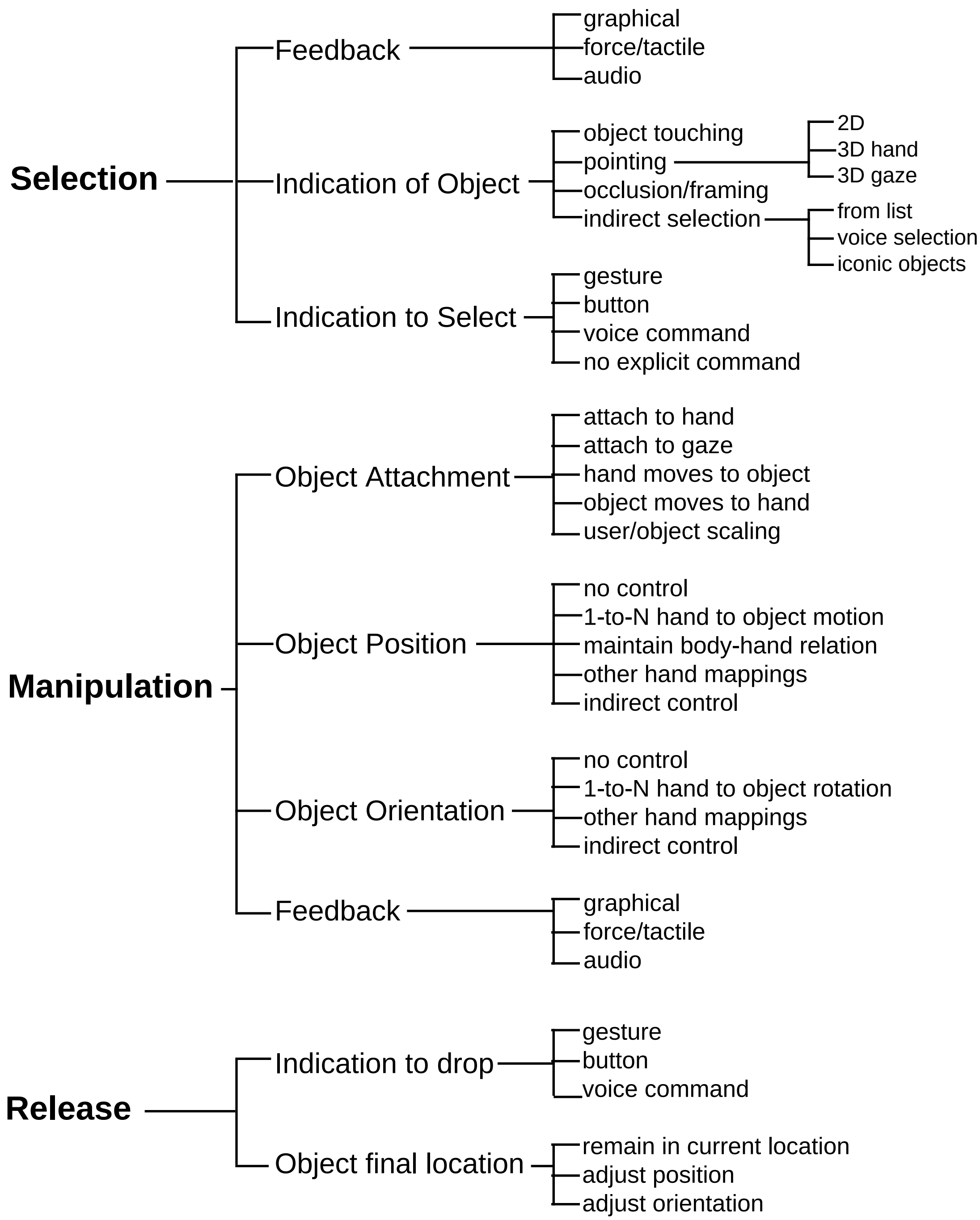


Figure 5.4 Taxonomy of Selection/Manipulation Techniques

The first thing that should be noted about the taxonomy are its three main branches, which break the task into its component parts: selection, manipulation, and release of the object. For selection-only tasks, the top branch of the taxonomy may be used alone. This division stems from the observation we made in our user study that selection and manipulation should be considered separately for optimal performance.

The main subtasks within the selection branch are the indication of the object and the indication to select the object. These subtasks are listed separately since the indication of the object does not necessarily imply that the object should be selected. For example, in a simple technique where the user touches objects, the user may touch many objects with his virtual hand, but only selects an object when a button is pressed while the object is being touched. Feedback is also given as a subcomponent of selection, but this is purely an interaction issue, and does not correspond to an actual user goal.

The second branch lists components and techniques for manipulation. Subtasks that are purely task-related are the indication to start manipulating the object (often the same as the indication to select, but not necessarily), indication of the center of rotation (not required), and the technique(s) to control object position and orientation. Object attachment is a technique consideration that may or may not apply – it refers to the way the object is attached to the manipulator (often the virtual hand). Feedback is also listed as an interaction component.

The final main branch concerns the release of a manipulated object. The only task-related component here is that the user must give some indication to drop the object (stop manipulation). From a technique point of view, however, the most important components of a release technique are what happens to the object and/or the virtual hand after release. For example, virtual gravity might be implemented which causes the object to fall naturally to a surface below. Also, in a technique where the virtual hand is displaced from the location of the physical hand (e.g. HOMER), the virtual hand position may need to be adjusted so that it once again coincides with the physical hand's position.

This taxonomy does not have the intuitive appeal of the broad technique categories mentioned above, but it is much more complete and general. It allows us to make interesting comparisons between various components of techniques, and general statements about performance. Perhaps even more important is the fact that this taxonomy encourages the guided design of new techniques because of its task-oriented structure.

5.4.2 Performance Measures

Like viewpoint motion control, selection and manipulation techniques can be evaluated for performance with a large number of possible metrics. Some techniques may trade off performance on one measure for better performance on another, and different applications may perform best with very different interaction techniques, due to different performance requirements. Again, we need to consider both quantitative and qualitative metrics, and those relating to the task as well as those relating to the user.

As in the case of ITs for travel, we have defined a list of metrics with which performance of techniques can be measured. Application designers can specify requirements for selection and manipulation in terms of those metrics, and choose ITs which meet those requirements.

Our list of performance metrics for immersive selection and manipulation techniques includes:

1. *Speed* (efficiency of task completion)
2. *Accuracy of Selection* (the ability to select the desired object)

3. *Accuracy of Placement* (the ability to achieve the desired position and orientation)
4. *Ease of Learning* (the ability of a novice user to use the technique)
5. *Ease of Use* (the complexity of cognitive load of the technique from the user's point of view)
6. *Presence* (the user's sense of immersion within the environment while using the technique)
7. *Expressiveness of Selection* (the number and distance of objects that can be selected)
8. *Expressiveness of Manipulation* (the ability to position and orient the object at any desired location in the environment)
9. *User Comfort* (lack of physical discomfort, including simulator sickness)

Speed and accuracy are important to many of the target applications, but more user-centric metrics such as user comfort can also play a major role. Many of the techniques which allow complete 6 DOF manipulation of virtual objects can force the user to assume awkward arm, wrist, or hand positions, for example. Also note that accuracy and expressiveness play a double role here, having different meanings for selection vs. manipulation.

5.4.3 Outside Factors

The final component of our formalized evaluation framework for selection and manipulation techniques is the consideration of other factors that could affect the performance of a technique. These factors were explicitly modeled in the evaluation testbed, so that performance differences could be attributed to the proper source. As before, we separate these outside factors into four categories: task, environment, user, and system characteristics.

5.4.3.1 Task Characteristics

A technique may perform very well for certain selection/manipulation tasks, but poorly on others. To determine these relationships, we can consider the following set of task characteristics:

- distance from the user to the object
- degrees of freedom required to be manipulated
- accuracy required
- task complexity (cognitive load induced)

5.4.3.2 Environment Characteristics

The environment (3D virtual world) surrounding the user can also have an effect on selection and manipulation. Interesting variables include:

- visibility
- number of objects
- size of objects
- shape of objects
- density of objects
- activity (motion)
- size of environment

- level of detail
- randomness/structure in the environment

5.4.3.3 User Characteristics

The individual user is also quite important for selection/manipulation techniques. For example, the Go-Go technique is less powerful for users with shorter arms. We have identified these user characteristics for consideration:

- age
- gender
- length of reach
- spatial ability
- height
- VE experience
- visual acuity
- manual dexterity
- ability to fuse stereo images
- technical/non-technical background

5.4.3.4 System Characteristics

Finally, the hardware and software comprising the VE system may themselves have effects on performance of selection/manipulation tasks. Such characteristics include:

- rendering technique
- use of shadows
- virtual body representation
- frame rate
- latency
- display type
- use of collision detection or constraints
- realism of physics model (e.g. gravity)

5.4.4 Guided Design

The selection and manipulation taxonomy has also proven useful as a framework for the design of new techniques. Because there are such a large number of techniques described in the literature, most of the techniques that arise from guided design are variants of techniques already available. However, small changes to certain subtasks can have a large effect on performance.

We have taken the guided design of selection and manipulation techniques to the next logical step by "implementing" the taxonomy in software. Five low-level subtasks (selection, attachment, positioning, orientation, and release), along with a large number of technique components for each of these subtasks, have been implemented in a modular fashion so that they can be arbitrarily combined automatically. In other words, a designer can create a new IT immediately simply by entering five codes into a program. Currently, there are 8 x 6 x 6 x 4 x 4 = 4608 possible combinations of technique components. However, because of dependencies and constraints in the design space, the number of possible techniques is reduced to 667.

Through experimentation with this system, a number of interesting possibilities have emerged. For example, a HOMER-like technique which uses gaze direction instead of pointing direction for selection frees the hands for other tasks until an object is selected. It also seems useful in some cases to separate positioning and orientation of objects by using two trackers instead of one that controls all six degrees of freedom. We can also combine techniques such as HOMER and Pierce's (1997) "sticky finger" technique, to use the best aspects of each. For example, occlusion selection might prove easier than 3D ray-casting, and so it could be used in a technique along with HOMER-style object manipulation.

5.5 Selection/Manipulation Testbed

The three components of the formal framework (taxonomy, performance measures, and outside factors) come together in the evaluation testbed for selection and manipulation. This testbed is a set of tasks and environments that measure the performance of various combinations of technique components for each of the performance metrics. Ideally, this testbed would vary all of the outside factors listed above, but such an experiment would not be completed for decades.

Therefore, we designed and implemented a simpler testbed system that can evaluate techniques in a number of what we consider to be the most important conditions. The analysis of importance is based on our experiences with real applications, our more informal study of selection and manipulation, and the requirements of our target application.

The testbed was designed to support the testing of any technique that can be created from the taxonomy. The tasks and environments are not biased towards any particular set of techniques. We have evaluated nine techniques, but others can be tested at any time with no loss of generality.

The tasks used are simple and general. In the selection phase, the user selects the correct object from a group of objects. In the manipulation phase, the user places the selected object within a target at a given position and orientation. Figure 5.5 shows an example trial. The user is to select the blue box in the center of the three by three array of cubes, and then place it within the two wooden targets in the manipulation phase. In certain trials, yellow spheres on both the selected object and the target determine the required orientation of the object.

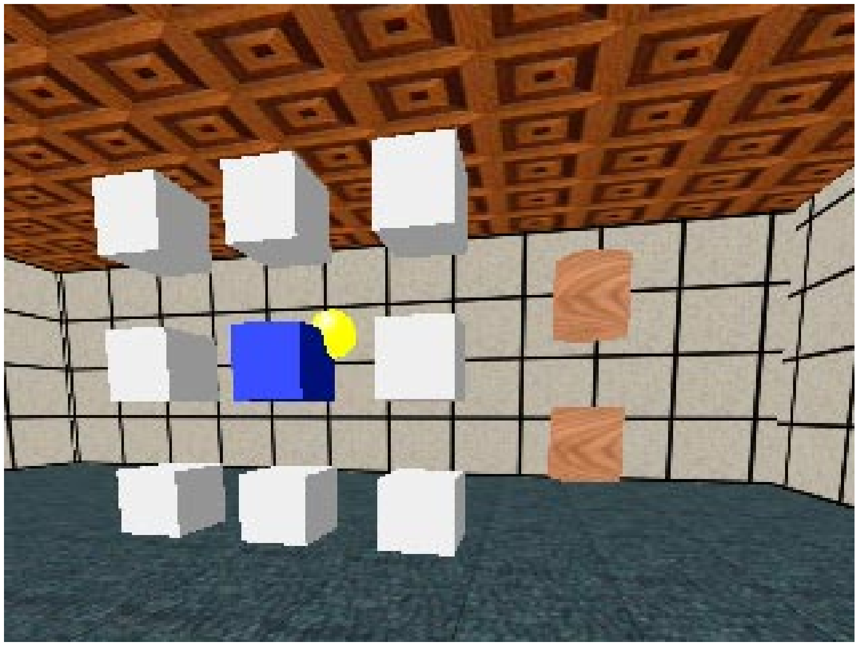


Figure 5.5 Example Trial Setup in the Selection/Manipulation Testbed

5.5.1 Method

Three within-subjects variables were used for the selection tasks. We varied the distance from the user to the object to be selected (three levels), the size of the object to be selected (two levels), and the density of objects surrounding the object to be selected (two levels).

The manipulation phase of the task also involved three within-subjects variables. First, we varied the ratio of the object size to the size of the target (two levels – this corresponds to the accuracy required for placement). Second, the number of required degrees of freedom varied (two levels), so that we could test the expressiveness of the techniques. The 2 DOF task only required users to position the objects in the horizontal plane (with constraints implemented that prevented the user from rotating the object or moving it vertically), while the 6 DOF task required complete object positioning and orientation. Finally, we changed the distance from the user at which the object must be placed (three levels), since this was a primary concern in our earlier user study.

Besides these explicit variables, we also included characteristics of the user in our analysis. We studied the effects of age, gender, spatial ability, VE experience, and technical background on the performance of techniques by having users fill out a pre-experiment questionnaire (Appendix A) and standardized spatial ability test (the ETS cube comparison test).

Response variables were the speed of selection, the number of errors made in selection, the speed of placement, and qualitative data related to user comfort (the same subjective reports as in the travel testbed – arm strain, hand strain, dizziness, and nausea on a ten-point scale; see Appendix B). We did not measure accuracy of placement; instead we required users to place the selected objects completely within the targets and within five degrees of the correct orientation on the six degree of freedom trials. Graphical feedback told the user when the object was in the correct location.

Forty-eight subjects (31 males, 17 females) participated in the study. Subjects were undergraduates from the Department of Psychology subject pool, and were given extra credit for their participation. Each subject completed 48 trials, except for three subjects who did not complete the experiment due to dizziness or sickness.

Nine different selection/manipulation techniques, taken from the taxonomy, were compared in a between-subjects fashion. Thus, there were five subjects per technique. First, we chose the Go-Go technique because of its importance and the fact that it was under consideration as the technique to be used in the Virtual Habitat application (chapter six). The other eight techniques were created by combining two selection techniques (ray-casting and occlusion), two attachment techniques (moving the hand to the object, scaling the user so the hand touches the object), and two positioning techniques (linear mapping of hand motion to object motion and the use of buttons to move the object closer or farther away).

Subjects wore a Virtual Research VR4 HMD, and were tracked using Polhemus Fastrak trackers. Input was given using a 3-button joystick. Subjects were allowed to practice the technique for up to five minutes in a room filled with furniture objects before the experimental trials began. Subjects completed four blocks of 12 trials each, alternating between trials testing selection and manipulation. After the practice session and after each block, subjective comfort information was taken.

5.5.2 Results

This complex experiment necessarily has a complex set of results. Here, we will present several major findings that emerge from the data. For complete results, see Appendix D. We performed a repeated measures analysis of variance (MANOVA) for both the selection and manipulation tasks.

