

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

# USING PROGRAMMING TOOLS IN VIRTUAL ENVIRONMENTS

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF SCIENCE

AT THE UNIVERSITY OF CAPE TOWN

IN FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

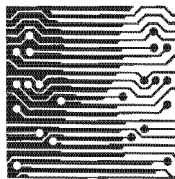
By

Shih-min Yang

May 2002

Supervised by

Gary Marsden



© Copyright 2002

by

Shih-min Yang

University of Cape Town

# Abstract

Presence is the sense of *being in* the computer-generated environment. It is regarded as the key to understanding the success of a virtual environment. In this research we focus on desktop virtual environment authoring tools. We believe that presence is also important in authoring tools because with greater presence, people can enjoy the process of building virtual environments and increase the task performance.

The type of desktop virtual environment authoring tool we are interested in is represented best by Alice. It allows novice users, who do not have knowledge of computing, to create virtual environments. We have identified some problems in Alice with respect to presence and human-computer interaction.

In this dissertation, we built four virtual environment prototypes with three different interaction methods. The three interactions were: the conventional interaction method; the “tool approach” and the “pin approach”. These four prototypes were used in two experiments.

In the first (presence) experiment, we investigated whether the sense of presence can be maintained with our novel interaction method. We hypothesised that the level of presence should be higher in the tool approach prototype than that in the conventional approach prototype. However, we found no difference between the mean of the presence scores in the two systems, although the presence generated in the tool approach prototype was slightly greater than in the conventional approach prototype.

Initially we thought that the users might behave differently depending on the interaction method in the virtual environment. In the second (usability) experiment, we found no differences in the user’s behaviour, however, and the users reacted in the virtual environments as if they were in reality.

# Acknowledgments

I would like to thank to my supervisor, Prof. Gary Marsden, for his guidance and help and as well as many helpful suggestions. I would also like to thank Dave Nunez for helping me to design the experiment and to analyse the results. I am grateful to Otto, Dave and everyone who helped to review drafts of my dissertation. I am also grateful to my friends and students in the CVC Lab who provide comments, suggestions and discussions on my work. Lastly, I wish to thank my parents for their love and supports.

# Contents

Abstract.....	iii
Acknowledgments.....	iv
Contents .....	v
List of Tables .....	viii
List of Figures.....	x
<b>Chapter 1 Introduction .....</b>	<b>1</b>
1.1 Virtual Reality Authoring Tools.....	4
1.2 VR Authoring Tools and Usability .....	4
1.3 VR Authoring Tools and the Sense of Presence .....	5
1.4 Aims .....	6
1.5 Methodology .....	7
1.6 Dissertation Overview.....	8
<b>Chapter 2 Background.....</b>	<b>10</b>
2.1 Presence.....	10
2.1.1 Definition of Presence.....	11
2.1.2 Factors of Presence.....	12
2.1.3 Measurement of Presence.....	15
2.2 Human-Computer Interaction in 2D and 3D Environments.....	17
2.2.1 Direct Manipulation .....	18
2.2.2 Examples of Direct Manipulation in Virtual Environments.....	19
2.2.3 Command Based User Interface.....	22
2.2.4 Modes in User Interfaces.....	23
2.2.5 Design Usable Interactions in Virtual Environments .....	25
2.3 Summary .....	27
<b>Chapter 3 Problem Specification and Proposed Solutions.....</b>	<b>29</b>
3.1 Alice, a 3D Interactive Authoring Tools .....	29
3.2 Problems in Alice .....	31

3.3	Proposed Solutions.....	35
3.3.1	Single-Window Approach.....	35
3.3.2	Tool Approach.....	35
3.3.3	“Mode per Object” Approach.....	37
3.4	Summary .....	38
<b>Chapter 4</b>	<b>Design and Prototypes of Virtual Environments.....</b>	<b>40</b>
4.1	Interaction Options.....	41
4.2	The Conventional Virtual Environment.....	43
4.2.1	Navigation and Changing the Viewpoint in the Virtual Environment .....	43
4.2.2	Interaction Method .....	44
4.3	The Tool Approach Virtual Environment .....	46
4.3.1	Navigation and Changing the Viewpoint in Virtual Environments .....	46
4.3.2	Interaction Method .....	47
4.3.3	Interaction Options.....	50
4.3.4	Navigation in the Virtual Environment .....	50
4.4	The Pin Approach Virtual Environment .....	50
4.4.1	Interaction Method .....	50
4.5	Summary .....	53
<b>Chapter 5</b>	<b>Presence Experiment.....</b>	<b>55</b>
5.1	Subjects .....	55
5.2	Questionnaires.....	56
5.3	Equipment and Prototype Systems.....	57
5.4	Experimental Tasks.....	57
5.5	Experimental Procedure .....	57
5.6	Analysis of Results.....	58
5.6.1	Variables and Hypotheses .....	58
5.6.2	Gathering Data .....	59
5.6.3	Statistical Analysis .....	61
5.7	Discussion of Results .....	63
5.8	Summary .....	64
<b>Chapter 6</b>	<b>Usability Experiment.....</b>	<b>66</b>
6.1	Aims .....	66
6.2	Participants .....	67
6.3	Experimental Tasks.....	68
6.4	Experimental Procedure .....	70

6.5	Equipment and Prototype Systems Used.....	70
6.6	Measurement of Usability .....	71
6.6.1	Questionnaire .....	71
6.6.2	Observation .....	72
6.7	Observation Results.....	72
6.8	Questionnaire Results.....	74
6.8.1	Statistical Analysis .....	74
6.8.2	Users' Comments .....	75
6.9	Discussions.....	76
6.10	Recommendations .....	78
6.11	Summary .....	79
<b>Chapter 7</b>	<b>Conclusion .....</b>	<b>81</b>
7.1	Delivered Virtual Environments .....	81
7.2	Experiments for this Research.....	82
7.2.1	Results Obtained from the Presence Experiment .....	82
7.2.2	Results Obtained from the Usability Experiment.....	83
7.3	Recommendations .....	85
7.4	Future Work .....	85
<b>Appendix A</b>	<b>Presence Questionnaire.....</b>	<b>87</b>
<b>Appendix B</b>	<b>Immersive Tendencies Questionnaire .....</b>	<b>90</b>
<b>Appendix C</b>	<b>User Manual of Control Group for Presence Experiment.....</b>	<b>94</b>
<b>Appendix D</b>	<b>User Manual of Experimental Group for Presence Experiment</b>	<b>99</b>
<b>Appendix E</b>	<b>Tasks for Presence Experiment .....</b>	<b>103</b>
<b>Appendix F</b>	<b>Experimental Tasks for the Usability Experiment.....</b>	<b>105</b>
<b>Appendix G</b>	<b>Computer System Usability Questionnaire.....</b>	<b>106</b>
	<b>Bibliography .....</b>	<b>108</b>

# List of Tables

<b>TABLE 1: OBJECTS AND THE FUNCTIONS ASSOCIATED WITH THEM. THE FIRST COLUMN IS THE OBJECT IN THE VIRTUAL ENVIRONMENT. THE SECOND COLUMN IS THE DEFAULT FUNCTION ASSOCIATED WITH IT. THIS DEFAULT FUNCTION CAN BE MANIPULATED DIRECTLY. THE LAST COLUMN IS THE TOOLS THAT CAN APPLIED TO THE OBJECT. FOR INSTANCE, ALL BASIC INTERACTION TOOLS CAN BE APPLIED TO THE DESK AND THE DRAWER OF THE DESK CAN BE OPEN AND CLOSED WITHOUT ANY TOOLS. THERE ARE NO DIRECT MANIPULATIONS THAT CAN BE DONE ON WALLS, FLOOR, AND CEILING – ONLY PAINT AND MAGIC WAND CAN BE APPLIED TO THEM.....</b>	<b>51</b>
<b>TABLE 2: INFORMATION OF SUBJECTS PARTICIPATING IN THE EXPERIMENT. THE CONTROL GROUP USED THE CONVENTIONAL VIRTUAL ENVIRONMENT AUTHORING TOOL AND THE EXPERIMENTAL GROUP USED THE NEW DESIGN OF VIRTUAL ENVIRONMENT AUTHORING TOOL. ....</b>	<b>56</b>
<b>TABLE 3: THE P SCORE AND IT SCORE OF PARTICIPANTS IN THE CONTROL GROUP. ALL DATA FOR THE CONTROL GROUP IS OBTAINED. ....</b>	<b>59</b>
<b>TABLE 4: THE P SCORE AND IT SCORE OF EACH PARTICIPANT IN THE EXPERIMENTAL GROUP. SUBJECT 1 MISSING THE IT SCORE AND SUBJECT 15 MISSING BOTH DATA. ....</b>	<b>60</b>
<b>TABLE 5: CORRELATIONS BETWEEN P SCORES AND IT SCORES. SIGNIFICANT CORRELATIONS (<math>P &lt; 0.05</math>) IS IN BOLD. ....</b>	<b>61</b>
<b>TABLE 6: CORRELATION MATRIX OF P AND IT SCORES. (I) THE CORRELATION BETWEEN P SCORE AND IT SCORE FOR THE CONTROL GROUP. (II) CORRELATION BETWEEN P SCORE AND IT SCORE FOR THE EXPERIMENTAL GROUP. THE SIGNIFICANT LEVEL IS <math>P &lt; 0.05</math>, IN BOLD. ....</b>	<b>62</b>
<b>TABLE 7: RESULT OF T-TEST ON IT SCORE OF TWO GROUPS. IT SHOWS NO SIGNIFICANT DIFFERENCE (<math>P &lt; 0.05</math>) BETWEEN THEM. ....</b>	<b>62</b>
<b>TABLE 8: RESULT OF T-TEST ON P SCORES OF TWO GROUPS. IT IS NOT SIGNIFICANT AT <math>P &lt; 0.05</math>. ....</b>	<b>63</b>
<b>TABLE 9: INFORMATION ABOUT PARTICIPANTS IN THIS EXPERIMENT. THE COURSES INFORMATION OF PARTICIPANT 3 OF PIN APPROACH PROTOTYPE GROUP IS MISSING</b>	