First, results for selection of objects matched most of the experience that we had in our earlier informal study. Selection technique proved to be significant ($f(2,42)=13.6$, $p < 0.001$), with the Go-Go technique (mean 6.57 seconds per trial) proving to be significantly slower than either ray-casting (3.278 secs.) or occlusion selection (3.821 secs.) in post-hoc comparisons (LSD and Bonferroni). There was no significant difference between ray-casting and occlusion. This is because selection using ray-casting or occlusion is essentially a 2D operation, while the Go-Go technique requires users to place the virtual hand within the object in three-dimensional space.

We also found significant main effects for distance ($p < 0.001$) and size ($p < 0.001$), with nearer and larger objects taking less time to select. There were also several interesting significant interactions. As shown in figures 5.6 and 5.7, the effects of distance and size varied depending on the selection technique being used ($p < 0.001$ in both cases). Figure 5.6 shows that selection time for the Go-Go technique increases with distance, while the other two selection technique times remain approximately constant, regardless of object distance. Figure 5.7 indicates that the Go-Go technique benefits much more from larger object sizes as compared to ray-casting and occlusion selection.

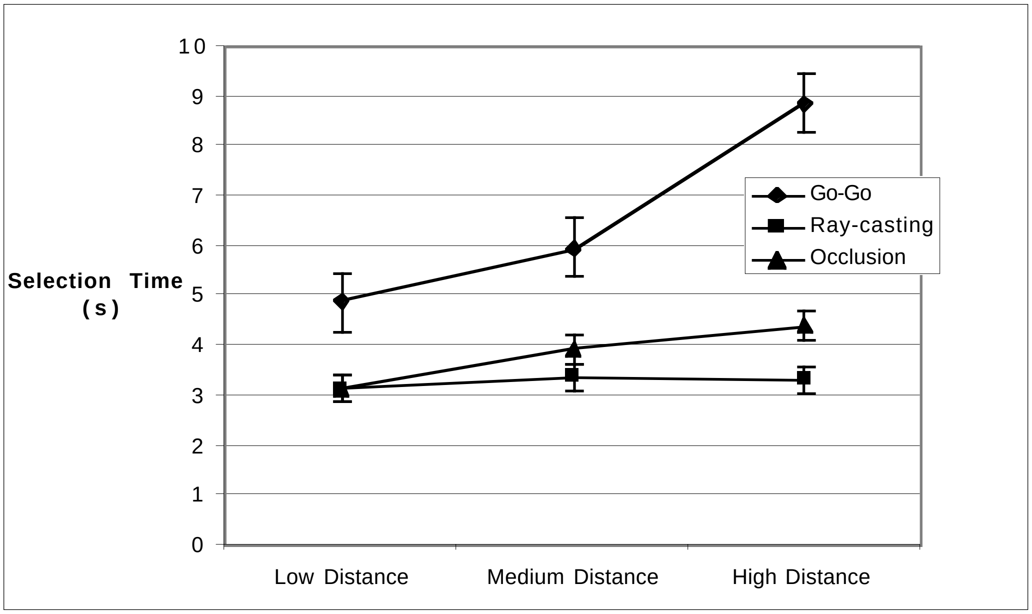


Figure 5.6 Interaction of Selection Technique with Object Distance for Selection Time Measure

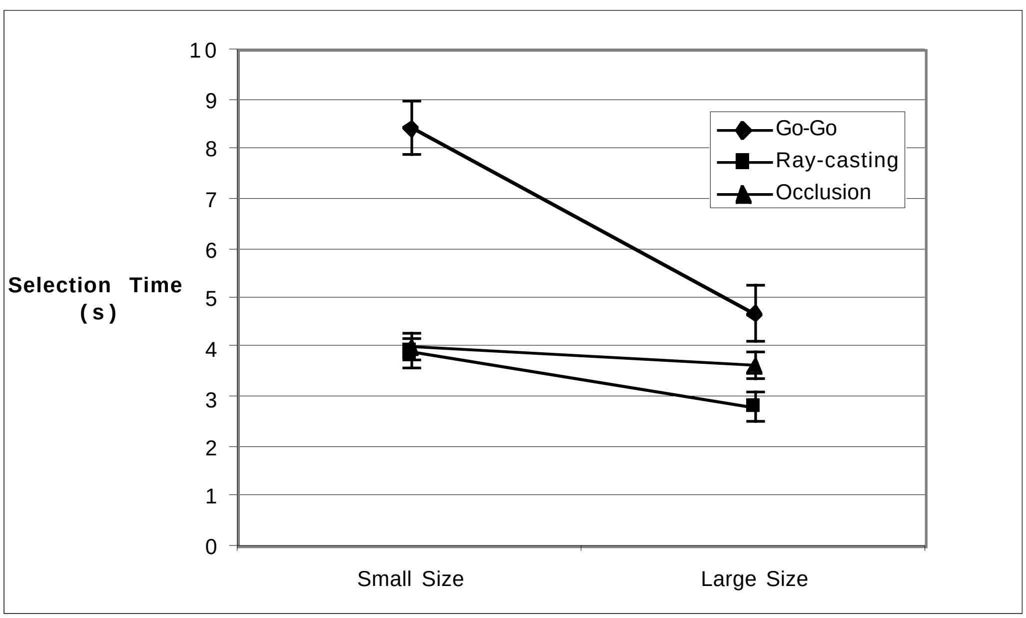


Figure 5.7 Interaction of Selection Technique with Object Size for Selection Time Measure

We found that the number of errors made during selection (errors included both selecting the wrong object and selecting no object) were significantly affected by both distance ($p < 0.001$) and size ($p < 0.001$). Interestingly, however, selection technique had no significant effect on errors.

It appears from this data that either ray-casting or occlusion is a good general-purpose choice for a selection technique. However, this is tempered by our findings with regard to user comfort. We found that selection technique had a high correlation to the reported final level of user arm strain (after all trials had been completed, approximately thirty minutes of use). Occlusion selection produced significantly higher levels of arm strain than ray-casting, because ray-casting allows the user to "shoot from the hip," while occlusion selection requires that the user's hand be held up in view. When selection takes a long time, as in the case of small or faraway objects, this can lead to arm strain of unacceptable levels.

The results for manipulation time were more difficult to interpret. Once the object had been selected, many of the techniques produced similar times for manipulation (table 5.1 shows the results for the nine techniques). We did find a significant main effect for technique ($f(8,36)=4.3$, $p < 0.001$) where technique is the combination of selection, attachment, and manipulation components. The only combinations that were significantly worse than others in the post-hoc tests were the two combinations that combined ray-casting with the attachment technique that scales the user, and this was likely due to poor implementation, from our observations of users. We found no significant effects of technique when attachment and manipulation techniques were considered separately.

Table 5.1 Mean Manipulation Time Results by Technique from Testbed Evaluation

(The linear mapping used in these cases was a one-to-one physical to virtual hand mapping)*

Tech	Selection	Attachment	Manipulation	Mean Time (s)
1	Go-Go	Go-Go	Go-Go	26.551
2	Ray-casting	Move hand	Linear mapping	32.047
3	Ray-casting	Move hand	Buttons	30.970
4	Ray-casting	Scale user	Linear mapping*	40.683
5	Ray-casting	Scale user	Buttons	39.851
6	Occlusion	Move hand	Linear mapping	31.800
7	Occlusion	Move hand	Buttons	22.537
8	Occlusion	Scale user	Linear mapping*	24.780
9	Occlusion	Scale user	Buttons	20.528

One interesting fact to note from table 5.1 is that for each pair of techniques using the same selection and attachment components, the technique using indirect depth control (button presses to reel the object in and out) had a faster mean time. Though this was not statistically significant, it indicates that an indirect, unnatural positioning technique can actually produce better performance. These techniques are not as elegant and seem to be less popular with users, but if speed of manipulation is important, they can be a good choice.

All three of our within-subjects variables proved significant. Distance ($f(2,72)=18.6$, $p < 0.001$), required accuracy ($f(1,36)=19.6$, $p < 0.001$), and degrees of freedom ($f(1,36)=286.3$, $p < 0.001$) all had significant main effects on manipulation time. As can be seen from the large f-value for degrees of freedom, this variable dominated the results, with the six degree of freedom task taking an average of 47.2 seconds to complete and the two degree of freedom task taking 12.7 seconds on average.

We also found a significant interaction between required accuracy and degrees of freedom, shown in table 5.2. The six degree of freedom tasks with a high accuracy requirement (small target size relative to the size of the object being manipulated) were nearly impossible to complete in some cases, indicating that we did indeed test the extremes of the capabilities of these interaction techniques. On the other hand, required accuracy made little difference in the 2 DOF task, indicating that the techniques we tested could produce quite precise behavior for this constrained task.

Table 5.2 Interaction Between Required Accuracy and Degrees of Freedom for Manipulation Time (seconds)

	2 DOFs	6 DOFs
Low Accuracy	11.463	40.441
High Accuracy	13.991	53.992

Unfortunately, these data cannot answer the question of whether there is a qualitative difference between the 2 DOF and 6 DOF tasks. Does the 2 DOF task have a constant slope regardless of the required accuracy or is its upward slope simply of lower magnitude than that of the 6 DOF task? In other words, does adding more degrees of freedom to a manipulation task create a different type of task, or does it simply add more of the same types of difficulty? The best way to answer these questions would be to include a middle condition with three degrees of freedom, and we propose this as future work. We can get some idea of the importance of this interaction by looking at these data on a log scale (figure 5.8). This graph does not appear to show an interaction, and thus we suggest that degrees of freedom may be additive, and not qualitatively different. This may be a fruitful topic for further research.

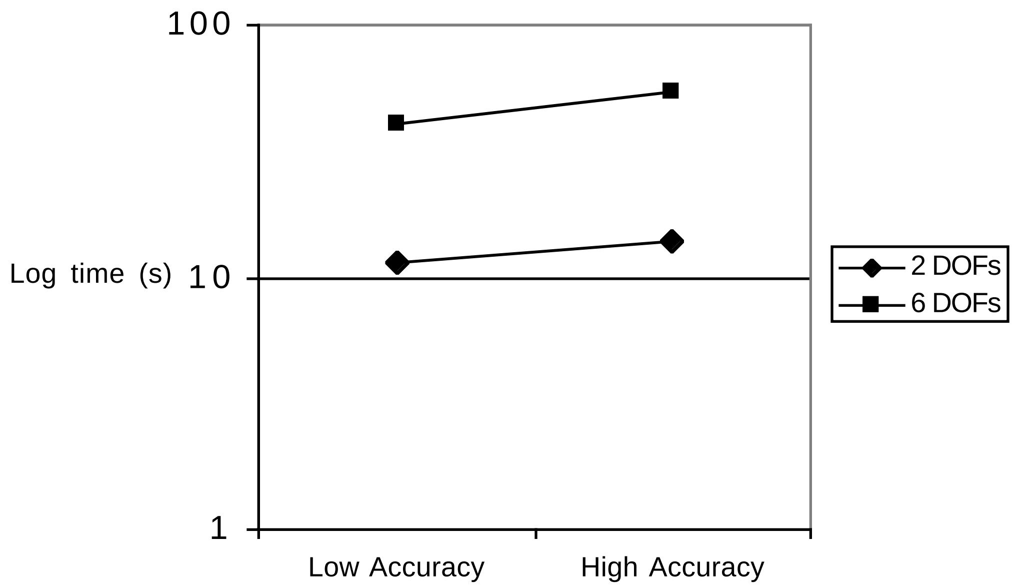


Figure 5.8 Logarithmic Scale Graph of Interaction Between DOFs and Accuracy

All of the significant results reported above have observed statistical power (computed using alpha = 0.05) of 0.92 or greater.

Finally, we found a demographic effect for performance. Males performed better on both the selection time ($p < 0.025$) and manipulation time ($p < 0.05$) response measures. Spatial ability and VE experience did not predict performance.

Again, looking at the results, we have any of a number of manipulation techniques to choose from which appear to have similar performance. The lowest mean times were achieved by techniques using occlusion selection and/or the scaling attachment technique (techniques 7, 8, and 9). The fact that the scaling technique produces better performance, especially on the six degree of freedom task, makes intuitive sense. If the user is scaled to several times normal size, then a small physical step can lead to a large virtual movement. That is, users can translate their viewpoint large distances while manipulating an object using this technique. Therefore, on the difficult manipulation tasks, users can move their viewpoint to a more advantageous position (closer to the target, with the target directly in front of them) to complete the task more quickly. We observed this in a significant number of users.

However, these techniques also have a price. We have already stated that occlusion selection increases arm strain. Similarly, scaled manipulation significantly increases the reported final level of dizziness relative to techniques where the user remains at the normal scale. Thus, an important guideline (hypothesis 2) is that such techniques should not be used when users will be immersed for extended periods of time.

5.6 Summary

In this chapter, we have used our design and evaluation methodology to study techniques for the selection and manipulation of objects in immersive VEs. These tasks will be found in most interactively complex VE applications, so it is crucial that we understand the performance characteristics of the various proposed ITs. Our initial user study of arm-extension and ray-casting techniques gave us useful information and understanding of these two metaphors, and allowed us to combine them for better performance in the HOMER techniques. We used this knowledge as a basis for our formal design and evaluation framework, including a taxonomy of selection and manipulation techniques, performance metrics, and outside factors that could influence performance. This framework was realized in our testbed evaluation, which produced complex but useful empirical results. In chapter six we apply these results to a complex VE application in order to increase its performance.

Several important principles come out of this research. Our user study showed that naturalism does not necessarily produce good performance on selection and manipulation tasks. Rather, magic techniques seemed to be easier, more efficient, and more acceptable to users. The testbed experiment showed that 2D selection metaphors based on ray-casting were more efficient, that the perceived size of virtual objects affects selection errors, and that scaled object manipulation can increase efficiency on difficult manipulation tasks. We also found user comfort to be a significant measure for selection and manipulation tasks. If speed were the only consideration, a technique such as Sticky Finger (occlusion selection combined with scaled object manipulation) would be an excellent choice. However, both of these components produced moderate to high levels of discomfort in users, which will not be acceptable in applications with longer exposure times.

CHAPTER VI

INTERACTION IN A REAL-WORLD VE APPLICATION

6.1 Integrating Techniques into an Application: Issues and Challenges

The formal categorization, design, and evaluation that has been discussed in the previous chapters cannot be an end unto itself. Rather, it must be done with a view towards applying those results to some practical, useful, real-world systems. The reason interaction techniques are so important is that they allow the user to act – carry out some task that is a part of the user's productivity, education, or enjoyment. Therefore, in this chapter we will consider a practical VE application with extensive interactivity requirements, and how that application can benefit from the formal evaluation of interaction techniques (hypothesis 3).

However, applying the results of our experiments to an application is not as straightforward as it might seem. Recall that in our methodology, application developers specify levels of performance that are required by the application (for many different performance metrics), and then implement techniques that have been shown to meet those requirements through testbed evaluation. There are a number of issues that we must deal with to accomplish this goal.

First, the specification of requirements is not a trivial matter. For quantitative metrics such as efficiency, the developer may have only a rough idea of the requirement. Qualitative measures will be even more difficult to specify. Also, since many VE applications are currently the first of their kind, one may not know the interaction requirements until testing has been done (and to do this testing, you need a working application). We can approach this issue by allowing developers to specify ranges of performance, and by standardizing more qualitative measures. The problem of unknown requirements is not likely to go away, and so iterative design will be imperative. We cannot hope to obtain the perfect set of interaction techniques on the first try for every system.

A second issue is that ITs cannot be considered in a vacuum. If we blindly choose those techniques that best fit our requirements, without regard for how well the techniques work together, we may create a more difficult-to-use application. The issue of *technique integration* is key. Developers of user interfaces have long held the principle that an appropriate overall interaction metaphor makes a system more usable. In the same way in a VE application, a set of three complementary interaction techniques may prove more usable than three unrelated techniques that meet every application requirement.

Finally, we must consider the specific tasks that are part of the application. VE systems for surgical training and interior design may both require accurate object manipulation techniques, but the same technique may not suffice for both applications. The surgery simulator likely needs a high level of realism, while the design application would only be concerned with final placement. Thus, the domain in which the tasks are performed is also important, and should be taken into account when ITs are chosen.

6.2 The Virtual Habitat

The immersive system to which we will apply our results has the goals of user learning and design, and has interesting requirements for interaction techniques for travel, selection, and manipulation. The domain of this system is environmental design, more specifically the design of animal habitats for zoos.

6.2.1 Original VR Gorilla Application

Figure 6.1 shows a wide view of the virtual gorilla exhibit, which is an accurate 3D model of the main gorilla habitat at Zoo Atlanta. The model includes terrain, rocks, trees, fallen logs, moats, an interpretive center, and four virtual gorillas. This model was originally used in an educational application aimed at middle school students (Allison et al, 1997). The students, by taking on the persona of an adolescent gorilla, could learn about gorilla behaviors, vocalizations, and social structure. The user could both explore the habitat freely and interact with the autonomous virtual gorillas.

Figure 6.1 The Virtual Reality Gorilla Exhibit

This original VR Gorilla system is quite interesting for the study of education in VEs and research into believable real-time virtual creatures, but is not as interesting from an interaction point of view, since the user only has to move through the space in some way. To accomplish this, the system uses a simple gaze-directed steering technique, with the user's vertical position constrained to a given height above the ground. However, the developers are considering a torso-directed technique, which is more like walking, and allows the user to look around while moving.

6.2.2 Application to Environmental Design Education

The system on which we will focus our attention is an extension of the original VR gorilla exhibit called the Virtual Habitat. This application is also educational, but is aimed at university-level students, and has the goal of teaching the principles of environmental design. The user is immersed within the same habitat model, with the only difference being that the virtual gorillas now remain stationary and do not react to the user.

The design of zoo exhibits is a topic on the boundaries between architecture, zoology, and psychology, and requires careful attention to a variety of sometimes conflicting requirements. The needs of the animals must be met, and so a naturalistic habitat is often a goal. The animals require some privacy, but visitors must also be allowed to see the animals. Plants need to match the region from which the animal has come, but must also be hardy enough so that they are not destroyed by the animals. In short, there are a number of interesting issues that can be explored by environmental designers (Coe, 1985).

Many of the details of this subject are difficult to learn without examples, and so we felt that the pre-existing virtual gorilla habitat would be an ideal way to provide these examples interactively. Therefore, in one component of the application, users can access embedded information about zoo exhibit design, which are positioned so that the abstract information and the real-world example are colocated. Thus, students have a more visual and interactive method of retrieving information. The embedded information can be in audio, text, or image formats (see figure 6.2). A small study (Bowman, Wineman, Hodges, & Allison, 1999) has shown that this approach, when paired with classroom teaching, may produce better learning and retention of information than a traditional lecture alone.

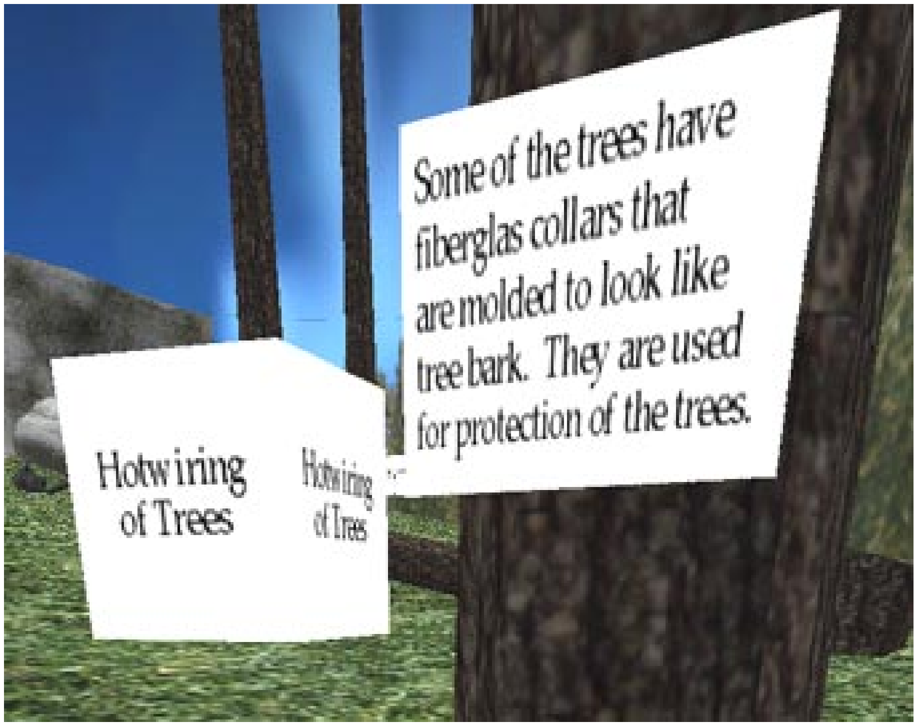


Figure 6.2 Embedded Audio and Text Information in the Virtual Habitat

We also want the user to be able to apply this newfound knowledge in a real-world setting. Therefore, the second component of the virtual habitat application allows *immersive design* – the modification of the existing habitat design while immersed within the habitat (Butterworth et al, 1992, Mine, 1997). Immersive design can tighten the design cycle and allow users to view the effects of changes immediately, but it is also quite different from the way architects are accustomed to designing. They must be able to reason and create in three dimensions, from within the design itself, rather than the normal 2D, outside-in view.

The immersive design component of the Virtual Habitat has three domain-specific tasks. First, users can modify the shape of the terrain, which is important for line-of-sight, privacy, and viewer subordination (Coe, 1985) issues. Second, the visual elements (trees, rocks, tufts of grass) in the habitat can be moved or deleted, or new ones can be created. These elements serve an important aesthetic purpose and influence the naturalism of the exhibit. Finally, the system allows modifications to the design of the visitor viewpoints into the habitat, including their position, viewing direction, and field of view. Issues here include viewing opportunities, privacy, and naturalism.

6.2.3 Interaction Requirements

As we have said, our methodology maps interaction techniques to applications through the use of requirements specification. That is, the application designer specifies levels of the various performance metrics that are required or desirable in the system, and then techniques which have been shown to exhibit those traits can be chosen. The Virtual Habitat application has a number of interesting requirements for interaction, and it includes all of the universal tasks: travel, selection, and manipulation.

There are essentially two different travel tasks that the user of the Virtual Habitat might wish to perform. First, general exploration of the environment needs to be supported. In this type of travel, the user is simply looking around, getting a feel for the layout, size, and features of the VE. For this purpose, a travel technique must be intuitive to the user, so that the focus can be on the environment and not on the technique. It must also allow continuous changes to the trajectory of motion, so that the user can instantaneously make course corrections. In terms of the performance metrics we have described for travel, a technique for exploration requires high levels of spatial awareness and information gathering. Ease of learning, ease of use, presence, and user comfort will also be important. Speed and accuracy are not requirements for such a technique.

Second, users may wish to travel to specific locations in the environment to obtain information. This type of travel has an explicit goal and direction, and is therefore unlike the exploration described above. It also has different requirements; in particular speed and accuracy will be quite important, since we do not wish to require the user to wait to get the desired information and we want the user to be able to move accurately to the location of the information. Since the user's focus is on the destination, not the path, spatial awareness and information gathering ability *during* travel may not be as important. Such a technique will still require moderately high levels of ease of use and user comfort.

The application needs one or more techniques for selection, including a stand-alone technique to select audio annotations for playback, and a technique to select objects for manipulation in the immersive design component. These techniques may be the same, or they may be individually considered, as was the case with the travel techniques. It is more likely here that we can find a single selection technique to do the job, since the requirements for both tasks are similar. In general, we need a technique that can be used at a reasonable distance, and which is quite intuitive and easy so that users can focus on the task at hand. In terms of performance metrics, the application requires high levels of accuracy of selection, ease of use, and user comfort, with speed also being a main consideration.

Finally, we need one or more manipulation techniques with which to accomplish the immersive design tasks (moving visual elements, for example). We need expressive techniques which can be used to place objects at any location, but which are also well-constrained and easy to use. An additional consideration is that the manipulation technique integrates well with the selection and travel techniques that are chosen. Expressiveness (the range of positions and orientations in which an object can be placed), accuracy of placement, and ease of use will be the most important requirements for designers, and speed and user comfort will be secondary considerations.

6.3 Interaction Design

We will present three levels of interaction technique design for the Virtual Habitat application, which should provide us with some measuring sticks by which we can determine the effectiveness of our formal design and evaluation methodology. The first interaction design comes from a previous application and was based on a naïve understanding of the tasks and techniques involved. The second level of design was actually implemented and tested in the virtual habitat, and is based on an intuitive understanding of the tasks and techniques, and informal evaluations of published ITs. Finally, we present an interaction design based on the results of testbed evaluation.

By looking at the usability of these three designs, we should be able to ascertain whether the process of formal design and evaluation produced any performance advantages (hypothesis 3). We will show that our final design has significant advantages in performance and usability relative to the other two interaction designs.

6.3.1 Naïve Interaction Design

The first interaction design we will consider is taken from our initial attempt at an immersive modeling system: the Conceptual Design Space (CDS) (Bowman, 1996). This system was also aimed at architectural design, but differed in that it allowed the user to create objects from scratch or modify existing models. In terms of interaction, however, its requirements were very similar to the virtual habitat application. Users needed a travel technique to specify the viewpoint and the position from which they would design, a selection technique to specify objects for manipulation or for commands, and a manipulation technique with which objects could be moved or scaled.

The CDS system used a gaze-directed steering metaphor for viewpoint motion control. That is, the user looks in the direction he wants to move and presses a button. The main reason this technique was chosen was its availability: it was the default travel technique for the underlying VE software. We made one improvement to this basic technique by including a "walking" mode, in which the user was constrained to moving in the horizontal plane at the current eye height. This allowed users to obtain more human-scale views of the objects they were modeling.

This gaze-directed technique was frustrating to many users, because many of the movement tasks in a design environment are relative motion tasks, as described earlier. That is, the user is moving to a new location in the space from which a desired view of the object under consideration can be obtained. For example, the user may wish to view a building under construction in elevation from directly in front of the building. If the user happens to be closer to the building than desired, she must turn around and move away from the building, with no idea of when to stop. This leads to a long cycle of move-stop-evaluate-correct which can frustrate users quickly. The walking mode was somewhat useful, but the fact that it was an explicit mode that had to be turned on or off was problematic. Users typically did not wish to remain in one mode or the other for long periods of time, and did not wish to issue a command to change travel mode each time they wanted to switch. Thus, walking mode was underused.

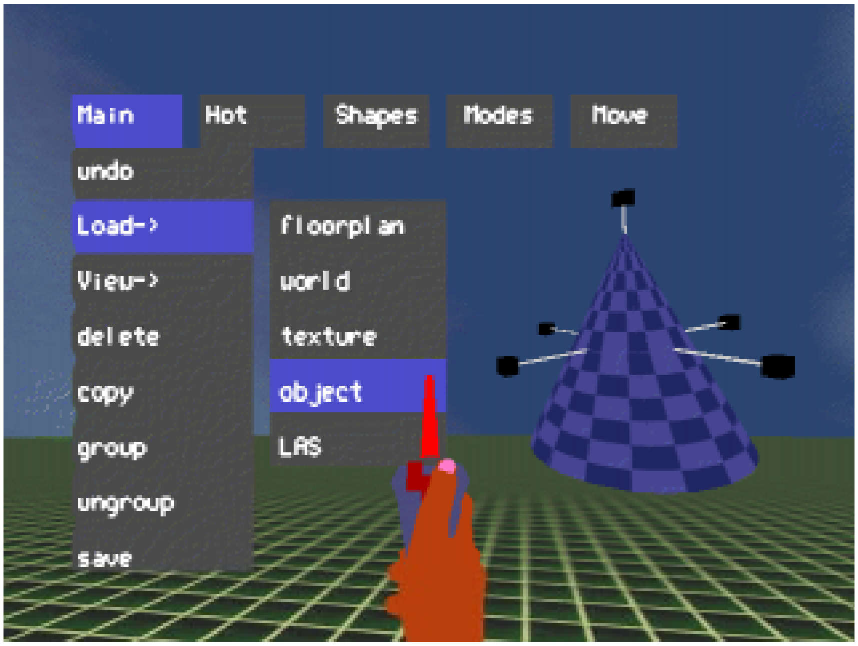


Figure 6.3 Virtual Menus in the CDS System

Selection and manipulation in CDS were based entirely on a ray-casting metaphor. A virtual light ray extended from the user's hand when a button was pressed. The light ray was used to select 3D objects, interface elements such as sliders and palettes, and object manipulation widgets. In addition, the ray was used to select items in the virtual menu system (see figure 6.3), which is similar to the one described in (Jacoby and Ellis, 1992). Menus contained commands for object creation, deletion, and copying, interface view commands, mode toggle switches, and so on. Objects could be manipulated directly with the light ray, or in a constrained manner using manipulation widgets attached to the object (figure 6.4). Depending on the mode, the user could translate, rotate, or scale the object in a constrained manner using these widgets.

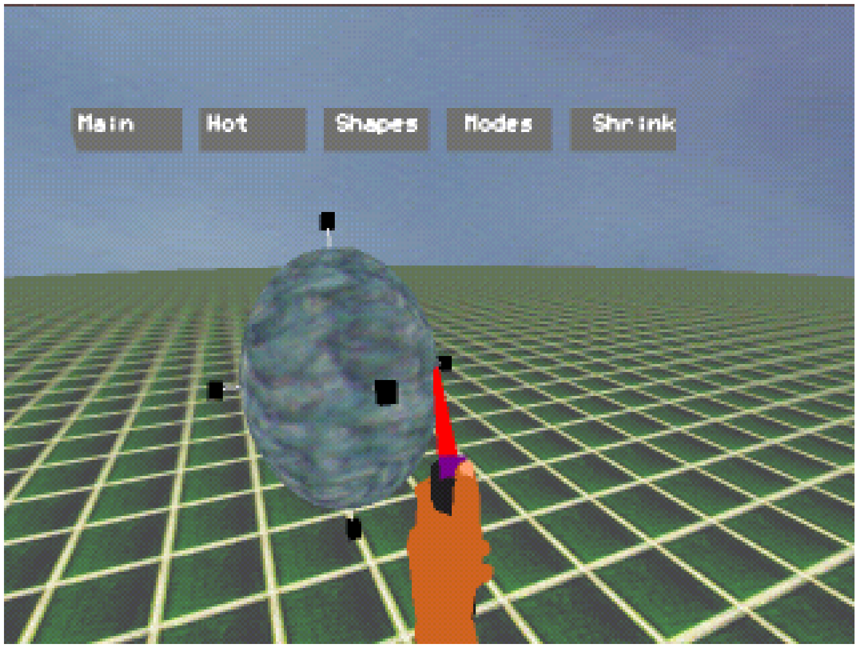


Figure 6.4 Constrained Object Manipulation in CDS with Ray-Casting

The ray-casting technique worked well in some areas, but fell short in others. Objects were easy to select, as were top-level menu items. However, items in submenus (which require precise pointing), and the small manipulation widgets were more difficult to hit with the ray. Manipulation of objects was quite imprecise when using the light ray directly, as we have already seen. Constrained manipulation was somewhat helpful, but getting an object into the desired position and orientation often took many attempts.

A usability evaluation with several graduate architecture students confirmed the advantages and disadvantages of this naïve interface. These users could see the promise of immersive design, with its immersive experience and immediate feedback, but were not very productive due to interaction issues.

6.3.2 Intermediate Design Iteration

Our second level of interaction design, based on experience, observation, informal evaluation, and the published literature, improved greatly on that of CDS. This was our initial design for interaction in the virtual habitat, which tried to provide many of the helpful constraints that were missing in CDS.