(ALTHOUGH WE CAN ASSUME THEY HAD NOT STUDIED COMPUTER GRAPHICS OR 3D MODELLING AS THEY WOULD NOT HAVE BEEN ALLOWED TO PARTICIPATE IN THE EXPERIMENT).....68

TABLE 10: **THE DATA COLLECTED FROM THE CSUQ.** THE SCORE OF THE QUESTIONNAIRE FOR EACH PARTICIPANT AND THE MEAN AND STANDARD DEVIATION FOR EACH GROUP.75

TABLE 11: **THE RESULT OF T-TEST.** THE P VALUE IS 0.0057, WHICH IS SIGNIFICANTLY DIFFERENT AT  $P < 0.05$ . .....75

University of Cape Town

# List of Figures

Figure 1: The screen shots of virtual environments. ....	2
Figure 2: Head Mounted Display (HMD). ....	3
Figure 3: Alice authoring tool with multiple windows. ....	5
Figure 4: The world in miniature (WIM). ....	20
Figure 5: Voodoo dolls technique. ....	21
Figure 6: Floating menu systems. ....	22
Figure 7: The mode per system approach user interface. ....	24
Figure 8: An example of mode per object application. ....	25
Figure 9: Examples of affordances. ....	26
Figure 10: Alice authoring tool. ....	31
Figure 11: An example of mention and use. ....	33
Figure 12: Interaction in real life. ....	36
Figure 13: An example of Tool approach application. ....	37
Figure 14: Marsden's user interface. ....	38
Figure 15: Selection Techniques as Discussed in Mine [46]. ....	41
Figure 16: The navigation in the conventional virtual environment. ....	44
Figure 17: The screen shot of the conventional virtual environment. ....	45
Figure 18: The screen shot of the tool approach VE (a). ....	48
Figure 19: A screen shot of the tool approach VE (b). ....	49
Figure 20: The pin approach virtual environment. ....	52
Figure 21: The texture of the drawing pin, which is attached to objects. ....	53
Figure 22: The top view of the virtual environment. ....	68
Figure 23: A set of arrangements of the room. ....	69

# Chapter 1

## Introduction

Virtual environments have been researched for several years. “Virtual environment” or “virtual reality” is a common term for describing the use of three-dimensional graphics displays to explore a computer-generated world. In Brooks [11], the author defines virtual reality as:

*any in which the user is effectively immersed in a responsive virtual world.*

The key features of virtual environments, according to Brooks [11], are:

- users are convinced that they are “there”
- users are allowed to navigate and interact in the virtual environment.

Virtual environments deliver a 3D computer-generated world in which the users are easily convinced that they are in the generated world instead of the physical environment. This feature is commonly believed to be the power of virtual reality. Furthermore, with virtual environments, participants can interact with life-like objects, or models, in a situation which is not usually possible to provide in real life [92].

Figure 1 shows some screen shots of this type of virtual environment.

Virtual environments are not only used in academic areas, but also in entertainment, and medical uses. The major market activity of virtual environments is in entertainment [99]. First-person shooting games and flight simulations are examples of successful entertainment applications. Besides entertainment, virtual environments are also successful in education, and in training. The virtual environment offers greater flexibility for training and provides some help to facilitate training, such as adjusting the level of difficulties [92]. In addition, the virtual environment offers enjoyment and entertainment, which enhances the motivation for learning [92]. Furthermore, virtual environments are able to simulate training in dangerous

environments without the physical dangers. An example of this is radioactive emergencies [92].



Figure 1: **The screen shots of virtual environments.** (i) This is the screen shot of first-person shooting game – Quake 2. (ii) This is the screen shot of a monastery using Genesis 3D, built by D. Nunez, CVC Lab, UCT.

Virtual environments, hence, are believed to have a great potential in many areas.

In terms of technology, there are two basic classifications of virtual environments:

- immersive virtual environment
- non-immersive virtual environment

Immersive virtual environments require specialised equipment. Normally this equipment includes a *head-mounted display* (HMD) and one or more *trackers*. The HMD gives the users 3D view of the environments. The trackers are attached on the HMD or the user's body. They give the system the orientation and position of the user. In this way the participants are isolated from the physical environment and included inside the virtual world, and thus a sense of *being in* the computer-generated environment is experienced [92]. The main disadvantage of immersive virtual environment technology is its cost.



Figure 2: **Head Mounted Display (HMD)**. It is a V6 Head Mounted Display manufactured by Visual Research.

Non-immersive virtual environments are a common alternative. The most popular type of system in this category is the desktop virtual environment. It uses an ordinary desktop computer system interacting with standard monitor, a mouse and a keyboard, or perhaps a special-purpose input device, such as a 3D mouse [92, 99]. Brooks [11] and many other researchers, [34], have quoted from “The Ultimate Display” by Ivan Sutherland [88].

*Don't think of that thing as a screen, think of it as a window, a window through which one looks into a virtual world.*

Several researchers [11] criticize desktop virtual environments as:

1. they do not block out the physical environments
2. virtual objects are not presented as life-size
3. desktop virtual environments do not provide the illusion of immersion

However, among all virtual environment systems, desktop virtual environments give the best image in terms of resolution, contrast, clarity and colour gamut [36].

It should be noted that this is not intended to be an exhaustive classification and there exist many other types of virtual environment display technologies, such as: projected virtual environment, Cave Automatic Virtual Environment (CAVE) and augmented virtual environments. Full classifications may be found in [34] and [99].

Our research, however, focuses on desktop virtual environments, not only because it is inexpensive, but also because there are some successful applications of desktop virtual environments, such as 3D games, or flight simulations. Despite the disadvantages of desktop virtual reality, it was reported as the most widely used virtual reality (VR) system in United

Kindom [85]. We therefore believe that desktop VR serves a worthwhile purpose and the relatively low cost of the hardware will allow the results of our research to have a much wider impact.

## 1.1 Virtual Reality Authoring Tools

The increasing popularity of 3D games means that many people have experienced 3D environments. People are attracted by the virtual environments and the technologies behind them. As a result, many people are willing to, and are interested in, creating their own virtual environments, even if they have no background in computer programming. Consequently, several researchers and commercial companies are working on creating easy to use virtual environment authoring tools.

To create a virtual environment requires software tools. There are several types of authoring software packages available for creating 3D virtual environments and modelling software to create 3D objects to populate the environment. Examples of modelling software packages are 3D Studio [1], and AutoCAD [5]. Furthermore, the users can not only create 3D objects but also generate interaction events with the virtual environment authoring software packages. This means that one can define the behaviour of objects in the authoring environments. For instance, the door will automatically open when the user walks close to it. Examples of authoring software include Genesis3D [24] and DIVE [16].

Most virtual reality (VR) authoring tools require advanced programming skills, such as C or C++, and linear algebra [57]. However, people who have the programming skills to create virtual worlds are not necessarily those who want to use the medium to create virtual environments. Fortunately, there are more and more VR authoring tools that are designed for novice users, and do not require the knowledge of programming. The Alice authoring tool [4] and EON Studio [19] are examples of this genre.

## 1.2 VR Authoring Tools and Usability

With the above-mentioned VR authoring tools, e.g. Alice, or EON Studio, the users are able to create their own virtual environments. However, the tools can be poorly designed. For

instance, in Alice<sup>1</sup>, there are three tabbed panes in the main window – *Open Scene*, *Animations*, and *Events* panes (see Figure 3). The “Open scene” pane is for viewing the setting of the virtual environment, the “animations” pane is used to start the animation, and the “events” pane allows the users to add events on objects. The users, thus, need to change between windows (or panes) for editing the virtual environment. The action of jumping between windows might confuse the users [30].