Just as virtual menus provided the system control infrastructure in CDS, we needed an overall system control scheme for the virtual habitat. We wished to avoid menus and explicit system modes based on previous experience and on general UI guidelines. Also, we wanted to avoid the imprecision of pointing in 3D space to select commands. To remedy this situation, we implemented a "pen & tablet" metaphor (Angus and Sowizral, 1995). This metaphor retains the advantages of using 2D interface elements in a 3D space (fewer DOFs to control, user familiarity, etc.), but also constrains the interaction so that it can be much more precise, efficient, and comfortable.

Figure 6.5 Physical Devices used in the Virtual Habitat Application

The physical input devices used in the pen & tablet interface are shown in figure 6.5. They consist of a physical tablet and a physical pen (or stylus), both of which are tracked in 3D space. The pen also has a single button. The tablet does not contain any electronic logic or have any display – it is simply a work surface. In the virtual environment, the user sees graphical representations of the pen and tablet, and a 2D interface is presented on the tablet surface (figure 6.6). The user interacts with this interface just as he would with a 2D interface controlled by a mouse, except that the pen is placed directly on the interface whereas a mouse indirectly controls a pointer on a screen. The interface can include menus, buttons, icons that can be dragged, and so on.

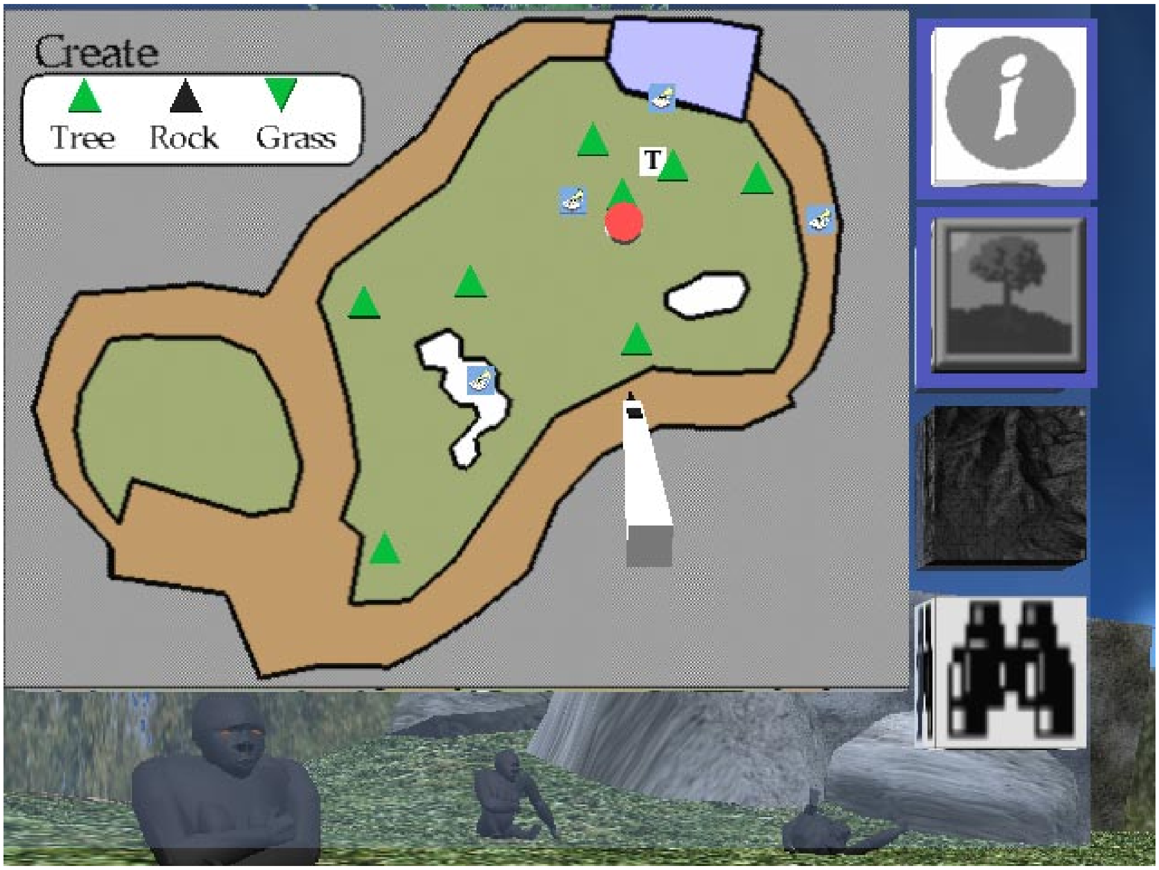


Figure 6.6 User's View of the Interface Tablet in the Virtual Habitat

The advantages of the pen & tablet metaphor are many. First, the interface is always available since the user carries it in her hand, but it can also be put away so that it does not obscure the environment (simply by placing the tablet out of the field of view). Second, the physical surface of the tablet provides an important constraint to input. Instead of pointing or gesturing in 3D space, with no guidance, the user can be assured of correct interaction as long as the tip of the pen is touching the surface of the tablet. This makes operations such as icon dragging much more precise and sure. Finally, this metaphor makes use of 2-handed interaction (Hinckley et al, 1997), where the non-dominant hand provides a frame of reference within which the dominant hand can work. This has been shown to be an efficient and effective method of 3D input.

With the pen & tablet metaphor as a basis, we began to design specific interaction techniques for the virtual habitat. Our design philosophy was to provide both tablet-based (indirect) and direct manipulation techniques for each of the major interaction tasks.

In the area of travel, we wished to support both exploration and goal-based motion, as discussed previously. For exploration, a directly controllable technique was needed. Instead of the gaze-directed technique used in CDS, we chose a pointing technique, in which the user points the stylus in the desired direction of travel. In this way, relative motion was supported, which is important for a design application. Goal-based travel was achieved on the tablet. A red dot represented the user's current position on the map of the environment which was the main feature of the tablet interface. To move quickly to a new location, the user could drag this icon to a new location on the map. The user was not moved as the drag takes place. Rather, to promote spatial awareness, the user only moved when the dragging had ended, at which time he was flown smoothly from the current location to the new one. Instead of an explicit walking mode, we chose to allow users complete 3D freedom of motion, except for a constraint on going below a certain height above the ground. Thus, users could simulate walking mode by simply pointing slightly downward, so that they traveled along at a constant height above the terrain.

Selection and manipulation of virtual objects could also be performed both directly and via the tablet. The direct technique chosen was the Go-Go technique described earlier, in which the user's virtual arm length grows at a non-linear rate as she stretches it away from her body. This allowed the user to select objects (such as trees or rocks) at a large distance, but with little cognitive load, as it is natural to stretch out one's arm to grab an object. Manipulation could then take place in the virtual hand. This type of manipulation supports more precise placement of objects. For coarse-grained placement, object icons on the tablet interface could be dragged to new locations. This was useful, for example, to create a grove of trees in one corner of the environment. We also greatly constrained manipulation to make it easier for the user. Objects always remained on the ground, and the user had no control over object rotation, since all of the objects we wished to manipulate have a natural orientation. Thus, the user was only manipulating two degrees of freedom, which matches nicely with the 2D tablet input.

There are also other selection tasks in the Virtual Habitat application. Only the Go-Go technique was used to select audio annotations for playback, as we did not wish users to be able to play annotations from anywhere (so they could experience the information in its proper context), but the tablet can be used to enable or disable specific annotations. The tablet was also used to toggle the display of various types of information on the 2D interface, to create new objects (by dragging icons onto the map), to position visitor viewpoints, and to select the terrain model. All of these are tasks which are more easily performed indirectly and/or in 2D.

A usability study was performed on this initial version of the virtual habitat application, and although it was rated quite highly, there is still room for improvement. Six teams of students used the application to modify the design of the virtual habitat for a class presentation. The usability study confirmed the usefulness and promise of immersive design, but more importantly for our research, provided us with a set of user ratings on various aspects of interaction. Users were asked to rate usability issues on a scale of one to five, with five being the most usable. A summary of the results is presented in table 6.1.

Table 6.1 Mean Usability Ratings for the Intermediate Virtual Habitat Interaction Design

Usability Categories	**Rating**
tablet: object creation	4.43
tablet: dragging user icon to move	4.21
changing terrain	4.21
moving viewpoints	4.20
moving viewpoint barriers	4.10
tablet: general interaction	3.86
tablet: object manipulation	3.86
user movement with stylus	3.71
go-go object manipulation	3.14

The entries in this table reveal some interesting trends. First, notice that when there was a choice of interaction technique (one using the tablet and one using direct, 3D manipulation), the tablet-based technique was preferred. For example, dragging the user icon on the tablet to travel to a new location in the environment was preferable to pointing in the direction of travel using the stylus. This stems from the advantages of the tablet mentioned earlier: it is always available, it has a physical work surface to constrain input, and it requires the user to control only two degrees of freedom. However, the use of the tablet also caused some problems for users, most notably due to orientation differences between the map and the environment. Some users found it difficult and disorienting to drag the user icon in one direction and then move in a different direction, or to drag an object on the tablet to the left and see it moving to the right in the virtual world. Most users were able to adapt to these difficulties by focusing on only one context at a time, and by noting relationships between object positions instead of absolute locations. For example, a user viewing the environment might decide to move a tree to the left. To make it a relative positioning task, he would translate the goal to something like "move the tree closer to the visitors building." Using this goal, either the tablet or direct manipulation methods would work well.

Feedback on the direct manipulation techniques was mixed. Some users found it natural and intuitive to point in the direction they wished to fly, and enjoyed the simplicity and flexibility of this technique. Others became disoriented when they moved in a direction other than the direction of their gaze, and could not point as accurately as they hoped.

The Go-Go technique for object manipulation fit the intuition of most users: to move an object one simply reaches towards it. However, there were difficulties due to the size of our environment. In order to allow users to reach most of the environment, the non-linear portion of the Go-Go stretching function (see Figure 5.1) needed to be quite steep. This meant that when the virtual arm was far from the user's body, a very small movement in or out would result in a large virtual hand movement. This made object selection difficult at large distances.

This usability evaluation was performed before the spatial orientation experiment and both of the testbed evaluations. Interestingly, however, these experiments would have predicted most of the major usability problems found here. Disorientation was quite harmful to users, because we used a steering technique without advising users of the proper strategies for maintaining spatial orientation. An arm-extension technique was used for object selection, which we showed empirically to be quite slow and tedious. This technique also exhibited the arm-strain characteristic that we found in our testbed evaluation.

6.3.3 Final Interaction Design

Our final design for the interaction techniques and metaphors used in the Virtual Habitat is based on the results of formal design and evaluation. Results from testbed evaluation have been applied to this system based on its requirements to show the usefulness of our methodology of formal evaluation and design. It is important to note that although we noted many other minor usability problems in our first evaluation of the Virtual Habitat, we left these things unchanged for the final iteration. The only differences in this version and the previous one are the changed techniques for travel, selection, and manipulation based on our formal evaluation. This is to ensure that any gains in usability are due to the application of our methodology, and not to other interface modifications.

The results of the travel testbed (section 4.7) showed that our intermediate design iteration actually met the application’s performance requirements well. We found that speed and other metrics on both the exploratory and the directed travel tasks was best with continuous steering techniques, such as pointing. Although this was intended in the previous design iteration to be used for exploration, it appears to be well suited to the performance requirements of the goal-directed travel task. User comfort was not a major factor in the testbed experiment, but the pointing technique performed well in this category.

In our previous usability study, the map dragging technique was rated subjectively higher than the pointing technique. However, we noted some problems with it, and these problems were verified in the testbed evaluation. Most notably, users often did not know which direction to drag the user icon in order to move to a given location. In the usability study, we found that certain users were better with the map technique than others, and hypothesized that these people were able to do the mental rotations of the map necessary to determine direction. Therefore, we left the map dragging technique in place in the final design, but only encouraged users to utilize it after they are quite familiar with the spatial layout of the habitat.

A related usability problem that we found in the intermediate design iteration concerned the loss of spatial orientation on the part of users. Users often became lost or disoriented, especially after using the pointing technique to fly in a direction other than that of their gaze. Some users also had difficulty relating the static map information to the dynamic environment. These are exactly the concerns addressed by our spatial orientation experiment in section 4.6.3. In that evaluation, we found that subjects who used advanced strategies for maintaining orientation had the best performance. Therefore, in the final design iteration for this application, we modified our written and verbal instructions in order to train users in these strategies. Strategies relevant to the Virtual Habitat include 3D overview (fly up above the environment to get a survey view), backing in (moving backwards to a destination so that it is placed in the context of previously visited areas), proprioceptive pointing (reminding oneself of the location of known objects), stop & look (pausing to look around at the current location), and path retracing (moving again along previously traveled paths, often from a different direction). Users are not likely to use all of these strategies, but using one or more of them could increase spatial orientation.

The selection and manipulation testbed confirmed our informal observations of the Go-Go technique. It is not well-suited for selection of objects that are small and/or far away. Moreover, it was the lowest rated of the techniques in our usability study, due to the frustration people had with selecting distant objects. The testbed results showed that the HOMER technique was the best fit for the performance requirements specified above for selection and manipulation. It can select objects well at long distances, and ray-casting is quite easy to use and speedy. The manipulation component of HOMER is very expressive and also easy to use and moderately fast, according to the empirical results. HOMER was not near the top of the rankings for manipulation time in our study, but as stated above, speed of manipulation is not a key performance requirement of the Virtual Habitat.

Having chosen these techniques for our final implementation, we were faced with another problem: the stylus has only a single button, but both the pointing technique for travel and the HOMER technique for selection and manipulation would need that button. We implemented a solution that we felt would be easy for users to understand and use. Our implementation changes the use of the button depending on how long it is held down. The light ray is visible at all times, and objects are highlighted when intersected by the ray. If the user clicks the button (down and up) quickly (less than 0.7 seconds) and an object is highlighted, that object is selected. If the user holds down the button for more than 0.7 seconds, the ray disappears and the user begins to travel using the pointing technique. The single button also precluded us from using the indirect depth manipulation technique we studied in our experiment.

Several lessons can be gleaned from this design iteration. First, the technique that users prefer is not always the one with the best performance. Users preferred the map technique to pointing, but empirical evaluation showed pointing to be faster. Fortunately, we could include both techniques in our application. Also, when attempting to support better performance by using empirically proven interaction techniques, the tradeoffs and difficulties of integration must be taken into account. In our case, the usability problems with the intermediate iteration were severe enough that we were willing to work through the integration problems to solve them.

6.4 Final Usability Evaluation

When the interaction design was finalized, a new usability study was performed under similar circumstances and using the same evaluation metrics (interviews and usability ratings). In this way, we compared the usability of a system designed using intuition and observation to that of a system implemented based on formal evaluation and design methods (hypotheses 1 and 3). This study would validate the use of our formal design and evaluation methodology if increased performance were found.

Five user sessions were held, lasting for sixty to ninety minutes each. During the session, the users were instructed on the use of the techniques, allowed to explore the virtual habitat, shown how to access the information embedded in the environment, and presented with the design tools. Each user or group of users (users came singly or in a group of two) spent twenty to forty minutes using the design tools to modify the design of the gorilla habitat. Subjects were members of an undergraduate design class with experience in both traditional and computer-aided design. At the end of the session, each user or group was asked for their comments and observations on the system, as well as a set of usability ratings on the various features of the application. These ratings again were on a five-point scale, with five representing high usability. A summary of the results is presented in table 6.2, including average usability ratings and standard deviations for each of the system's features.