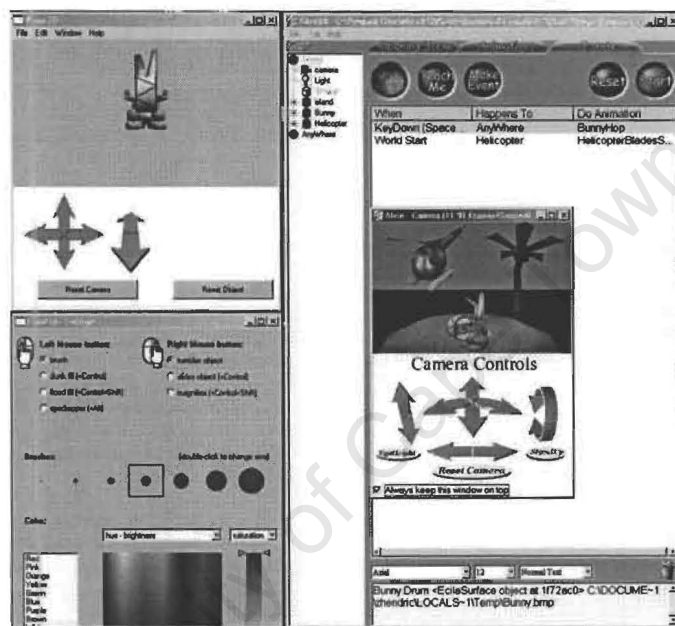


Figure 3: Alice authoring tool with multiple windows. The left two windows are for painting the object, a rabbit. And the two right windows are for simulation and setting the properties of objects and the scene (the larger one) and the camera view (the smaller one in the foreground). Once the left-hand-side windows are closed, the colour change will take effect in the camera view window.

### 1.3 VR Authoring Tools and the Sense of Presence

The use of multiple windows means that the experience of presence in the virtual environment is interrupted and the feeling of presence in virtual environments cannot be maintained. This

<sup>1</sup> Alice is the most developed of these systems and hence will be used as the main example of a tool for novice users

sense of “presence” is correlated with enjoyment [92], and it is also believed to be correlated with task performance within a virtual environment [42]. Though one might argue that it is not necessary to maintain this feeling while creating the virtual worlds, we believe that one can enjoy the process of building virtual environments and probably increase the task performance as the sense of presence increases.

## 1.4 Aims

In this research, therefore, we would like to investigate the development of desktop virtual environment authoring tools. We choose desktop virtual environments not only because it is cheap, but also because it is the most widely used virtual environment system outside the research laboratory. Additionally, most VR authoring tools use desktop virtual environment systems.

Our goal is to build a system which combines both the positive features of virtual environment applications, and VR authoring tools. The users can experience the virtual world without breaking the feeling of presence and they can create, modify, and interact with the virtual environments.

We also wish to maintain the sense of presence while experiencing virtual environments. In order to achieve this, we will investigate this notion of “presence”. We will examine the definition, the factors, and the measurement of presence.

Furthermore, we will examine current interaction techniques for virtual environments. We will examine the interaction methods used in different disciplines, including real world, and computer interface design techniques. We expect these techniques and methods to be useful in desktop virtual environments. As a result of the useful techniques and methods, the users of any experience level can easily create their own virtual environments while the sense of “presence” is maintained and the usability of the system is preserved.

Finally, we would like to design a user interface which is easy to use and easy to learn. For this purpose, we will investigate the different approaches for improving the usability of user interfaces, e.g. direct manipulation, and the correct use of modes.

At the end of this research, we aim to build a desktop virtual environment system, which allows the user to interact with, and to manipulate, a virtual environment. This system will be easy to learn and easy to use. Furthermore, the sense of presence should be maintained while experiencing the virtual environments.

We will consider our research successful when we have built the virtual environment, which, according to the principles of usability and theory of presence, has fewer usability problems and reduces the negative factors of presence. Furthermore, we will conduct experiments with users to evaluate our design.

## 1.5 Methodology

We will develop four virtual environment prototypes. One of them will be a mimic of the traditional VR authoring tool, which uses multiple windows for viewing the scene and editing virtual objects. The other three will explore different interaction metaphors aiming to achieve our goal mentioned in Section 1.4. All four prototypes will be evaluated in the experiments. We will conduct two experiments as follows:

### The Presence Experiment

The aim of this experiment is to investigate whether our new design of virtual environment authoring tool can generate higher presence than the conventional virtual environment authoring tool. The experiment will be presented in Chapter 5.

We will measure presence with the **Presence Questionnaire**, which was developed by Witmer and Singer [97]. We will also use the **Immersive Tendencies Questionnaire**, which was proposed by Witmer and Singer [97] too, to measure the immersive differences of individuals. The levels of presence in two virtual environments will be compared; and the correlation between presence level and the immersion tendencies of individual will be measured.

### The Usability Experiment

We also conduct a usability experiment. The aim of this experiment is to investigate the difference between two interaction methods which introduce editing capabilities to the authoring tool in different ways: one has a global editing mode and the other allows only individual items to be edited. This experiment will be described in Chapter 6.

In this experiment, we will observe the ways that the users perform tasks in the virtual environments. The experiment sessions will be video recorded for easy observation after the experiment. We will also use a usability questionnaire, the **Computer System Usability Questionnaire**, to measure the usability of the virtual environments.

## **1.6 Dissertation Overview**

### **Chapter 2 – Background**

In this chapter, we describe the background work relevant to our research. We first describe the definitions of presence and some theories of presence. We present the factors which are believed to affect the sense of presence in virtual environments. We also describe the different measurements of presence and the advantages and disadvantages of each. Thereafter, we describe some user interface design principles used in the traditional (2D) user interface. These design principles are thought to be useful and applicable in virtual environments.

### **Chapter 3 – Problem Specification and Proposed Solutions**

In this chapter, we identify the problems of VE authoring tool design, and propose some solutions. We firstly describe the features of Alice in detail, and then discuss the problems that we found in Alice which are related to presence and usability. We also discuss the reasons that we consider them as problems. We then present some solutions that were also proposed for direct manipulation user interface programming languages.

### **Chapter 4 – Design and Prototypes of Virtual Environments**

In this chapter, we present four prototype virtual environments. These virtual environments combine the positive features of virtual environments and that of VR authoring tools – the goal of this research. All four virtual environments are used in the experiments. All these virtual environments are built using Java3D; and they are desktop systems with standard input devices, mouse and keyboard, and standard output devices – monitor.

### **Chapter 5 – The Presence Experiment**

In this chapter, we present the presence experiment. We begin with information pertaining to the participants involved in the experiment and then the description of equipment, procedure and tasks of the experiment. Then the results of the experiment will be presented. We will then discuss the statistical results. This chapter ends with a summary and discussion of the experiment.

### **Chapter 6 – The Usability Experiment**

We present the usability experiment in this chapter. We describe the details of the setting of the experiment, and the questionnaire used for measuring usability. The statistical analysis is

then performed on the measurement of usability. The findings from the observation and statistical analysis are discussed. From the result of the experiment, we propose recommendations on the possible applications for the different interaction approaches. This chapter ends with the major findings and a summary of these recommendations.

### **Chapter 7 – Conclusion**

The dissertation ends with a conclusion on the major findings and interesting outcomes. Lastly we give the suggestions for future work and design guidelines for VE authors of the future.

University of Cape Town

## Chapter 2

# Background

In this chapter, we discuss the concept of presence and the human-computer interaction principles used in 2D and 3D environments, which are relevant to this research.

We start with the definition of presence, the factors and the measurements of presence in Section 2.1. We also discuss some human-computer interaction techniques used in both 2D and 3D environments in Section 2.2.

### 2.1 Presence

*Presence* is the sense of being in the computer-generated environment, and is regarded as the key to understanding the success of a virtual environment. It is referred to as *being there* [26]. It is believed that the higher the presence experienced in a virtual environment, the more efficient the user's task performance will be; the knowledge learned in the virtual environment can be transferred to the real world [42]. The best example is flight simulation – the pilots learn more effectively in the flight simulator than from the textbooks. It is therefore essential that a VE gives a high level of presence.

Presence is thought to be the central issue for virtual environments [75, 86]. Most researchers, such as Lombard and Ditton [42], agree that “An enhanced sense of presence is central to the use, and therefore the usefulness and profitability, of the new technologies mentioned above ...”. The new technologies include virtual reality, simulation rides, home theatre, 3D IMAX, etc. Furthermore, most researchers believe that presence could enhance task performance since in real life, involvement and attention play a great role in performance [71], even though there is no solid evidence proving this as yet.

### 2.1.1 Definition of Presence

Although the idea of virtual reality and presence are not novel, there are still no universal definitions of presence. One possible reason is that there are diverse groups of people studying presence. Thus presence is defined by the following groups accordingly.