Table 6.2 Mean Usability Ratings (standard deviations in parentheses) for the Final Virtual Habitat Interaction Design

(features changed since the previous iteration)*

Usability Categories	Final iteration	Intermediate iteration
selecting annotations*	4.70 (0.45)	N/A
changing terrain	4.20 (0.76)	4.21 (1.15)
user movement with stylus*	4.10 (0.89)	3.71 (1.11)
tablet: dragging user icon to move*	4.10 (0.74)	4.21 (0.81)
direct object manipulation*	4.00 (0.35)	3.14 (1.18)
tablet: object creation	4.00 (0.71)	4.43 (0.53)
moving viewpoints	3.55 (0.94)	4.20 (0.84)
tablet: object manipulation	3.50 (1.00)	3.86 (0.94)
moving viewpoint barriers*	3.40 (1.39)	4.10 (1.02)
tablet: general interaction	2.90 (0.89)	3.86 (0.90)

The most important result from table 6.2 is that our application of the results of formal design and evaluation had positive results on reported usability. This is most easily seen for the direct object manipulation feature, which was changed from the Go-Go technique to the HOMER technique, and which received a much higher usability rating in the final iteration. This is despite the fact that this group of users seemed to have a lower baseline rating overall (for all unchanged components, the average usability rating was lower than the corresponding rating from the intermediate iteration. Also, ray-casting proved to be very easy to use as a selection mechanism for the audio annotations, receiving the highest rating of any feature. Although we did not measure the usability of the Go-Go technique for annotation selection in the previous iteration, it was the source of many verbal usability complaints by users.

Second, we note that the reported usability of the pointing technique was improved in the final iteration. Although the implementation of this technique did not change, the training given to users in the proper use of this technique was modified. Both written and verbal instructions were given to users telling them how to use this technique to maintain spatial orientation (e.g. flying upwards to get a survey view of the environment). This result validates our earlier finding that the training of specific strategies can have an effect on overall performance.

The map dragging technique for travel was rated highly, but slightly lower than the rating from the previous iteration. Again, this is consistent with other features that remained unchanged. Therefore, the additional training in strategies for spatial orientation did not increase the usability of this technique, again validating our earlier findings. Strategy sophistication can increase performance with steering techniques, but performance using target-specification techniques is relatively constant no matter what strategies are used. Also, fewer of the strategies are possible when using the map-dragging technique.

The comment of one subject is particularly enlightening with regard to the travel techniques used in this system. Although the map technique performed poorly in the testbed evaluation and is not useful on its own, it can be a good complement to a steering technique. The subject stated that he would not rate the map technique highly, except that it worked well in conjunction with the pointing technique. This leads to the general principle that multiple, redundant interaction techniques should sometimes be used to improve usability.

The only disappointment in this study was the use of HOMER to manipulate the foliage barriers at visitor viewpoints, which received a very low rating. From user comments, we feel this was due to the nature of the task. The barriers are very close to the user at the viewpoints. Since HOMER maps the body-hand distance to the body-object distance to determine the mapping between hand and object motion, near objects are difficult to move farther away. With the Go-Go technique, the same depth range can be accessed regardless of the original object distance. An indirect depth specification scheme using buttons would solve this problem, but is not possible with our single-button stylus.

The use of subjective ratings to measure usability is somewhat problematic, as we have no measure of the validity or reliability of this metric. The same is true for the comfort ratings used in the testbed experiments. We do have information on the variability of these ratings, which seems to be reasonable, but the results would be more powerful if usability or other types of performance had been measured with a proven metric, whether quantitative or qualitative. We leave the development of such a metric as future work.

On the whole, this usability study provided an unequivocal endorsement of our methodology. The use of the formal design and evaluation framework, testbed evaluation, and application of results based on performance specification caused a measurable increase in usability, supporting hypothesis 3.

CHAPTER VII

CONCLUSIONS AND FUTURE WORK

In this research, we have obtained a large body of results pertaining to the performance of interaction techniques for universal tasks in immersive virtual environments. These results are useful in choosing the appropriate techniques for VE applications, given their interaction requirements. We have also produced several new techniques, using our methodology, that provide more options to VE developers. This work also resulted in a VE application that we have shown to be both useful and usable for environmental designers.

Beyond these immediate results, however, our research has also produced some more abstract and high-level improvements in our understanding of the design and evaluation of VE interaction. Here, we will briefly discuss several of these important contributions.

7.1 VE Interaction Guidelines

In practical reality, few application developers are likely to take the time and effort required to quantify the interaction requirements of their systems, compare these to the results of testbed evaluation, and choose a set of ITs in a systematic fashion. One solution to this would be to create an interactive system that would accept a set of application requirements and automatically suggest possible ITs that match those requirements (discussed in the section on future work, below).

However, there is a well-established tradition in the HCI community of publishing sets of *guidelines* for user interfaces, interaction techniques, and the like. Guidelines are principled, practical aids that help a designer create interaction that is usable and performs well. Guidelines for VE interaction are not new (e.g. Kaur, 1999), but most sets of guidelines have two drawbacks. First, they are too general and subject to interpretation. They do not reduce the space of possible techniques far enough to allow the developer to make an informed decision. Second, guidelines have been simply adapted from 2D HCI guidelines, or they come from experience and intuition only. This does not ensure that the guidelines will be sound or that their use will produce well-performing systems.

Therefore, the VE community needs a set of interaction guidelines that are specific and practical, and which come directly from evaluation of techniques in the laboratory and in deployed systems (hypothesis 2). Our experiments and usability studies are a valuable source of such guidelines, and we present some of them here. Although all of these guidelines can be found elsewhere in the text, it is useful to view them together here.

7.1.1 Generic VE Interaction Guidelines

Do not assume that techniques based on a natural, real-world metaphor will be the most intuitive or that they will have the best performance.

Our initial user study on selection and manipulation techniques showed that techniques closer to the natural mapping often exhibited serious usability problems. Testbed evaluation has confirmed this fact. Therefore, the use of “magic” techniques which differ greatly from the natural mapping (but which may still take advantage of well-developed human skills) is an important principle for VE interaction. Natural interaction techniques may still be useful, especially in situations where the VE is used as training for a real-world task, or where the target user population has no VE experience and will only use the VE for a short time.

Provide redundant interaction techniques for a single task.

One of the biggest problems facing evaluators of VE interaction is that the individual differences in user performance seem to be quite large relative to 2D interfaces. Some users seem to comprehend complex techniques easily and intuitively, while others may never become fully comfortable. Work on discovering the human characteristics that cause these differences is ongoing, but one way to mitigate this problem is to provide multiple interaction techniques for the same task. For example, one user may think of navigation as specifying a location within a space, and therefore would benefit from the use of a technique where the new location is indicated by pointing to that location on a map. Another user may think of navigation as executing a continuous path through the environment, and would benefit from a continuous steering technique. In general, “optimal” interaction techniques may not exist, even if the user population is well known, so it may be appropriate to provide two or more techniques each of which have unique benefits. Of course, the addition of techniques also increases the complexity of the system, and so this must be done with care and only when there is a clear benefit.

7.1.2 Guidelines for the Design of Travel Techniques

Make simple travel tasks simple by using target-specification techniques.

If the goal of travel is simply to move to a new location, such as moving to the location of another task, target-based techniques provide the simplest metaphor for the user to accomplish this task. In many cases, the exact path of travel itself is not important; only the end goal is important. In such situations, target-based techniques make intuitive sense, and leave the user’s cognitive and motor resources free to perform other tasks. The use of target-based techniques assumes that the desired goal locations are known in advance or will always coincide with a selectable position in the environment. If this is not true (e.g. the user wishes to obtain a bird’s-eye view of a building model), target-based techniques will not be appropriate.

Avoid the use of teleportation; instead, provide smooth transitional motion between locations.

Teleportation, or "jumping," refers to a target-based travel technique in which velocity is infinite – that is, the user is moved immediately from the starting position to the target. Such a technique seems very attractive from the perspective of efficiency. However, evaluation (Bowman et al, 1997) has shown that disorientation results from teleportation techniques. Interestingly, all techniques that used continuous smooth motion between the starting position and the target caused little disorientation, even when the velocity was relatively high.

If steering techniques are used, train users in strategies to acquire survey knowledge. Use target-specification or route-planning techniques if spatial orientation is required but training is not possible.

Spatial orientation (the user's spatial knowledge of the environment and her position and orientation within it) is critical in many large-scale VEs, such as those designed to train users about a real world location. The choice of interaction techniques can affect spatial orientation. In particular, evaluation (Bowman, Davis, Hodges, & Badre, 1999) has shown that good spatial orientation performance can be obtained with the use of steering techniques, where the user has the highest degree of control, but only if sophisticated strategies are used (e.g. flying above the environment to obtain a survey view, moving in structured patterns). If such strategies are not used, steering techniques may actually perform worse, because users are concentrating on controlling motion rather than viewing the environment. Techniques where the user has less control over motion, such as target-based and route-planning techniques, provide moderate levels of spatial orientation due to the low cognitive load they place on the user during travel – the user can take note of spatial features during travel because the system is controlling motion.

Constrain the user's travel to fewer than three dimensions if possible to reduce cognitive load.

Our information gathering experiment (Bowman, Koller, & Hodges, 1998) showed that the higher the dimensionality of the path the user travels, the more likely he is to forget information seen along that path. Many VE applications allow the user to fly in three dimensions, even when it is not necessary. A simple constraint that keeps the user on the ground plane should reduce cognitive load. The use of this guideline, however, must be tempered with the fact that 3D flying may also increase spatial orientation if used correctly.

Use non-head-coupled techniques for efficiency in relative motion tasks. If relative motion is not important, use gaze-directed steering to reduce cognitive load.

Relative motion is a common VE task in which the user wishes to position the viewpoint at a location in space relative to some object. For example, an architect wishes to view a structure from the proposed location of the entrance gate, which is a certain distance and direction from the front door – movement must be relative to the door, and not to any specific object. A comparison of steering techniques (Bowman, Koller, and Hodges, 1997) showed that a pointing technique performed much more efficiently on this task than gaze-directed steering, because pointing allows the user to look at the object of interest while moving, while gaze-directed steering forces the user to look in the direction of motion. Gaze-directed steering performs especially badly when motion needs to be in the opposite direction from the object of interest. Thus, techniques that are not coupled to head motion support relative motion tasks. On the other hand, non-head-coupled techniques are slightly more cognitively complex than gaze-directed steering, so it may still be useful if relative motion is not an important task.

7.1.3 Guidelines for the Design of Selection Techniques

Use ray-casting techniques if speed of remote selection is a requirement.

Evaluation (Bowman & Hodges, 1999) has shown that ray-casting techniques perform more efficiently than arm-extension techniques over a wide range of possible object distances, sizes, and densities. This is due to the fact that ray-casting selection is essentially 2D (in the most common implementation, the user simply changes the pitch and yaw of the wrist). Ray-casting includes both the virtual light ray metaphor and image plane techniques such as occlusion and framing.

Ensure that the chosen selection technique integrates well with the manipulation technique to be used.

Selection is most often used to begin object manipulation, and so there must be a seamless transition between the selection and manipulation techniques to be used in an application. Arm-extension techniques generally provide this transition, because the selected object is also manipulated directly with the virtual hand, and so the same technique is used throughout the interaction. As demonstrated by the HOMER technique, however, it is possible to integrate ray-casting techniques with efficient manipulation techniques.

If possible, design the environment to maximize the perceived size of objects.

Selection errors are affected by both the size and distance of objects, using either ray-casting or arm-extension techniques (Bowman & Hodges, 1999). These two characteristics can be combined in the single attribute of visual angle, or the perceived size of the object in the image. Unless the application requires precise replication of a real-world environment, manipulating the perceived size of objects will allow more efficient selection.

7.1.4 Guidelines for the Design of Manipulation Techniques

Reduce the number of degrees of freedom to be manipulated if the application allows it.

Provide general or application-specific constraints or manipulation aids.

These two guidelines address the same issue: reducing the complexity of interaction from the user's point of view. This can be done by considering the characteristics of the application (e.g. in an interior design task, the furniture should remain on the floor), by off-loading complexity to the computer (using constraints or physical simulation), or by providing widgets to allow the manipulation of one or several related DOFs (Mine, 1997).

Allow direct manipulation with the virtual hand instead of using a tool.

Tools, such as a virtual light ray, may allow a user to select objects from great distances. However, the use of these same tools for object manipulation is not recommended, due to the fact that positioning and orienting of the object is not direct – the user must map desired object manipulations to the corresponding tool manipulations. Manipulation techniques that allow the direct positioning and orienting of virtual objects with the user's hand have been shown empirically (Bowman & Hodges, 1999) to perform more efficiently and to provide greater user satisfaction than techniques using a tool. For efficient selection and manipulation, then, we need to combine a 2D selection metaphor such as ray-casting with a hand-centered, direct manipulation technique. This is the basis of techniques such as HOMER and Sticky Finger (Pierce et al, 1997).

Avoid repeated, frequent scaling of the user or environment.

Techniques that scale the user or the world to allow direct manipulation have some desirable characteristics. The user's perception of the scene does not change at the moment of selection, and small physical movements can allow large virtual movements. However, experimental data (Bowman & Hodges, 1999) shows a correlation between the frequent use of such techniques and discomfort (dizziness and nausea) in users. Techniques that scale the user or environment infrequently and predictably should not suffer from these effects.

Use indirect depth manipulation for increased efficiency and accuracy.

Indirect control of object depth, using joystick buttons for example, is not a natural technique (although it borrows from a real-world "fishing reel" metaphor), and requires some training to be used well. However, once this technique is learned, it provides more accurate object placement, especially if the target is far from the user (Bowman & Hodges, 1999). This increased accuracy leads to more efficient performance as well. Moreover, these techniques do not exhibit the arm strain that can result from the use of more natural arm-extension techniques.

7.2 Formal Design & Evaluation Frameworks

A second major contribution of this work is the framework and methodology we proposed and used in all of the design and evaluation components of the research. The methodology includes the use of taxonomy, guided design, multiple performance metrics, consideration of outside factors on performance, and testbed evaluation.

Such a framework has several advantages. First, formalism is a great aid to understanding. In order to create a useful and believable taxonomy, for example, it was necessary to study and consider both the interaction task and the techniques proposed for that task. Second, the use of the methodology in multiple experiments allows us to view all of the results within a common framework. For example, we know many of the relative merits of the common steering techniques gaze-directed steering and pointing due to the multiple evaluations. Third, the framework provides a common ground for discussion among researchers in the field, allowing more precise and well-understood conversations.

Finally, special mention needs to be made of the utility of guided design in creating new interaction technique possibilities. We have shown that combining previously untried sets of components can produce useful and interesting techniques (such as HOMER). Furthermore, there seems to be a slowing of the publication of completely novel interaction techniques and metaphors for immersive environments. It is possible, though certainly not proven, that we have identified many, or most, of the fundamental components of VE interaction for these universal tasks. If so, then guided design becomes the best method for covering the design space.

7.3 Focus on Applications and Usability

A third major contribution of this research has been our focus from the beginning on improving the usability (and more generally, the performance) of immersive VE applications. As we noted in the first chapter, there are very few in-use applications of VEs due to the usability problems associated with high levels of interactivity. Therefore, this research has not been simply an academic exercise. It has had as its goal from the beginning to improve VE interaction for real-world applications. This led us to a methodology that included applications and their requirements explicitly.

There have been other efforts to quantify the performance of VE interaction techniques, but few of them have extended this work to real applications. On the other hand, a large number of applications have been prototyped, but interaction was developed in an ad hoc manner, based on intuition. This work has bridged the gap, providing empirical evidence and practical guidelines for real applications based on formal evaluation.

We believe that this philosophy of research will be fruitful in other types of virtual environments, such as tabletop stereo displays, and in many emerging areas of interactive systems, such as augmented reality, ubiquitous computing, and wearable computing. Because of their newness, such areas need empirical, low-level studies to quantify performance and effectiveness. However, these research areas are also under pressure to produce real applications to prove that the research funding is worthwhile. Using the philosophy embodied in this thesis, which we call "basic research with an applied focus," can allow both of these things to happen in the same research program.

7.4 Future Work

Research in a relatively new area usually raises more questions than it answers, and this work is no exception. There are a multitude of topics in the general area of VE interaction that still need to be explored in depth. In particular, there are four areas directly related to the current work that we claim would be extremely useful.

7.4.1 Automatic Interaction Design and Performance Modeling

Our testbed evaluations and other experiments have produced a large body of empirical results for IT performance on various tasks. However, it is still difficult for application designers to wade through these numbers in order to choose an appropriate set of techniques for a particular system. Therefore, it would be useful to create a tool that automates some of this process for the developer. Such a tool would likely ask the developer a series of questions about the application, including what tasks were involved, what requirements existed for the various aspects of performance, and what devices were available. It could then, based on evaluation results, suggest a set of interaction techniques that would fit the requirements.