**Virtual presence** was first proposed by Sheridan [70], which is derived from “telepresence”, as the “sense of being physically present with visual, auditory, or force displays generated by a computer”.

Several researchers, [76, 97], agree with Sheridan and define presence as a state of consciousness, the subjective experience of being in the virtual environment. While experiencing the virtual environment, users become more engaged with the virtual environment than the physical surroundings. Witmer and Singer [97] suggest that presence is a matter of focus – while experiencing a virtual environment, one is required to focus on the meaningful set of stimuli in order to have the sense of presence.

Another definition of presence is an *illusion of non-mediation* – a person fails to perceive the existence of a medium and responds as if the medium were not there [32]. Several advanced technologies, such as virtual environments and simulation rides, are designed to provide an illusion that the media is not there. Hence the users feel present in the mediated world instead of the physical world.

Since presence is a perceptual illusion, or a psychological sense of “being there”, it is a property of a person [42]. Sheridan, [71], and many other researchers, IJsselsteijn *et al.* [32] and Welch *et al.* [93], agree that presence varies across individuals and across time. Therefore, presence is not just a property of the medium, but is dependent on the characteristics of the medium’s observer.

Lombard *et al.* [42], summarizes six different conceptualisations of presence. The six conceptualisations are:

1. presence as social richness
2. presence as realism
3. presence as transportation
4. presence as immersion
5. presence as social actor within medium
6. presence as medium as social actor

The definition associated with this research are primarily (3) and (4) – presence as transportation and presence as immersion.

**Presence as transportation** can be identified as:

1. “You are there”, in which the users are transported to some other place
2. “It is here”, in which other people or objects are brought to one’s environment
3. “We are together” is the sense that one is in an environment with some other people [42]

The first definition, “You are there”, is the oldest version of presence. It is the feeling that one is actually “there” at the remote site, or in the computer-generated worlds [43]. “It is here” occurs when one fails to distinguish between image and referent. One responds directly as if those objects or beings are in one’s environment [43]. The last definition of presence as transportation is found in literature concerning video conferencing and virtual reality. It is the sense that people feel that they are together in the shared space (co-presence) [43].

**Presence as immersion** is when one is isolated from the physical environment [42]. This includes perceptual and psychological immersion. Perceptual immersion refers to the degree to which a virtual environment submerges the perceptual system of the user. Psychological immersion includes the psychological component: one feels involved, absorbed, engaged, and engrossed [43].

In this research, we are only interested in *being there* and the feeling of involvement, since these two definitions of presence are the most common in presence literature relating to VR authoring tools. The sense of being with some other beings, however, is not relevant as we are only interested in VR authoring tools for a single user – collaborative authoring falls outside the scope of our research.

### 2.1.2 Factors of Presence

In order to measure presence, we need to know the factors contributing to presence. These factors, however, are highly dependent on the definition and theory of presence; thus various sets of factors of presence are proposed. These proposed sets of factors are not exactly the same, but overlap with each other.

In general, there are three basic factors contributing to presence: a) vividness, b) interaction, and c) user characteristics [86]. The first two factors are technological aspects, while the last factor concerns the perceiver.

### **Vividness**

*Vividness* refers to the richness of sensory information in virtual environments. Vividness is defined as “the representational richness of a mediated environment as defined by its formal features, that is, the way in which an environment presents information to the senses” [86]. It is agreed that the richer the sensory information is used in the virtual environment, the greater the presence experienced. For instance, one study, [73], shows that greater presence is experienced with audio-visual task-based interaction than with an audio-only system [42]. Another good example is the entertainment systems used in Disneyland – several sensory stimuli are used to make people feel that they are present in the simulated environment.

An important issue of vividness is the consistency of several sensory outputs [27, 42]. For instance, one should hear an explosion, see a flash, and smell the smoke simultaneously if all three sensory stimuli are provided. There should not be any conflict [86]. Failure to meet this criteria emphasizes that the world is artificial and thus reduces presence.

Vividness, however, falls outside the scope of our study as it relates to the quality of underlying media and not how one interacts with that media. The level of vividness will be kept constant across all our prototype environments to ensure that it does not affect subjects’ perception of presence.

### **Interaction**

*Interaction* refers to the degree to which the observer can interact with, and control, parameters of the virtual environment [18]. Some researchers, such as Slater *et al.* [74], have argued that the sense of presence is the ability to act there, rather than just see the images. In addition, most researchers have assumed, or explicitly suggested, that the primary cause of presence is the ability to interact in the virtual environment [42, 71, 74]. In Witmer and Singer, [97], the authors argue that presence should increase as “one’s ability to modify physical objects in that environment increases”. Thus interaction is an important factor in facilitating the feeling of presence.

Three sub-factors of interaction are *range*, *mapping* and *response time*. *Range* refers to the number of attributes that can be manipulated. Generally the more control a person has over

the virtual environment, the greater the sense of presence experienced [86]. However, when the control parameters grow larger, it becomes more difficult for the participants to remember all controls. They might rely on a user manual, and thus the sense of presence is diminished.

*Mapping* refers to the relationship between two things [50]. In virtual environments, it refers to “the way in which human actions are connected to actions within a mediated environment” [86]. For example, the action of throwing a ball results in a virtual ball being thrown. It has been suggested that familiar manipulation strategies could contribute more to presence than indirect manipulation methods. Due to the above reasons, it is suggested by several researchers that the best interaction strategies should match the natural actions as closely as possible [86]. The most well-known and successful mapping in virtual environments is changing the viewpoint with a head-mounted display (HMD) and trackers. The view is changed automatically as the user’s head moves. However, in some case when there is no natural mapping appropriate, a standardized mapping strategy or conventional mapping is better than no strategy or arbitrary mapping [86]. Hence, as is widely believed, effective virtual environment metaphors could enhance human performance, especially for novice users, in virtual environments [84].

The last sub-factor of interaction is *response time* – the delay between the user input and the system output. The ideal response time should be instantaneous. However, this is not possible to achieve real time responses because of hardware and software limitations. It is, therefore, suggested that the lag time should not be noticeable [86].

### User Characteristics

Presence is defined as a subjective experience of the virtual environment. It is, therefore, dependent on individual differences [42, 71, 86]. Two important variables of users’ characteristics are *willingness to suspend disbelief*, and *the knowledge of and prior experience with the medium*. *Willingness to suspend disbelief* is to engage in the activity and get involved in the virtual environment, instead of reminding oneself that one is looking at a mediated world. The willingness to suspend disbelief varies across individuals and within the same individual across time [42]. *The knowledge of and prior experience with the medium* could affect the experience of presence. It is easier for users who are not familiar with how environments are created to experience presence, than those who know about the medium [42]. For instance, the designers of the virtual environment might find it difficult to get involved and feel presence since they might focus on the technological part of the virtual environment and this reminds them that the experience is mediated.

### 2.1.3 Measurement of Presence

The definition and the factors affecting presence are necessary for measuring presence. However, there are no standard definition and dimensions of presence. This results in several approaches for measuring presence. These measurements depend on the definitions and theories of presence which the researchers use.

#### Subjective Measurement of Presence

The most frequently used approach of measurement is using subjective questionnaires. The participants are required to answer a set of questionnaires regarding their experience in the virtual environment after the experiment. Sheridan [71] suggested that presence is a subjective sense of being there, thus “subjective report is the essential basic measurement” [71]. However, there is no standard questionnaire due to the variety of theories of presence.

Witmer and Singer, [97], have designed a questionnaire named the Presence Questionnaire (PQ). This questionnaire consists of thirty-two questions. These questions are based on the four factors of presence proposed by Witmer and Singer – control factors, sensory factors, distraction factors, and realism factors [97]. The questions, such as “How natural did your interactions with the environment seem?”, are quantified on a Likert scale. It is seven-point scale with 1 meaning strongly disagree, and 7 meaning strongly agree. The total presence score is calculated by adding up all questions (with some questions using reversed scoring). Besides these four technological factors of presence, Witmer and Singer have suggested that the individual differences might influence the level of presence experienced [97]. Consequently, another questionnaire was developed: the Immersive Tendencies Questionnaire (ITQ). This questionnaire is used to measure how easily one can be involved and immersed in virtual environments. This questionnaire asks about the users’ involvement and focus, for example, “Do you easily become deeply involved in movies or TV drama?” [97]. These questions are scored from one to seven and the total ITQ score is calculated by adding up the scores of all questions. According to Witmer and Singer, [97], if one gets a higher ITQ score, it means that one is easily involved in virtual environments. Therefore, one should report more presence in the PQ.

An alternative questionnaire was developed over a number of studies by Slater and colleagues [91]. The questions are based on one of three themes: the sense of being in the virtual environment, the extent to which the virtual environment becomes the dominant reality, and the extent to which the virtual environment is remembered as a place [91]. Each question is scaled from one to seven, where the highest score indicates greater presence. The presence

score is the number of questions that are scored six or seven. A sample question is: "Please rate your sense of 'being in' the virtual environment, on the following scale from 1 to 7, where 7 represents your normal experience of being in a place" [91].

Using questionnaires to measure presence is claimed to be reliable and valid [43] and, despite their faults, are the currently the most reliable means of assessment. However, questionnaires need to be used with great care since they produce unstable and inconsistent results across individuals and across time, as well as the study setting. Furthermore, they are subject to an inaccurate recall of the experience [33, 41]. In the study conducted by Freeman *et al* [22], they found that participants' understanding of presence affected the results of the presence measurement. The participants build a model of what the experimenters mean by *presence* based on the tasks and the experimental cues [22]. Additionally, in this study, they found that prior experience of virtual environments affects presence ratings. In spite of the problems of questionnaire measurement, most researchers use subjective questionnaire measurement in their studies because this way of measuring presence appears to be valid (they are logically related to our understanding of presence) and also because it is inexpensive and easy to use [41]. Additionally, it does not break the presence experience in the virtual environment.

Other approaches to subjective measurement are "continuous presence assessment" based on the "Turing test". Continuous presence assessment, proposed by IJsselsteijn *et al.* [32], overcomes some limitations of post-test questionnaire. It does not depend on recall and provides continuous presence scoring during the experiment [32]. The participants are required to adjust the presence level during the experiment and this is recorded. However, it requires the users to split their attention between experiencing a virtual environment, and adjusting a temporal presence score with one hand. Thus, this approach is only suitable in non-interactive virtual environments.

Schloerb, [66], proposes a quantitative measurement: the probability of correctly identifying real and virtual environments (real vs. artificial). This approach takes the form of a *Turning test*. The author claims that human beings can always distinguish between real and virtual environments. Thus some limitations are imposed on the real scene, for example, degradation of the image, or limiting the field of view in both environments [66, 72]. A problem with this approach is that it might end up testing the users' ability to distinguish real and virtual worlds, instead of measuring presence. Additionally, in the case of imaginary worlds and some situations which are not possible in real life, this method is unsuitable [33].

### **Objective Measurement of Presence**

The subjective measurement of presence is unstable since it requires the users' complete understanding of presence, and is inconsistent across individuals and time. An objective measurement, on the other hand, records the participants' physiological or behavioural responses and does not require conscious introspection. This approach is expensive and difficult to administer. Also, there is no clear evidence showing the relationship between the physiological or behavioural responses measured and the degree of presence that the participants experience [33, 41].

### **Presence Measurement Used in this Research**

We will use questionnaires to measure presence despite their disadvantages – unstable and inconsistent across individuals and time, and inaccurate recall of the experiment. Ease of administration, general acceptance within the field and cost constraints are the overriding factors why we choose questionnaires as our measurement method. The sets of questionnaires that we will use are the Presence Questionnaire and the Immersive Tendencies Questionnaire. These two questionnaires have been found to be reliable and valid as claimed by Witmer and Singer [97].

## **2.2 Human-Computer Interaction in 2D and 3D Environments**

Virtual environments are specialized human-computer interfaces. The users are involved and immersed in the environments. It seems that there is no boundary between the computer system and the users. Virtual environments are therefore quite different from the traditional user interfaces and they offer new possibilities and challenges to human-computer interface design [15]. Unfortunately there is little research on how to design usable virtual environments. Most virtual environment applications suffer from usability problems, such as the user's frustration in trying to manipulate the interface while trying to experience virtual environments. One possible reason for lack of usability is that designers are not concerned about usability initially as there are other major problems to solve [15].

In this section, we present some useful 2D design guidelines and strategies which might be possible to apply in virtual environments. There is some research on the usability guidelines for virtual environments, which will be discussed here as well.

### 2.2.1 Direct Manipulation

Direct manipulation is an interaction style for graphic user interfaces. Direct manipulation allows the user to interact with objects directly, as suggested by the name. The best known application of direct manipulation is the Windows system – one can copy, move, and delete files using the mouse to directly apply actions to the icons of files [25]. Studies have found that direct manipulation interface users have higher productivity and learn more in less time than users using the command line [25].

The features of direct manipulation, according to Heeter, [25], are:

1. the objects and actions which can be applied to those objects are visible
2. the interface is transparent
3. the user interacts with objects instead of intermediaries

Visibility helps the users to remember the actions that can be performed on objects. It shows the users what can be done, where they are and where they can go.

One method of making the interface transparent is to use the strategies, or methods that people already know and use, such as gesture recognition or handwriting recognition [25]. These methods are natural and familiar. They are powerful and easy to learn; however, they are difficult to implement especially in virtual environments. For instance, voice recognition allows the users to interact with, and directly manipulate objects by speaking normally. However, the correct recognition of commands is difficult to achieve since the voices of people are different. Therefore it requires that system be trained to recognise the users voice. Additionally, the users are required to memorise the commands [9].

An alternative approach is using a familiar metaphor. A metaphor is defined as “an invisible web of terms and associations that underlie the way we speak and think about a concept.” [25]. The users can construct a conceptual model of the system with a well-designed metaphor. This conceptual model helps the users to understand and predict the behaviour of the system [44]. The users, therefore, think of the system as a model that they build, instead of as the interface. A typical example is the “desktop” metaphor. People think of the system as a desk and all objects – files, folders, even a dustbin, are placed on the desk. To find a file in the file folder, is analogous to finding files in the file folder on the desk, instead of clicking the folder and clicking the desired file.

Virtual environments are a good example of direct manipulation applications. The objects, as well as the actions, are all visible. In real life, people interact with objects directly and most

virtual environments simulate the real world situations; therefore, direct manipulation is thought to be the most useful and efficient way of interacting with objects in virtual environments.

### **2.2.2 Examples of Direct Manipulation in Virtual Environments**

Direct manipulation or direct user interaction in virtual environments refers to the techniques that users employ to interact directly with virtual objects within virtual environments. Several researchers, [61, 62, 87], are working on direct manipulation in virtual environments for it is widely believed that the interaction methods should be as natural as possible [86, 97, 100]. It is also assumed that the most realistic system will be the most useable and useful systems [9]. Additionally, the sense of presence should increase as suggested by Sheridan [71]. We will describe some interaction techniques proposed for immersive virtual environments. For the purpose of this research, we are only concerned with the interaction techniques.

#### **Worlds in Miniature (WIM)**

The “Worlds in miniature” (WIM) metaphor is one example of direct manipulation in immersive virtual environments. It is an approach for manipulation, selection and navigation in an immersive virtual environment. It was proposed by Stoakley *et al* [87] and offers a second viewpoint of the virtual environment. The virtual environment is miniaturized into a hand-held size, as shown in Figure 4, and the users can rotate WIM to get a different view of the world. In such a way, the users can view objects hidden from view and manipulate objects that are not reachable in the virtual environment. Thus, there are two ways of manipulating and selecting objects: one is moving or selecting the objects in the virtual environment and the other is moving or selecting the proxy in the WIM. Moving objects on the model moves the corresponding objects in the virtual environment and vice versa [87].

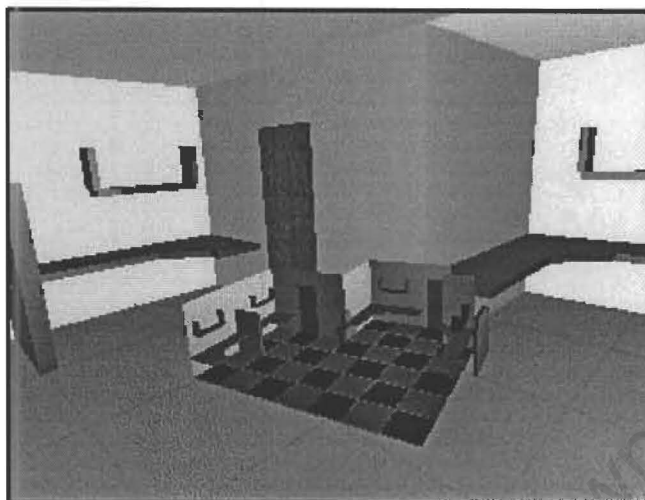


Figure 4: **The world in miniature (WIM)**. The world in miniature is on top of the life-size world [87].

As WIM is attached to the users' hand and it is separate from the virtual environment – there is no explicit mode required for changing between WIM and the virtual environment [87]. However, it forces splitting the display real estate between the original environment and the miniature copy [48]. In addition, it is difficult to select, and manipulate objects especially for fine-grained manipulations as the entire environment is scaled down into a hand-held size [48].

WIM allows the users to interact with object directly – either manipulate the actual life-size objects or manipulate the objects in the miniature. The metaphor is quite intuitive and easy to understand because it uses direct manipulation and it requires little learning. The metaphor, however, cannot maintain the sense of presence because working with two different worlds emphasizes that the world is artificial. Additionally, the users need to hold a clipboard or some kind of prop for WIM [87] which is physically different to the miniaturised world and hence reminds the users that the world is not real.