This leads to another problem, however, in that such a tool would only be able to suggest the use of techniques that had actually been tested experimentally. It would be more useful if the tool could predict the performance of an untested technique by interpolating the results from related techniques. Fortunately, our taxonomy and framework is set up to allow the creation of these predictive models of performance.

Consider a simple example. A task has two subtasks, each of which has two components. The components are numbered one through four (figure 7.1). An experiment found that technique A, composed of components 1 and 3, scored 5.0 on a certain metric; technique B, composed of components 1 and 4, scored 10.0; and technique C, composed of 2 and 3, scored 8.5. A simple prediction algorithm would guess that component 2 is responsible for 1.5 units more than component 1 (based on the scores of techniques B and C). So, to predict the score for technique D, composed of components 2 and 3, we can take the score for technique A (1 and 3), and add 1.5, for a score of 6.5. The same result would be obtained if we first determined the contribution of component 3 relative to 4 (3.5 units less), and then added this to the technique B score (10.0-3.5 = 6.5).

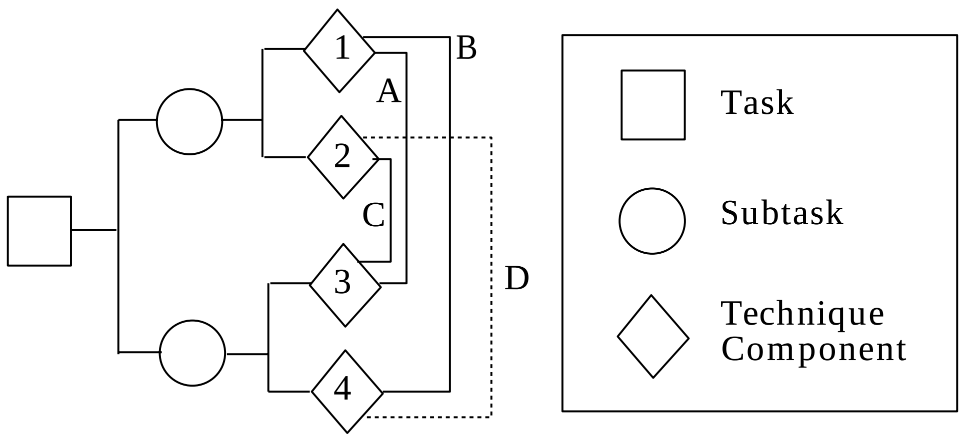


Figure 7.1 Example Taxonomy and Technique Components: If Performance Results for Techniques A, B, and C are Known, the Performance of Technique D can be Inferred

With more complex results, such a simple prediction is not possible, but the same concept holds. Regression or other types of analysis of the experimental data would lead to predictive models that would predict the performance of any technique which falls within the space defined by the techniques actually tested.

7.4.2 Cross-task Interaction Techniques

In this work, we have found a number of times that a technique originally designed for one task is useful for another task, with slight modifications. For example, the route-planning technique for travel actually uses manipulation of objects in a small version of the environment, similar to the World in Miniature (WIM) technique (Pausch et al, 1995).

This concept can be generalized when one realizes that all three of the universal tasks have as their basis the specification of a spatial position and/or orientation. Travel sets the position and orientation of the viewpoint, manipulation does the same for an object, and selection can be thought of as specifying the position of an object as a naming mechanism. This means that we can consider a technique designed for any one of the tasks as a possible technique for any of the others. We call these “cross-task” interaction techniques, because they cross the boundaries between the tasks.

In fact, many such techniques have already been developed, most of which use manipulation techniques to effect travel. There are other possibilities, however, such as using travel for object manipulation (the user “becomes” the object and sets the position and orientation from a first-person point of view), or using object selection for manipulation (place the object in my hand next to the selected object). This analysis also implies that the taxonomies for the three tasks are actually linked together, creating a single unified design space, as shown in figure 7.2.

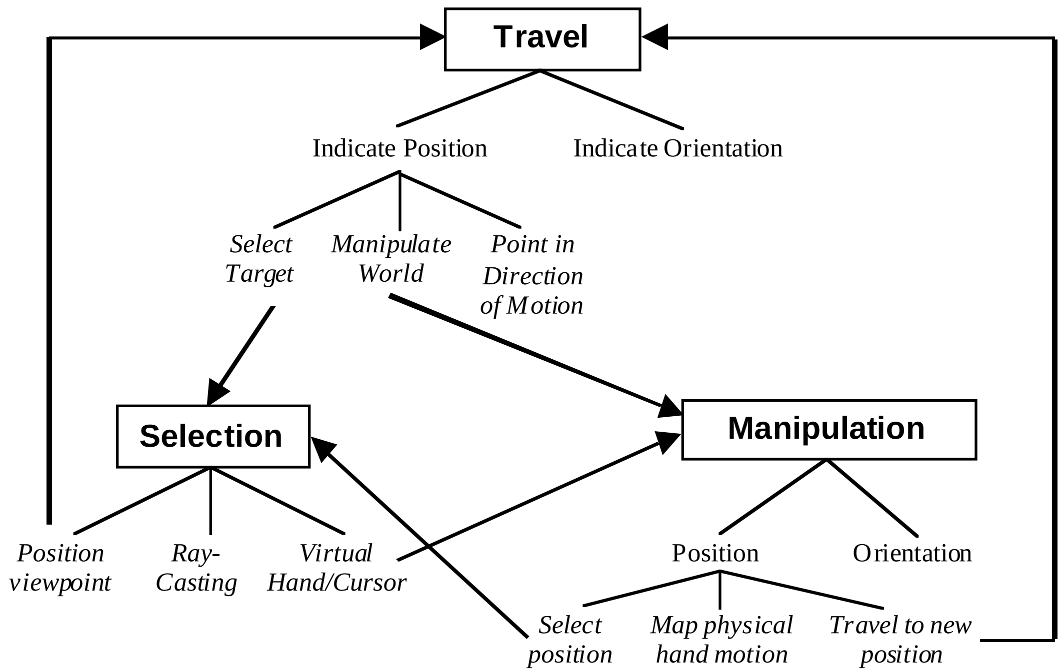


Figure 7.2 Simplified Taxonomies Linked Together by Cross-Task Techniques

We believe that cross-task interaction techniques can be useful and powerful in VEs. In particular, they have the advantage that the same metaphor may be used for multiple tasks, increasing the consistency of the interface and reducing the amount of complexity with which the user has to cope. Further research into such techniques should prove fruitful.

7.4.3 Comparison with Usability Engineering

Our design, evaluation, and application methodology has proven to be useful in increasing our understanding of VE interaction and in increasing the usability and performance of a specific VE application. However, our methodology is not the only way to improve system usability. One particular method that has received attention recently is usability engineering.

Usability engineering has a tradition in 2D HCI research, and has now been applied to VEs (Gabbard & Hix, 1998). The basic approach is centered on a particular VE system, and the iterative design and evaluation of the interaction and interface in that system. Like our methodology, it relies on a formal task and user analysis. Unlike our techniques, it uses more qualitative performance metrics, and performs evaluation within the system rather than in a generalized testbed.

Obviously, these two methodologies each have their advantages and disadvantages, and it is likely that they are complementary techniques. However, it would be instructive to do a controlled comparison of the two to determine where most of the gains in performance and usability come from. We would hazard to guess that neither method alone is sufficient. Usability engineering will not work unless it begins with a set of possible interaction techniques that have good performance characteristics, and our methodology will likely produce an application that would still benefit from iterative design and evaluation.

7.4.4 Interaction in Other Display Modalities

Finally, our work has focused solely on immersive VEs that are implemented using head-mounted displays. While this is still the most common VE display device, it has fallen out of favor in some circles, and other displays such as tabletop stereo displays and spatially immersive displays (e.g. the CAVE™) are being widely tested.

However, the VE community has no notion of how these various display modalities differ or what applications or tasks for which each is appropriate. Some vague notions exist based on intuition and limited experience, but for the most part a given display is used simply because it is available.

The studies we have presented have some generality, and the principles derived from them can be applied in a variety of VEs. On the other hand, interaction in the other display modalities is likely to be somewhat different from interaction in an HMD-based VE, and so further work in this area is needed. In particular, it would be interesting to study whether the relative performance of various ITs changes as we move to a new display modality. A study of task appropriateness in the different modalities would also be instructive.

APPENDIX A

STANDARD USER QUESTIONNAIRE

Please tell us about your background by answering these questions. Feel free to add comments to clarify your answers. If you need extra space, you may use the back of the page.

1. Specify your job title, if any. If you are a student, indicate your class and major.

2. What is your age? __________

3. Are you:

a) male   b) female

4. Are you:

a) right-handed   b) left-handed   c) ambidextrous

5. How often do you use a computer? (Circle the best answer)

a) Daily   b) A few times a week   c) A few times a month   d) Rarely or never

6. What computer platform(s) are you familiar with? (Circle all that apply)

a) PC
b) Macintosh
c) UNIX workstations
d) Other ____________________________________

7. Which, if any, of these input devices are you familiar with? (Circle all that apply)

a) keyboard
b) mouse
c) joystick
d) touch screen

e) pen/stylus (e.g. Apple Newton, PalmPilot)
f) drawing tablet
g) 3D input devices (e.g. trackers, 3D mice)
h) Other ___________________________________

8. Have you ever used virtual reality (VR) or a virtual environment (VE) which used a head-mounted display? ___________

If so, please describe the system and the input devices used below (use back if necessary):

APPENDIX B

COMFORT RATINGS FORM

1 = normal conditions (comfortable)
5 = moderate discomfort
10 = extreme discomfort

After VE familiarization:

arm strain:	1	2	3	4	5	6	7	8	9	10
hand strain:	1	2	3	4	5	6	7	8	9	10
dizziness:	1	2	3	4	5	6	7	8	9	10
nausea:	1	2	3	4	5	6	7	8	9	10

After initial practice:

arm strain:	1	2	3	4	5	6	7	8	9	10
hand strain:	1	2	3	4	5	6	7	8	9	10
dizziness:	1	2	3	4	5	6	7	8	9	10
nausea:	1	2	3	4	5	6	7	8	9	10

After segment 1:

arm strain:	1	2	3	4	5	6	7	8	9	10
hand strain:	1	2	3	4	5	6	7	8	9	10
dizziness:	1	2	3	4	5	6	7	8	9	10
nausea:	1	2	3	4	5	6	7	8	9	10

After segment 2:

arm strain:	1	2	3	4	5	6	7	8	9	10
hand strain:	1	2	3	4	5	6	7	8	9	10
dizziness:	1	2	3	4	5	6	7	8	9	10
nausea:	1	2	3	4	5	6	7	8	9	10

After segment 3:

arm strain:	1	2	3	4	5	6	7	8	9	10
hand strain:	1	2	3	4	5	6	7	8	9	10
dizziness:	1	2	3	4	5	6	7	8	9	10
nausea:	1	2	3	4	5	6	7	8	9	10

APPENDIX C

COMPLETE RESULTS OF THE TRAVEL TESTBED EXPERIMENT

Table C.1 Results of Primed Search Task

Condition	Technique	Mean ThT	Std Dev ThT	Mean TrT	Std Dev TrT
R5V0	Gaze-directed	1.64	0.63	0.11	0.03
	Pointing	2.11	1.35	0.11	0.02
	Torso	2.29	1.16	0.25	0.21
	HOMER	3.88	2.17	0.23	0.08
	Map	23.92	12.69	0.26	0.11
	Ray-casting	2.57	2.48	0.35	0.24
	Go-Go	3.38	1.81	0.20	0.11
R5V1	Gaze-directed	1.35	0.63	0.05	0.01
	Pointing	2.04	1.06	0.05	0.01
	Torso	1.49	0.34	0.06	0.01
	HOMER	2.96	1.58	0.15	0.07
	Map	12.55	2.66	0.16	0.07
	Ray-casting	1.85	1.79	0.11	0.09
	Go-Go	1.84	1.12	0.05	0.01
R10V0	Gaze-directed	1.73	1.11	0.10	0.05
	Pointing	2.49	2.23	0.10	0.04
	Torso	3.61	2.59	0.21	0.18
	HOMER	3.82	1.43	0.30	0.15
	Map	17.24	12.51	0.24	0.07
	Ray-casting	1.61	0.48	0.24	0.13
	Go-Go	1.95	1.18	0.15	0.09
R10V1	Gaze-directed	1.62	0.90	0.05	0.01
	Pointing	2.02	1.28	0.06	0.01
	Torso	1.30	0.13	0.06	0.01
	HOMER	2.38	1.98	0.12	0.04
	Map	15.48	10.93	0.21	0.12
	Ray-casting	1.98	0.56	0.16	0.08
	Go-Go	1.61	0.81	0.06	0.03

Notes: For naïve search results, see table 4.3. R5 refers to trials with required accuracy radius of 5 m; R10 refers to trials with required accuracy radius of 10 m. V0 refers to trials with target not visible from start location; V1 refers to trials with target visible from start location. ThT refers to cognitive/perceptual processing (or thinking) time. TrT refers to travel time. Travel time is normalized: time per 100 meters of travel.

Table C.2 Demographic and Comfort Rating Summary

Technique	Gaze	Pointing	Torso	HOMER	Map	Ray-cast	Go-Go
# Left-Handed	0	0	0	0	0	0	0
# Females	0	1	2	0	0	0	2
Avg. Age	18	20.2	20.2	20	19	18.8	21
# Experienced VEs	1	0	1	1	1	1	0
Avg. SA score	7	8.8	9.4	9.8	8.8	8.4	12.6
Arm 1	1.6	1.2	1	2.4	1.4	1	1.6
Hand 1	1.2	1.2	1	1.4	2.6	1	1.4
Dizzy 1	3.6	2	3	2.6	2.6	1.2	2.4
Nausea 1	1	1.8	1.2	1.8	2.2	1	2
Arm 2	2.4	1	1	2.2	1.8	1.2	2.2
Hand 2	1.8	1	1	1.6	2.6	1	1.8
Dizzy 2	2.8	2	2.8	2.2	3.2	1.2	3
Nausea 2	1.4	1.6	1.4	1.8	3	1.2	2.4
Arm 3	2.4	1	1	2.8	2	1.4	2.2
Hand 3	2.4	1	1	1.6	2.6	1.2	1.4
Dizzy 3	2.8	1.8	2.2	2.6	3.4	1.2	3.2
Nausea 3	1.8	1.4	1.4	2.2	3.4	1.2	2.8
Arm 4	2	1.2	1	3.2	2.4	1.2	2.2
Hand 4	1.8	1.2	1	2	2.8	1	1.4
Dizzy 4	3.6	1.8	2.8	2.8	3.8	1.4	3
Nausea 4	2.2	1.2	1.4	2	3.4	1.4	2.6

Notes: VE Experience refers to any use of an immersive VE system prior to the experiment. SA score refers to the average score on the cube comparison test of spatial ability (maximum score 21).

APPENDIX D

COMPLETE RESULTS OF THE SELECTION/MANIPULATION TESTBED EXPERIMENT

Table D.1 Speed Results for Selection Task

Condition	Selection Technique	Selection Time	Standard Deviation of Time
D0S0N0	Go-Go	6.89	1.68
	Ray-casting	3.78	1.62
	Occlusion	2.77	1.44
D0S0N1	Go-Go	5.33	1.54
	Ray-casting	3.88	1.27
	Occlusion	3.82	1.26
D0S1N0	Go-Go	3.85	0.96
	Ray-casting	2.19	0.57
	Occlusion	2.34	0.57
D0S1N1	Go-Go	3.41	0.89
	Ray-casting	2.69	1.40
	Occlusion	2.67	0.86
D1S0N0	Go-Go	8.60	4.45
	Ray-casting	3.43	1.30
	Occlusion	4.11	1.81
D1S0N1	Go-Go	5.74	1.75
	Ray-casting	4.08	1.72
	Occlusion	4.18	1.41
D1S1N0	Go-Go	5.09	1.62
	Ray-casting	3.32	1.94
	Occlusion	4.32	2.88
D1S1N1	Go-Go	4.43	0.94
	Ray-casting	2.68	0.88
	Occlusion	3.05	1.24
D2S0N0	Go-Go	12.34	10.05
	Ray-casting	4.02	1.57
	Occlusion	4.18	1.09
D2S0N1	Go-Go	11.70	10.80
	Ray-casting	3.34	1.08

	Occlusion	3.98	1.21
D2S1N0	Go-Go	4.38	1.59
	Ray-casting	2.75	1.31
	Occlusion	3.61	2.16
D2S1N1	Go-Go	7.06	1.30
	Ray-casting	3.19	1.37
	Occlusion	5.83	7.30

Notes: D0, D1, D2 refer to the three levels of distance of the objects from the user. S0 and S1 refer to the two sizes of objects. N0 and N1 refer to the two levels of density of the object array.