### **Voodoo Dolls Technique**

The “Voodoo Dolls technique” was proposed and implemented by Pierce *et al* [61]. This technique allows the users to interact with objects that are beyond one's physical reach. *Dolls* are hand held, transient objects; they are replicas of the objects in the virtual environment. The user creates dolls to represent specific objects and the dolls vanish once they are no

longer needed. The users use two-hand interaction – the dominant hand works in the reference frame defined by the non-dominant hand. In Figure 5, the soldier on the left (non-dominant) hand is served as a frame reference for the right (dominant) hand; while the right hand moves closer to the left hand, instead of moving in the physical location of the right hand, the pin (held in the right hand) is moving closer to the soldier. In such a way, the users can move two objects related to each other easily and quickly. For instance, if one wants to put a lamp on a table, one creates both objects and holds the table on the non-dominant hand and the lamp on the dominant hand. Now placing the lamp doll on the top of the table doll causes the lamp to be moved on the table in the virtual environment.

The Voodoo Dolls interaction technique allows direct manipulation of an object by creating miniature copies of the object. This method does not require any other physical objects, such as a clipboard in the WIM technique; and it gives the users an illusion of interacting as if in real life. However, the miniature copies of objects have different properties when they are held on the user's right or left hand [61]. This feature might confuse novice users since they might not fully understand the concept of the metaphor.



Figure 5: **Voodoo dolls technique.** The user can use both hands to interact with *dolls*. The non-dominant hand defines the frame of reference for the dominant hand.

### 2.2.3 Command Based User Interface

Although the most natural way to interact with virtual environments is direct manipulation, it is necessary to have some other indirect interaction methods. Mostly it is because some tasks cannot be performed with only direct manipulation. For instance, to add text to a virtual wall, one needs to select “add text” instead of writing on the virtual wall by hand. Designing non-command user interfaces is thought to be the next generation of user interfaces (for both the traditional user interface and virtual environment user interface), user commands will not be completely eliminated because not all tasks are suited for direct manipulation [9].

One possibility of issuing commands in virtual environments is the use of voice-recognition. Using voices to give orders is the usual and most natural method in real life; hence it should be appropriate in virtual environments as well. However, the users need to learn the valid commands and accurate speak-recognition is difficult to achieve [9, 36].

Menu-based commands are the alternative approach and overcome the problem of forcing users to “recall” commands – it is much easier to recognise them [20]. Menus in virtual environments are called “virtual menus” in [47]. The menus are used to store selectable options in user interfaces. They allow the users to perform some tasks that are difficult to specify by direct manipulation. In general, the principles for designing useful virtual menus are consistency, feedback, natural mapping, and good visibility [36, 47].

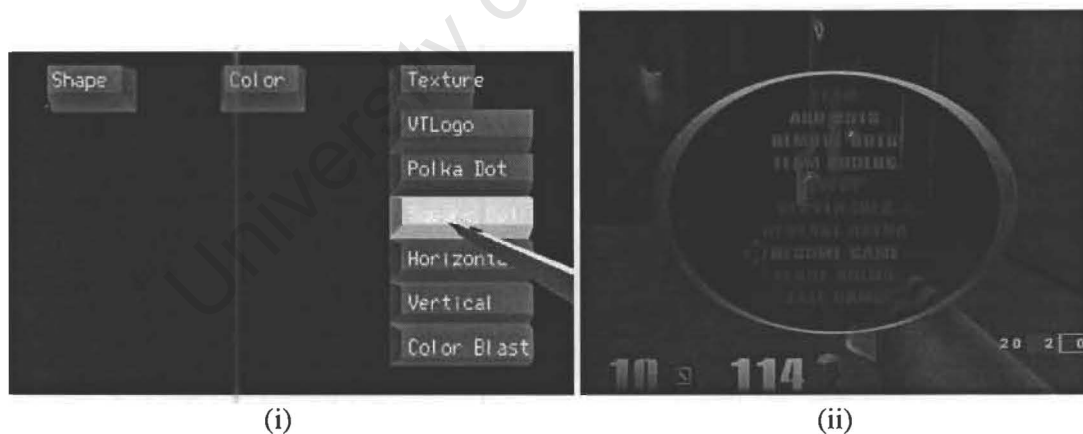


Figure 6: **Floating menu systems.** (i) This is a floating menu system (for immersive virtual environment system) implemented in [10]. The titles of the floating menu are always visible. Selecting a menu title causes the menu items to drop down. (ii) This is the floating menu from a first-person shooting game (Quake 3). It is used in desktop virtual environment system.

Virtual menus limit the number of possible choices and the users (especially novice users) can easily issue appropriate commands by selecting the desired option [9]. They are commonly used in desktop computing and they are familiar for most users. Our goal is to build a system for desktop virtual environments. Thus, it is worthwhile to have virtual menus in virtual environments. Figure 6 shows two examples of virtual menus.

#### 2.2.4 Modes in User Interfaces

Virtual environments, as well as any direct manipulation user interface applications, need some methods to let the system determine the meaning of the user's actions. Modes provide a way of indicating the user's actions. There are two major ways of using modes – one is mode per system and the other is mode per object [81].

“Mode per system” is the most commonly used approach. This method allows the whole system to be set to a particular mode and the users work with this mode until they switch to some other mode. For example, authoring tools usually have a mode for designing and a mode for execution. The discrimination of the modes is pre-defined by the system. In different modes, objects behave differently. For instance, in the design mode, a property sheet of a button appears as the button is clicked and in the execution mode, clicking the button executes the function of the button. The inconsistency confuses users especially when there is no adequate indication of the active modes [46, 81]. Figure 7 is an example of *mode per system* user interface.

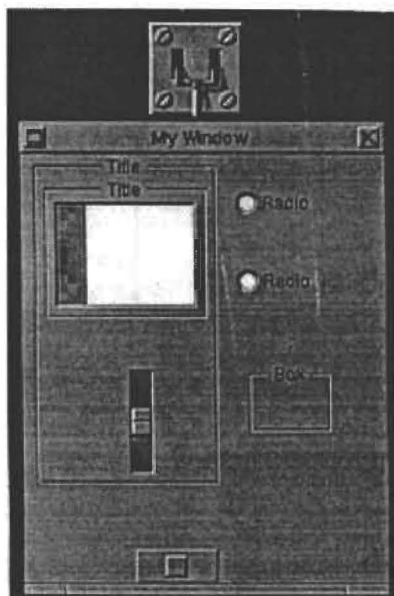


Figure 7: **The mode per system approach user interface.** This interface is one example discussed in [81]. The knife switch object is able to switch the mode from mention to use.

“Mode per object” is another way of using modes. It allows different objects in the same application to be set in different modes (see Figure 8). This means that there will be multiple active modes in one environment [46, 81]. For instance, one button can be edited while the other button is executable. Furthermore, the approach is normally used with affordances, so that one can know which objects are in design mode and which are in execution mode. The advantage of this approach is that the users do not need to switch between modes once the mode of an object is set. Besides, it is possible to use the rest of the environment even if some part of the environment is being edited.

There is no evidence showing that one approach is better than the other; and whether the approach is appropriate depends on the purpose of the virtual environment applications. In this research, we will investigate the mode per system and the mode per object approaches in our virtual environments. We hope that we can conclude on the appropriate method for the types of virtual environments.



Another design strategy is to borrow the ideas and techniques from other disciplines. The ideas and techniques, such as affordance [28], and colour coding [28], are important to good user interface design. There are some ideas that can be learned from cinematography since these techniques have been researched in order to make the viewers more focused on the scene [28].

Affordances are one good strategy that originated from industrial design. Affordances imply a certain action that one can act on an object [50]. Well-designed affordances help the users to understand how the objects work. For instance, a button affords to be pushed and the door handle affords pushing or pulling as shown in Figure 9. Again, virtual environments mimic the real world and in the real world, affordances work well.

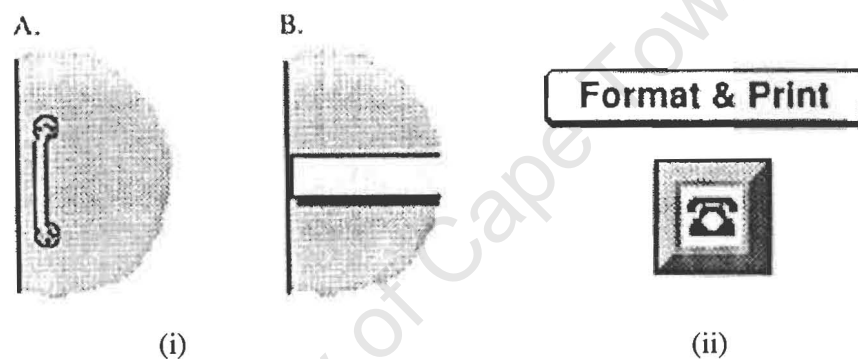


Figure 9: **Examples of affordances.** (i) Different door handles affords different actions [23]. (ii) Buttons used in computer systems. Both of them afford pushing [23].