Table D.2 Speed Results for Manipulation Task

Condition	Attach/Manip Technique	Manipulation Time	Std. Dev. for Time
D0S0F0	Go-Go	5.43	1.50
	Move hand/linear mapping	5.28	1.04
	Move hand/buttons	6.38	3.80
	Scale user/linear mapping	10.44	6.95
	Scale user/buttons	7.43	3.53
	Move hand/linear mapping	5.95	2.75
	Move hand/buttons	6.18	2.01
	Scale user/linear mapping	4.18	1.32
	Scale user/buttons	4.20	1.09
D0S0F1	Go-Go	30.63	19.33
	Move hand/linear mapping	31.48	18.75
	Move hand/buttons	38.34	15.71
	Scale user/linear mapping	42.24	28.33
	Scale user/buttons	59.34	70.53
	Move hand/linear mapping	31.38	17.09
	Move hand/buttons	49.74	41.97
	Scale user/linear mapping	22.79	16.30
	Scale user/buttons	21.27	12.11
D0S1F0	Go-Go	8.19	4.78
	Move hand/linear mapping	5.38	2.02
	Move hand/buttons	5.19	1.10
	Scale user/linear mapping	8.50	3.23
	Scale user/buttons	10.90	8.13
	Move hand/linear mapping	11.58	7.82
	Move hand/buttons	5.97	2.53
	Scale user/linear mapping	5.55	2.11
	Scale user/buttons	4.68	1.01
D0S1F1	Go-Go	4.59	37.36
	Move hand/linear mapping	36.96	19.22
	Move hand/buttons	44.05	16.55
	Scale user/linear mapping	59.57	36.07
	Scale user/buttons	61.09	29.07
	Move hand/linear mapping	62.02	42.35

	Move hand/buttons	35.02	30.17
	Scale user/linear mapping	61.95	33.77
	Scale user/buttons	51.58	15.34
D1S0F0	Go-Go	8.92	4.49
	Move hand/linear mapping	9.41	3.45
	Move hand/buttons	6.68	1.89
	Scale user/linear mapping	14.97	13.97
	Scale user/buttons	12.61	1.04
	Move hand/linear mapping	6.91	3.37
	Move hand/buttons	9.29	2.93
	Scale user/linear mapping	6.08	2.71
	Scale user/buttons	8.12	2.85
D1S0F1	Go-Go	44.71	26.45
	Move hand/linear mapping	52.57	28.11
	Move hand/buttons	47.20	29.19
	Scale user/linear mapping	70.09	19.92
	Scale user/buttons	42.61	22.88
	Move hand/linear mapping	43.93	34.87
	Move hand/buttons	29.94	21.15
	Scale user/linear mapping	21.69	12.84
	Scale user/buttons	24.79	15.45
D1S1F0	Go-Go	17.89	13.60
	Move hand/linear mapping	11.16	3.94
	Move hand/buttons	12.33	8.83
	Scale user/linear mapping	15.63	5.04
	Scale user/buttons	15.52	6.52
	Move hand/linear mapping	14.34	7.33
	Move hand/buttons	6.66	1.22
	Scale user/linear mapping	11.81	6.73
	Scale user/buttons	9.52	3.26
D1S1F1	Go-Go	40.51	19.78
	Move hand/linear mapping	53.94	28.02
	Move hand/buttons	39.17	31.56
	Scale user/linear mapping	65.39	34.52
	Scale user/buttons	75.01	39.85
	Move hand/linear mapping	60.39	21.78
	Move hand/buttons	23.74	12.75
	Scale user/linear mapping	36.29	9.52
	Scale user/buttons	24.88	19.90
D2S0F0	Go-Go	13.92	4.92
	Move hand/linear mapping	29.61	23.05
	Move hand/buttons	16.50	5.27
	Scale user/linear mapping	37.75	13.95
	Scale user/buttons	28.79	8.73
	Move hand/linear mapping	13.70	6.36
	Move hand/buttons	9.06	2.98
	Scale user/linear mapping	10.81	1.73
	Scale user/buttons	10.89	6.70
D2S0F1	Go-Go	19.63	8.20

	Move hand/linear mapping	67.02	49.78
	Move hand/buttons	48.25	39.98
	Scale user/linear mapping	45.41	36.11
	Scale user/buttons	52.75	32.71
	Move hand/linear mapping	41.54	25.48
	Move hand/buttons	33.71	12.50
	Scale user/linear mapping	45.30	38.19
	Scale user/buttons	33.56	22.66
D2S1F0	Go-Go	14.43	8.44
	Move hand/linear mapping	22.28	11.31
	Move hand/buttons	21.14	14.34
	Scale user/linear mapping	29.63	10.42
	Scale user/buttons	26.18	12.73
	Move hand/linear mapping	27.91	13.18
	Move hand/buttons	11.28	4.64
	Scale user/linear mapping	26.37	24.15
	Scale user/buttons	17.74	12.21
D2S1F1	Go-Go	69.68	30.90
	Move hand/linear mapping	59.50	23.01
	Move hand/buttons	86.39	47.19
	Scale user/linear mapping	88.58	21.99
	Scale user/buttons	85.97	13.74
	Move hand/linear mapping	61.96	32.07
	Move hand/buttons	49.84	29.68
	Scale user/linear mapping	44.53	18.69
	Scale user/buttons	35.08	16.00

Notes: D0, D1, and D2 refer to the three levels of distance from the object to the target. S0 and S1 refer to the two sizes of the target. F0 and F1 refer to the 2 DOF and 6 DOF conditions, respectively.

Table D.3 Demographic and Comfort Rating Summary

Technique	1	2	3	4	5	6	7	8	9
# Left-Handed	0	1	0	1	0	0	0	0	0
# Females	1	2	0	5	1	1	2	4	1
Avg. Age	21.2	19.8	19.4	18.6	19.6	21.8	20.6	18.4	23.6
# Experienced VEs	3	1	4	1	2	4	1	0	0
Avg. SA score	12.4	8	12.6	5.4	9.4	11	10.8	12.2	8.2
Arm1	1	1.2	1	1	1	1	1	1.2	1
Hand1	1	1	1	1	1	1	1	1.2	1
Dizzy1	1.6	1.2	2.4	1	1	1	1.4	1.6	1.4
Nausea1	1	1	1	1	1	1	1	1	1
Arm2	1.8	1.2	1	1	1	1.4	1.4	1.6	2.2
Hand2	1.4	1.2	1	1	1.2	1	1.2	1.4	1.4
Dizzy2	1.4	1.2	1.6	1.4	1.2	1.2	1.4	1.6	1.2
Nausea2	1	1	1	1	1	1	1	1	1
Arm3	3.4	1.6	1.4	1	1.2	3	1.8	2.4	3.8
Hand3	1.8	1.4	1.4	1	1.8	1.2	1.6	2	1.6
Dizzy3	1.2	1.8	1.8	3	1.4	1.4	1.6	2	1.8
Nausea3	1	1	1.2	1.6	1.2	1	1	1.2	1
Arm4	5.8	1.8	1.6	1	1.2	4.4	3	3.2	4.8
Hand4	2.4	1.6	1.4	1	1.6	1.2	2	2.4	1.8
Dizzy4	2	2	1.8	4	1.6	1.4	1.4	2.8	1.8
Nausea4	1.8	1	1.6	2.8	1.2	1	1	1.8	1
Arm5	5	2	1.6	1	1.4	3.4	1.8	2.6	3.8
Hand5	2.2	1.4	1.6	1	1.8	1.2	1.4	1.8	1.4
Dizzy5	1.8	1.8	1.8	3.6	1.8	1.4	1.2	2	1.6
Nausea5	1.8	1.2	1.2	2.6	1.2	1	1	1.4	1
Arm6	5.2	2	1.4	1	1.2	4.8	3.4	3.4	4.6
Hand6	2.6	1.6	1.6	1	1.8	1.2	1.8	2	1.4
Dizzy6	1.6	1.8	1.8	3.6	2.2	1.4	1.2	2.2	2
Nausea6	1.4	1.2	1.2	2.6	1.4	1	1	1.6	1

REFERENCES

Allison, D., Wills, B., Bowman, D. Wineman, J., and Hodges, L. (1997). The Virtual Reality Gorilla Exhibit. *IEEE Computer Graphics & Applications*, *17*(6), 30-38.

Angus, I. and Sowizral, H. (1995). Embedding the 2D Interaction Metaphor in a Real 3D Virtual Environment. *Proceedings of SPIE, Stereoscopic Displays and Virtual Reality Systems*, *2409*, 282-293.

Arns, L., Cook, D., & Cruz-Neira, C. (1999). The Benefits of Statistical Visualization in an Immersive Environment. *Proceedings of IEEE Virtual Reality*, 88-95.

Baddeley, A. (1983). Working Memory. *Philosophical Transactions of the Royal Society London B*, *302*, 311-324.

Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G. (1997). The Rockin' Mouse: Integral 3D Manipulation on a Plane. *Proceedings of CHI*, 311-318.

Barfield, W., Zeltzer, D., Sheridan, T., and Slater, M. (1995). Presence and Performance Within Virtual Environments. In Barfield, W., and Furness, T. (Eds.)*Virtual Environments and Advanced Interface Design*. Oxford University Press.

Barfield, W., Hendrix, C., and Bystrom, K. (1997). Visualizing the Structure of Virtual Objects Using Head Tracked Stereoscopic Displays. *Proceedings of the Virtual Reality Annual International Symposium*, 114-120.

Bolter, J. Hodges, L., Meyer, T., and Nichols, A. (1995). Integrating Perceptual and Symbolic Information in VR. *IEEE Computer Graphics & Applications*, *15*(4), 8-11.

Bowman, D. (1996). Conceptual Design Space: Beyond Walk-through to Immersive Design. In Bertol, D. *Designing Digital Space*. John Wiley & Sons, New York, 225-236.

Bowman, D. and Hodges, L. (1995). User Interface Constraints for Immersive Virtual Environment Applications. Graphics, Visualization, and Usability Center Technical Report GIT-GVU-95-26.

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

Bowman, D., Hodges, L., and Bolter, J. (1998). The Virtual Venue: User-Computer Interaction in Information-Rich Virtual Environments. *Presence: Teleoperators and Virtual Environments*, *7*(5), 478-493.

Bowman, D., Koller, D., and Hodges, L. (1998). A Methodology for the Evaluation of Travel Techniques for Immersive Virtual Environments. *Virtual Reality: Research, Development, and Applications*, *3*(2), 120-131.

Bowman, D., Koller, D., and Hodges, L. (1997). Travel in Immersive Virtual Environments: An Evaluation of Viewpoint Motion Control Techniques. *Proceedings of the Virtual Reality Annual International Symposium*, 45-52.

Bowman, D., Wineman, J., Hodges, L., and Allison, D. (1999). The Educational Value of an Information-Rich Virtual Environment. *Presence: Teleoperators and Virtual Environments*, *8*(3), 317-331.

Brooks, F. et al. (1992). Final Technical Report: Walkthrough Project. Report to National Science Foundation.

Bruce, V. and Green, P. (1990). *Visual Perception: Physiology, Psychology, and Ecology*. Erlbaum.

Bryson, S. (1996). Virtual Reality in Scientific Visualization. *Communications of the ACM*, *39*(5), 62-71.

Bryson. S. and Levit, C. (1992). The Virtual Wind Tunnel. *IEEE Computer Graphics and Applications*, *12*(4), 25-34.

Bukowski, R. and Sequin, C. (1995). Object Associations: A Simple and Practical Approach to Virtual 3D Manipulation. *Proceedings of the ACM Symposium on Interactive 3D Graphics*, 131-138.

Butterworth, J., Davidson, A., Hench, S., and Olano, M. (1992). 3DM: A Three Dimensional Modeler Using a Head-Mounted Display. *Proceedings of the ACM Symposium on Interactive 3D Graphics*, 135-138.

Card, S., Mackinlay, J., and Robertson, G. (1990). The Design Space of Input Devices. *Proceedings of CHI*, 117-124.

Card, S., Moran, T., and Newell, A. (1980). The Keystroke-Level Model for User Performance Time with Interactive Systems. *Communications of the ACM*, *23*(7), 398-410.

Card, S., Moran, T., and Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, NJ, Erlbaum.

Card, S., Moran, T., and Newell, A. (1986). The Model Human Processor. In Boff, K., Kaufman, L., and Thomas, J. (Eds.) *Handbook of Perception and Human Performance*. Vol. II. Wiley, 45:1-45:35

Chance, S., Gaunet, F., Beall, A., and Loomis, J. (1998). Locomotion Mode Affects the Updating of Objects Encountered During Travel. *Presence: Teleoperators and Virtual Environments*, *7*(2), 168-178.

Chin, J., Diehl, V., and Norman, K. (1988). Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface. *Proceedings of CHI*, 213-218.

Chung, J. (1992). A Comparison of Head-Tracked and Non-head-tracked Steering Modes in the Targeting of Radiotherapy Treatment Beams. *Proceedings of the Symposium on Interactive 3D Graphics*, 193-196.

Coe, J. (1985). Design and Perception: Making the Zoo Experience Real. *Zoo Biology*, *4*, 197-208.

Cooper, L. and Shepard, R. (1978). Transformations on Representations of Objects in Space. In Carterette, E. and Friedman, M. (Eds.) *Handbook of Perception*. Vol. 8, Academic Press.

Cruz-Neira, C., Sandin, D., and DeFanti, T. (1993). Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE. *Proceedings of SIGGRAPH*, in *Computer Graphics*.

Darken, R. (1996). Wayfinding Behaviors and Strategies in Large Virtual Worlds. *Proceedings of CHI*.

Darken, R. and Cevik, H. (1999). Map Usage in Virtual Environments: Orientation Issues. *Proceedings of IEEE Virtual Reality*, 133-140.

Darken, R., and Sibert, J. (1993). A Toolset for Navigation in Virtual Environments. *Proceedings of the ACM Symposium on User Interface Software and Technology*, 157-165.

Darken, R. and Sibert, J. (1995). Navigating Large Virtual Spaces. *International Journal of Human-Computer Interaction*, October.

Davis, E. and Hodges, L. (1995). Human Stereopsis, Fusion, and Virtual Environments. In Barfield, W., and Furness, T. (Eds.) *Virtual Environments and Advanced Interface Design*. Oxford University Press.

Dede, C., Salzman, M., and Loftin, R. (1996). ScienceSpace: Virtual Realities for Learning Complex and Abstract Scientific Concepts. *Proceedings of the Virtual Reality Annual International Symposium*, 246-252.

Dinh, H., Walker, N., Hodges, L., Song, C., and Kobayashi, A. (1999). Evaluating the Importance of Multi-Sensory Input on Memory and the Sense of Presence in Virtual Environments. *Proceedings of IEEE Virtual Reality*, 222-228.

Durlach, N. (1991). Auditory Localization in Teleoperator and Virtual Environment Systems: Ideas, Issues, and Problems. *Perception*, *20*, 543-554.

Eberts, R. (1994). Experimental Designs and Analysis. In Eberts, R. *User Interface Design*. Prentice Hall, 82-125.