### Using 2D User Interface Design Principles

The last approach is to use 2D user interface design principles. These principles should help UI designers to design usable and useful interface. However, there are differences between conventional user interface and the virtual environments.

*The difference between working in a conventional computer environment and working immersed are analogous to the differences between a craftsman at a workbench and one moving about a worksite wearing a toolbelt [48].*

Due to the differences, the conventional user interface guidelines are believed to be only partially applicable in 3D environment applications [15]. For example, WIMP (Windows,

Icons, Menus, and Pointers) is successful in a 2D interface, but it is not appropriate in 3D environments. One possible reason is that WIMP forces the separation of the virtual environment and the user interface [28], which is not desirable because the awareness of user interface would reduce the sense of presence [97]. Another reason is that some phenomena, such as the third dimension in virtual environments are not considered in current desktop interfaces and therefore we cannot apply these guideline blindly.

### **Choosing Strategies**

It is not clear when one strategy is better than the other. There are pros and cons for each approach. Therefore, while designing user interaction techniques, we need to examine the purpose and the nature of the virtual environment applications and the tasks involved. Then we can decide which strategy is better.

## **2.3 Summary**

In this chapter, we have discussed the work relevant to our research. Firstly, we described the definition, theory, and factors affecting presence.

Presence in this research is defined as the subjective or psychological sense of being in a virtual environment. We present the factors contributing to presence. Among the factors of presence, we are most interested in interaction. We believe that in order to increase the sense of presence experienced in virtual environments, the user should be able to interact with objects in the virtual environment as close to natural methods as possible.

The methods for measuring presence are discussed. The methods are divided into two categories: subjective measurement, and objective measurement. As presence is the subjective experience, it is reasonable to measure presence subjectively. Besides, it is easy to administrate, it is inexpensive, and it does not interrupt the experience of virtual environments.

We have presented useful user interface approaches – direct manipulation, command based user interface, and modes. We have discussed the importance of direct manipulation in desktop computing and we have suggested that direct manipulation is important in virtual environments because virtual environments have all the properties of direct manipulation. Thus they are an example of direct manipulation applications. The command based user interface is a different approach to direct manipulation. Nevertheless, we have discussed the usage and importance of command based user interfaces in virtual environments since in

virtual environments, there are some actions that are not possible to perform with direct manipulation.

Since there are several different interactions available in VE authoring tools, a strategy is required to determine the meaning of the user's action. Using modes is one possible solution for this. Therefore, we have discussed modes in user interface.

Lastly we have discussed several strategies that could be used to design a usable and useful virtual environment. These strategies include using some principles from 2D interface design, the methods used in other disciplines, and experience from real life.

## Chapter 3

# Problem Specification and Proposed Solutions

This chapter focuses on the Alice authoring tool because it is designed to be easy to use for experienced VR authors through to novices. However, we are concerned that there may be some problems with respect to presence and human-computer interaction. In this chapter, we will present the problems.

We begin with a description of the Alice authoring tool in Section 3.1. The reasons that we consider presence and human-computer interaction (HCI) in Alice are also presented. We will discuss the problems found in Alice which are associated with presence and human-computer interaction in Section 3.2. In Section 3.3, we present the possible solutions for solving the problems of presence and usability. We will end this chapter with a summary in Section 3.4.

### 3.1 Alice, a 3D Interactive Authoring Tool

This research is mainly concentrated on VR authoring tools and how to create a novel metaphor for the virtual environment so that the system can be used as an authoring tool and a VR application. The Alice authoring tool is one of this genre. We are interested in Alice because it uses several good designing principles to make it easy to use and easy to learn. In this section, we will first describe Alice, and its features.

Alice was developed by the User Interface Group at the University of Virginia and is now under development at Carnegie Mellon University [4, 54, 55]. Alice provides ease of authoring; therefore even non-experienced users can create and edit a 3D environment. One of

the biggest improvements Alice gives over other authoring tools is that it allows the users to create a virtual environment without knowledge of computer programming and computer graphics [55]. Therefore the users can concentrate on designing the virtual world instead of the programming language.

Alice is written in object-oriented language, C++, and the control of environments is specified by the interpreted language, Python [63]. With an interpreted language, the effect of the interaction can be seen without re-compiling the entire program [55].

As Alice supports users who do not have any knowledge of computer programming and computer graphics, there are several features which use common knowledge instead of computer graphical terms (such as using forward, left, right, and down instead of the direction along X, Y, and Z axes) [54, 57, 59]. Because of its ease of use, it is used as a rapid prototyping environment for conducting virtual reality research in Carnegie Mellon University, such as [58] and [87], and Alice is also used widely in various levels of classrooms [55, 87].

There are three ways of modifying a running Alice program. They are:

1. Support for immersed, interactive users to manipulate objects within the virtual reality directly.
2. The users can dynamically modify source code to restart the simulation or to change the current state incrementally.
3. The user can use a graphic user interface (GUI) control panel to make changes. [54]

Figure 10 shows the three ways of changing the state of a running Alice program. In the opening scene of Alice, the users can move objects directly with a mouse clicks and drags. The options for the mouse modes can be selected in the control panel as shown in Figure 10 (i).

The users can start the animation simply by clicking on the "Start" button at the top of the window when the "Animations" pane is active. In the animation panes, the users can specify how the objects in the scene will animate; the events, such as a mouse click, can also be specified in the event panes. The animation pane is shown in Figure 10 (ii).

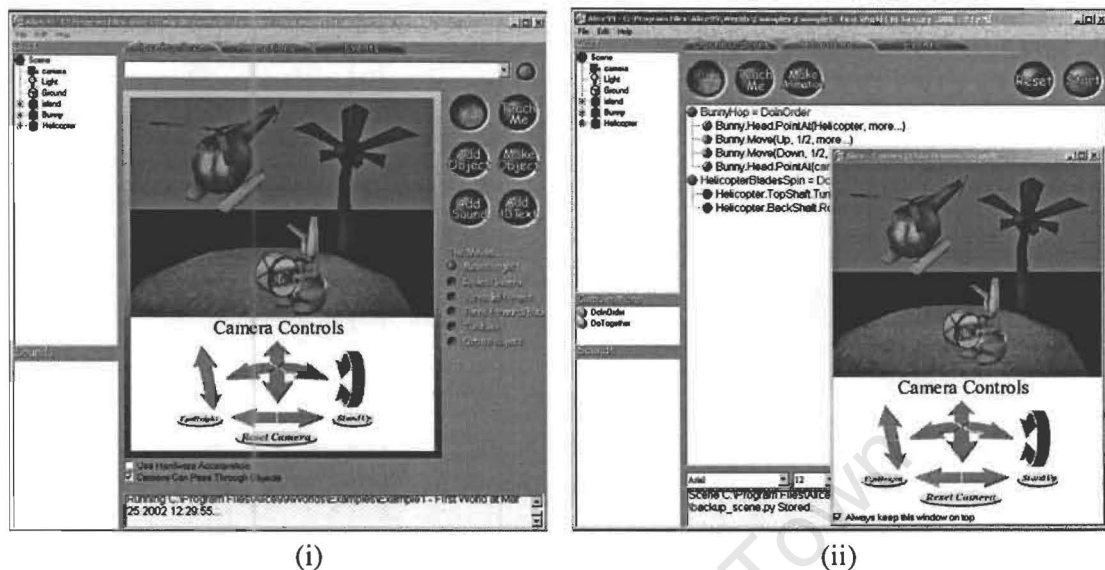


Figure 10: **Alice authoring tool**. The objects can be moved with mouse directly clicking and dragging on the objects in the scene. (i) The user can interact with objects with the control panel at the right hand side. (ii) The “start” button at the top of the window allows the users to start the animation. The animations are specified using Python.

### 3.2 Problems in Alice

The three ways of interaction in the Alice program, as mentioned above, are easy to use [4]. However, we are concerned about some problems which may cause by the three interaction methods.

The problems are associated with presence and usability. They are:

- disruption of experiencing with presence
- unnatural interaction methods
- the problem of “mention and use”

In the rest of this section, we will discuss the above problems.

### **Disruption of Experiencing with Presence**

Presence is thought to be the central issue of a virtual reality, even though the actual usefulness of presence is still unknown [75, 86]. One might argue that presence is not an issue in VR authoring tools. However, we believe that it is important to maintain presence in Alice-like VR authoring tools because this kind of authoring tool is not only an authoring tool, but also a VR application – people can view the virtual world as they construct it. Additionally, being immersed in a virtual environment makes the process of creating and editing the virtual world more interesting.