Fleishman, E. and Quaintance, M. (1984). Taxonomies of Human Performance: The Description of Human Tasks. Orlando, FL, Academic Press.

Foley, J. (1979). A Standard Computer Graphics Subroutine Package. *Computers and Structures*, *10*, 141-147.

Foley, J., van Dam, A., Feiner, S., and Hughes, J. (1990). *Interactive Computer Graphics: Principles and Practice*. Addison-Wesley, Reading, MA.

Gabbard, J., and Hix, D. (1998). Usability Engineering for Virtual Environments through a Taxonomy of Usability Characteristics. Submitted to *Presence: Teleoperators and Virtual Environments*.

Gentner, D. and Stevens, A. (Eds.) (1983). *Mental Models*. Erlbaum.

Goble, J., Hinckley, K., Pausch, R., Snell, J., and Kassell, N. (1995). Two-Handed Spatial Interface Tools for Neurosurgical Planning. *IEEE Computer*, July, 20-26.

Gomez, D., Burdea, G., and Langrana, N. (1995). Integration of the Rutgers Master II in a Virtual Reality Simulation. *Proceedings of the Virtual Reality Annual International Symposium*, 198-202.

Hendrix, C. and Barfield, W. (1996). Presence within Virtual Environments as a Function of Visual Display Parameters. *Presence: Teleoperators and Virtual Environments, 5*(2).

Herndon, K., van Dam, A., and Gleicher, M. (1994). The Challenges of 3D Interaction. *SIGCHI Bulletin*, *26*(4), October, 36-43.

Hettinger, L. and Riccio, G. (1992). Visually Induced Motion Sickness in Virtual Environments. *Presence: Teleoperators and Virtual Environments, 1*(3), 306-310.

Hinckley, K., Pausch, R., Goble, J., and Kassell, N. (1994). Design Hints for Spatial Input. *Proceedings of the ACM Symposium on User Interface Software and Technology*, 213-222.

Hinckley, K., Pausch, R., Profitt, D., Patten, J., and Kassell, N. (1997). Cooperative Bimanual Action. *Proceedings of CHI*, 27-34.

Hix, D., Swan, J., Gabbard, J., McGee, M., Durbin, J., and King, T. (1999). User-Centered Design and Evaluation of a Real-Time Battlefield Visualization Virtual Environment. *Proceedings of IEEE Virtual Reality*, 96-103.

Hix, D. and Hartson, H. (1993). *Developing User Interfaces: Ensuring Usability Through Product and Process*. John Wiley & Sons.

Hodges, L., Rothbaum, B., Kooper, R., Opdyke, D., Meyer, T., North, M., de Graff, J., and Williford, J. (1995). Virtual Environments for Treating the Fear of Heights. *IEEE Computer*, *28*(7), 27-34.

Holtzblatt, K. and Jones, S. (1993). Contextual Inquiry: A Participatory Technique for System Design. In Namioka, A. and Schuler, D. (Eds.) *Participatory Design: Principles and Practice*. Erlbaum.

Ingram, R. and Benford, S. (1995). Legibility Enhancement for Information Visualization. *Proceedings of Vizualization*, 209-216.

Iwata, H. and Fujii, T. (1996). Virtual Perambulator: A Novel Interface Device for Locomotion in Virtual Environment. *Proceedings of the IEEE Virtual Reality Annual International Symposium*. 60-65.

Jacoby, R. and Ellis, S. (1992). Using Virtual Menus in a Virtual Environment. *Proceedings of SPIE, Visual Data Interpretation*, *1668*, 39-48.

Kaur, K. (1999). Designing Virtual Environments for Usability. Doctoral Dissertation, University College London.

Kennedy, R., Lane, N., Berbaum, K., and Lilienthal, M. (1993). A Simulator Sickness Questionnaire (SSQ): A New Method for Quantifying Simulator Sickness. *International Journal of Aviation Psychology*, *3*(3), 203-220.

Kessler, G., Kooper, R., and Hodges, L. (1998). The Simple Virtual Environment Library, Version 2.0 User's Guide. Graphics, Visualization, and Usability Center Technical Report GIT-GVU-98-13.

Kheddar, A., Chellali, R., and Coiffet, P. (1995). Implementation of Head-Behavior Based Control for Navigation within Virtual Reality Applications. *Proceedings of IEEE International Conference on Systems, Man and Cybernetics: Intelligent Systems for the 21st Century*, 4644-4649.

Kijima, R. and Hirose, M. (1996). Representative Spherical Plane Method and Composition of Object Manipulation Methods. *Proceedings of the Virtual Reality Annual International Symposium*, 195-204.

Kitamura, Y., Yee, A., and Kishino, F. (1996). Virtual Object Manipulation Using Dynamically Selected Constraints with Real-Time Collision Detection. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, 173-181.

Kobsa, A. and Wahlster, W. (Eds.) (1989). *User Models in Dialog Systems*. Berlin, Springer.

Koller, D., Mine, M., and Hudson, S. (1996). Head-Tracked Orbital Viewing: An Interaction Technique for Immersive Virtual Environments. *Proceedings of the ACM Symposium on User Interface Software and Technology*, 81-82.

Lampton, D., Knerr, B., Goldberg, S., Bliss, J., Moshell, J., and Blau, B. (1994). The Virtual Environment Performance Assessment Battery (VEPAB): Development and Evaluation. *Presence: Teleoperators and Virtual Environments*, *3*(2), 145-157.

Lynch, K. (1960). *The Image of the City*. Cambridge, MA, MIT Press.

MacKenzie, I. (1995). Input Devices and Interaction Techniques for Advanced Computing. In Furness, T. and Barfield, W. (Eds.) *Virtual Environments and Advanced Interface Design* . Oxford University Press, 437-472.

Mackinlay, J., Card, S., and Robertson, G. (1990). Rapid Controlled Movement Through a Virtual 3D Workspace. *Proceedings of SIGGRAPH*, in *Computer Graphics*, *24*(4), 171-176.

Mapes, D. and Moshell, J. (1995). A Two-Handed Interface for Object Manipulation in Virtual Environments. *Presence: Teleoperators and Virtual Environments*, *4*(4), 403-416.

McGee, M. (1979). *Human Spatial Abilities*. Praeger.

Mercurio, P., Erickson, T., Diaper, D., Gilmore, D., Cockton, G., and Shackel, B. (1990). Interactive Scientific Visualization: An Assessment of a Virtual Reality System. *Proceedings of INTERACT*, 741-745.

Meyer, K. and Applewhite, H. (1992). A Survey of Position Trackers. *Presence: Teleoperators and Virtual Environments*, *1*(2), 173-200.

Microsoft Corporation. (1998). Microsoft Sidewinder Game Devices. Available at http://www.microsoft.com/sidewinder/default.htm

Miller, G. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information. *The Psychological Review*, *63*(2), 81-97.

Mine, M. (1995). Virtual Environment Interaction Techniques. UNC Chapel Hill Computer Science Technical Report TR95-018.

Mine, M. (1997). ISAAC: A Meta-CAD System for Virtual Environments. *Computer-Aided Design*, *29*(8), 547-553.

Mine, M., Brooks, F., and Sequin, C. (1997) Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction. *Proceedings of SIGGRAPH*, in *Computer Graphics*, 19-26.

Nemire, K. (1996). Evaluating Visual and Auditory Enhancements to a Virtual-Object Manipulation Task. *Proceedings of SPIE: Stereoscopic Displays and Virtual Reality Systems III*, 249-260.

Nielsen, J. and Molich, R. (1992). Heuristic Evaluation of User Interfaces. *Proceedings of CHI*, 249-256.

Nielsen, J. (1993). Noncommand User Interfaces. *Communications of the ACM*, *36*(4), 83-99.

Norman, D. (1990). *The Design of Everyday Things*. Doubleday, New York.

North, M., North, S., and Coble, J. (1996). Virtual Reality Therapy in the Treatment of Psychological Disorders. *Proceedings of Human Factors and Ergonomics Society Conference*.

Pausch, R., Burnette, T., Brockway, D., and Weiblen, M. (1995). Navigation and Locomotion in Virtual Worlds via Flight into Hand-Held Miniatures. *Proceedings of SIGGRAPH*, in *Computer Graphics*, 399-400.

Pausch, R., Shackelford, M., and Proffitt, D. (1993). A User Study Comparing Head-Mounted and Stationary Displays. *Proceedings of the IEEE Symposium on Research Frontiers in Virtual Reality*, 41-45.

Pierce, J., Forsberg, A., Conway, M., Hong, S., Zeleznik, R., and Mine, M. (1997). Image Plane Interaction Techniques in 3D Immersive Environments. *Proceedings of the ACM Symposium on Interactive 3D Graphics*, 39-44.

Pierce, J., Stearns, B., and Pausch, R. (1999). Voodoo Dolls: Seamless Interaction at Multiple Scales in Virtual Environments. *Proceedings of the ACM Symposium on Interactive 3D Graphics*, 141-146.

Plaisant, C., Carr, D., and Shneiderman, B. (1995). Image-Browser Taxonomy and Guidelines for Designers. *IEEE Software*, March.

Polson, P., Lewis, C., Rieman, J., and Wharton, C. (1992). Cognitive Walkthroughs: A Method for Theory-Based Evaluation of User Interfaces. *International Journal of Man-Machine Studies*, *36*, 741-773.

Poupyrev, I., Billinghurst, M., Weghorst, 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*, 79-80.

Poupyrev, I., Weghorst, S., Billinghurst, M., and Ichikawa, T. (1997). A Framework and Testbed for Studying Manipulation Techniques for Immersive VR. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, 21-28.

Price, B., Baecker, R., and Small, I. (1993). A Principled Taxonomy of Software Visualization. *Journal of Visual Languages and Computing*.

Richard, P., Birebent, G., Coiffet, P., Burdea, G., Gomez, D., and Langrana, N. (1995). Effect of Frame Rate and Force Feedback on Virtual Object Manipulation. *Presence: Teleoperators and Virtual Environments*, *5*(1), 95-108.

Robinett, W. and Holloway, R. (1992). Implementation of Flying, Scaling, and Grabbing in Virtual Worlds. *Proceedings of the ACM Symposium on Interactive 3D Graphics*, 197-208.

Rygol, M. et al. (1995). Tools and metaphors for user interaction in virtual environments. In Earnshaw, R., Vince, J., and Jones, H. (Eds.) *Virtual Reality Applications*. Academic Press Ltd, 149-161.

Schieser, E. (1986). Principles of Navigation. In Belove, C. (Ed.) *Handbook of Modern Electronics and Electrical Engineering.* Wiley.

Slater, M., Usoh, M., and Steed, A. (1994). Depth of Presence in Virtual Environments. *Presence: Teleoperators and Virtual Environments*, *3*(2), 130-144.

Slater, M., Usoh, M., and Steed, A. (1995). Taking Steps: The Influence of a Walking Technique on Presence in Virtual Reality. *ACM Transactions on Computer-Human Interaction*, *2*(3), 201-219.

Slater, M., Steed, A., and Usoh, M. (1995). The Virtual Treadmill: A Naturalistic Metaphor for Navigation in Immersive Virtual Environments. *Virtual Environments '95: Selected Papers of the Eurographics Workshops*, New York, SpringerWien, 135-148.

Song, D. and Norman, M. (1993). Cosmic Explorer: A Virtual Reality Environment for Exploring Cosmic Data. *Proceedings of the IEEE Symposium on Research Frontiers in Virtual Reality*, 75-78.

Spacetec IMC. (1998). The Spaceball 3003 RealLife 3D (TM) Controller. Available at http://www.spacetec.com/.

Stanney, K. (1995). Realizing the Full Potential of Virtual Reality: Human Factors Issues That Could Stand in the Way. *Proceedings of the IEEE Virtual Reality Annual International Symposium*, 28-34.

Stoakley, R., Conway, M., and Pausch, R. (1995). Virtual Reality on a WIM: Interactive Worlds in Miniature. *Proceedings of CHI*, 265-272.

Strommen, E. (1994). Children's use of mouse-based interfaces to control virtual travel. *Proceedings of CHI*, 405-410.

Sutherland, I. (1968). A head-mounted three dimensional display. *Proceedings of Fall Joint Computer Conference*, *33*, 757-764.

Tate, D., Sibert, L., and King, T. (1997). Virtual Environments for Shipboard Firefighting Training. *Proceedings of the IEEE Virtual Reality Annual International Symposium*, 61-68.

Taylor, R., Robinett, W., Chi, V., Brooks, F., Wright, W., Williams, R., and Snyder, E. (1993). The Nanomanipulator: A Virtual-Reality Interface for a Scanning Tunneling Microscope. *Proceedings of SIGGRAPH*, in *Computer Graphics*.

Waller, D., Hunt, E., & Knapp, D. (1998). The Transfer of Spatial Knowledge in Virtual Environment Training. *Presence: Teleoperators and Virtual Environments*, *7*(2), 129-143.

Ware, C. (1990). Using Hand Position for Virtual Object Placement. *Visual Computer*, *6*(5), 245-253.

Ware, C. and Slipp, L. (1991). Using Velocity Control to Navigate 3D Graphical Environments: A Comparison of Three Interfaces. *Proceedings of the Human Factors Society 35th Annual Meeting*, 300-304.

Ware, C. and Jessome, D. (1988). Using the Bat: a Six-Dimensional Mouse for Object Placement. *IEEE Computer Graphics and Applications*, *8*(6), 65-70.

Ware, C. and Osborne, S. (1990). Exploration and Virtual Camera Control in Virtual Three Dimensional Environments. *Proceedings of the ACM Symposium on Interactive 3D Graphics*, in *Computer Graphics*, *24*(2), 175-183.

Ware, C. and Balakrishnan, R. (1994). Reaching for Objects in VR Displays: Lag and Frame Rate. *ACM Transactions on Computer-Human Interaction*, *1*(4), 331-356.

Warren, R. and Wertheim, A. (Eds.) (1990). *Perception and Control of Self-Motion*. Hillsdale, NJ, Erlbaum.

Watson, B., Walker, N., and Hodges, L. (1995). A User Study Evaluating Level of Detail Degradation in the Periphery of Head-Mounted Displays. *Proceedings of Framework for Immersive Virtual Environments*, 203-212.

Watson, B., Spaulding, V., Walker, N., and Ribarsky, R. (1997). Evaluation of the Effects of Frame Time Variation on VR Task Performance. *Proceedings of the IEEE Virtual Reality Annual International Symposium*, 1997, 38-44.

Wickens, C. and Baker, P. (1995). Cognitive Issues in Virtual Reality. In Barfield, W. and Furness, T. (Eds.) *Virtual Environments and Advanced Interface Design*. Oxford University Press.

Williges, R. (1984). Evaluating Human-Computer Software Interfaces. *Proceedings of International Conference on Occupational Ergonomics*.

Wloka, M. and Greenfield, E. (1995). The Virtual Tricorder: A Unified Interface for Virtual Reality. *Proceedings of the ACM Symposium on User Interface Software and Technology*, 39-40.

Zhai, S. and Milgram, P. (1993). Human Performance Evaluation of Manipulation Schemes in Virtual Environments. *Proceedings of the IEEE Virtual Reality Annual International Symposium*, 155-161.

VITA

Douglas A. Bowman was born in Chattanooga, Tennessee on November 9, 1971. He received his Bachelor of Science degree in Mathematics and Computer Science from Emory University in Atlanta, Georgia in 1994. He also earned a Master of Science degree in Computer Science from the Georgia Institute of Technology in 1997. He has received numerous honors and awards, including the Woodruff Scholarship at Emory University, membership in Phi Beta Kappa, and a National Science Foundation graduate fellowship. His work has been presented at conferences including ACM SIGGRAPH, the IEEE Virtual Reality Conference, and the ACM Symposium on Interactive 3D Graphics, and in journals including *Presence: Teleoperators and Virtual Environments* and *IEEE Computer Graphics & Applications*. Doug will begin a position as Assistant Professor of Computer Science at the Virginia Polytechnic Institute and State University in the fall of 1999.