Presence is the relationship of an individual to an environment. The degree of presence is quantifiable and may vary more or less continuously. At any moment, people receive the stimuli from the physical world and the virtual world. As people increasingly respond to the stimuli generated by computer; the sense of presence in the virtual environment overtakes the sense of presence in real world [79]. Conversely, people lose the sense of presence in virtual environments if they focus more on real world stimuli. The causes of shifting attention are many. Factors include the virtual environment sensory information glitches, the weight of HMD, the auditory input from real world and the internal mental process, such as suddenly realizing that the world is not real [79].

On top of this, the bad design of user interface may also interrupt the sense of presence – the users are aware of the user interface and hence they are reminded that the world is artificial. In Alice, the users are required to change between environments for modifying and for experiencing the virtual environment. Changing between windows has a definite potential to disrupt the experience of presence. In desktop virtual environments, the disruption may be subtle since one simply changes windows. However, in immersive virtual environments, the disruption introduced by changing environments can affect the sense of presence hugely because the users need to remove the HMD to make changes, and put it back on again once the changes are made. Even in desktop VR, the sense of presence is interrupted by the awareness of the interface (in case of desktop virtual environments), or by the physically change of the environments (in immersive virtual environment situation).

### **Unnatural Interaction Method**

Another problem caused by using separate environments for design and execution is unnatural interaction methods. Virtual environments usually attempt to mimic reality. Having different environments for design and execution, however, makes the experience inconsistent with the real world. The interaction method, therefore, is unnatural. Several researchers believe that the more natural the interaction in virtual environments, the greater the sense of presence

experienced [86]. Additionally, the users can learn the interaction methods more easily because of their naturalness and the consistency with real life experience.

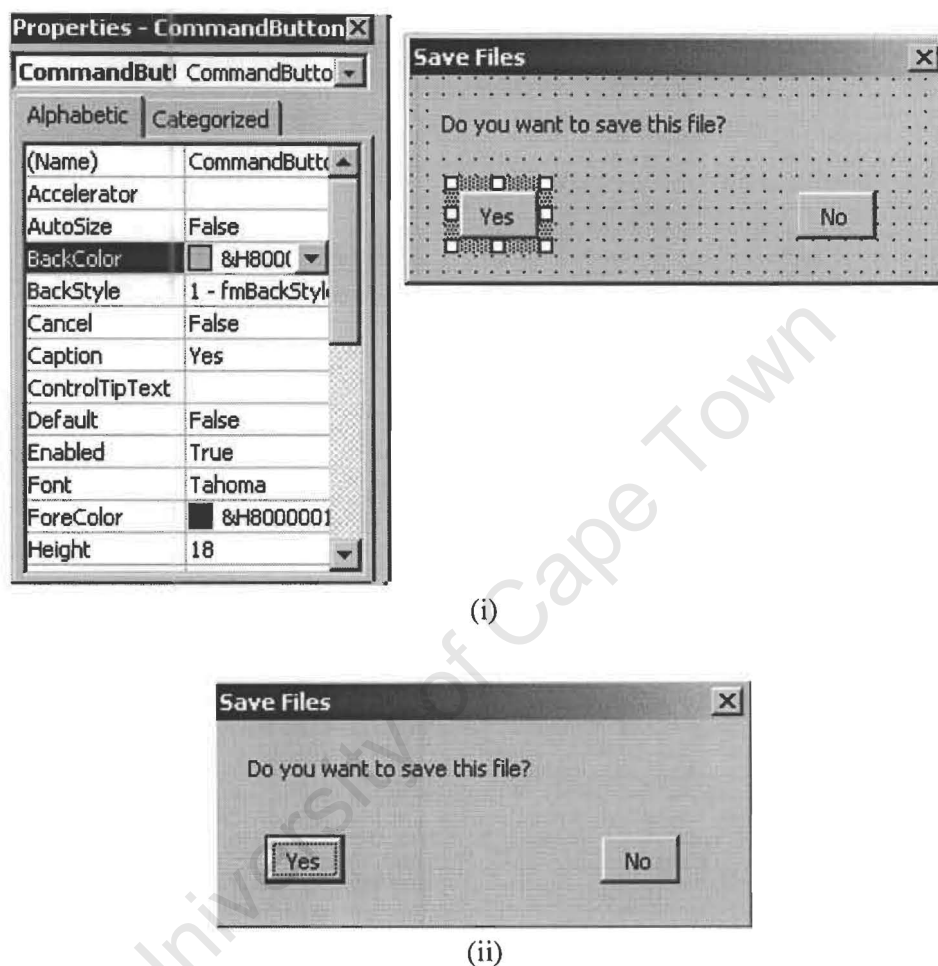


Figure 11: **An example of mention and use.** (i) The user interface in design mode (mention). The property sheet of the button is displayed. When clicking the “yes” button, the user can edit the text of the button. (ii) The “save files” window is executed. Clicking on the “yes” button results in saving the file.

### Mention and Use

Virtual environments may exhibit some problems that occur in 2D user interfaces because they are a special kind of direct manipulation interface. In Alice, we have identified a problem

which can be found in most direct manipulation interfaces for programming languages – the “mention and use” problem.

“Mention and use” is a major issue in most direct manipulation interfaces for programming languages. “Mention” refers to altering an interface component without invoking its functionality: for instance, to change the text that appears on a button. This is shown in Figure 11 (i). “Use”, on the other hand, refers to invoking the represented function of the interface component without changing the attributes of the button [81], such as pushing the “yes” button to save the file as shown in Figure 11 (ii). Thus, the interface components have two different behaviours depending on the mode that is currently active [44]. This inconsistency is not desirable in user interfaces [50] – it confuses the users and makes the interface difficult to learn.

Virtual environment authoring tools, like Alice, are quite similar to programming language in this respect. Both of them are used to create and edit (design mode, or mention), and the users are allowed to experience the result (execution mode, or use). In Alice, “mention” and “use” are explicitly distinguished. There are two modes; one is design mode (or mention), and the other is execution mode (or use) [81]. This method of distinguishing “mention” and “use” is called the “modal approach”, which is one type of the “mode per system” approach [81]. The discrimination between mention and use is pre-defined by the system. The users have to use different modes for designing and for execution. In Alice, for instance, when the mode is set to “Opening scene”, the users can move objects in the virtual environment whereas when the animation is started, the mouse clicks on objects result on triggering the event (if it is specified). Distinguishing mention and use solves the “mention and use” problem by limiting the actions available in the system. The users know and will learn what they can do and what they cannot do in a particular mode.

In spite of the clear feedbacks provided by the modal approach, it does not preserve the sense of presence in virtual environments. For example, changing from a running animation to the “Opening Scene” in Alice, the sense of presence is interrupted. Additionally, in a poorly designed user interface, using separated programs or environments might make the interface difficult to learn and to use [44]. With separated environments, the users are required to remember how to change between modes and which modes belong to which actions. Also, a lack of adequate indication of which mode is currently active confuses novice users, sometimes even expert users [46].

## Summary

We have identified the problems found in the Alice authoring tool. These problems are associated with presence and HCI. In this research, we will consider solutions to try to reduce and eliminate these problems (if possible) and to create better virtual environments in terms of presence, HCI, and usability.

## 3.3 Proposed Solutions

We have investigated several existing approaches that could be used to solve the problems that we find in Alice. These approaches are single-window approach, the tool approach, and the “mode per object” approach. In this section, we will describe each approach in detail.

### 3.3.1 Single-Window Approach

Firstly we have identified that multiple-window approach causes problems in terms of presence and usability. One improvement would be to eliminate the use of multiple windows. We propose using only *one* window in editing virtual environments. This means that *one* window, or *one* environment is available. Thus breaks in presence are reduced.

Besides reducing the possible breaks in presence, this approach improves presence by means of improving the naturalness of the interaction. The single-window method is consistent with everyday experience – the environment that people walk and look around is the same environment in which people modify objects. Because of the natural interaction method, we expect the sense of presence to be enhanced.

However, there is a problem with this approach – mention and use. There is now only one environment, yet two types of interaction. Therefore there must be some mechanism to distinguish between “mention” and “use” within this environment.

### 3.3.2 Tool Approach

The “tool” approach is one of the proposed solutions for the problem of “mention and use” in a single window[81]. Although this solution is proposed for direct manipulation user interface programming languages, it should be feasible in virtual environment authoring tools.

The tool approach is derived from everyday experience by adopting the interaction methods used in real life. One way of interacting with objects in real life is direct manipulation and the

other way is indirect manipulation. To interact with objects directly, people use their hand to grab a pen or move a chair. There are no tools required. On the other hand, interacting with objects indirectly, one needs to use some tools or devices. For example, one writes words with a pen, not by moving their finger on paper. Figure 12 illustrates the difference between direct manipulation and indirect manipulation in real life.

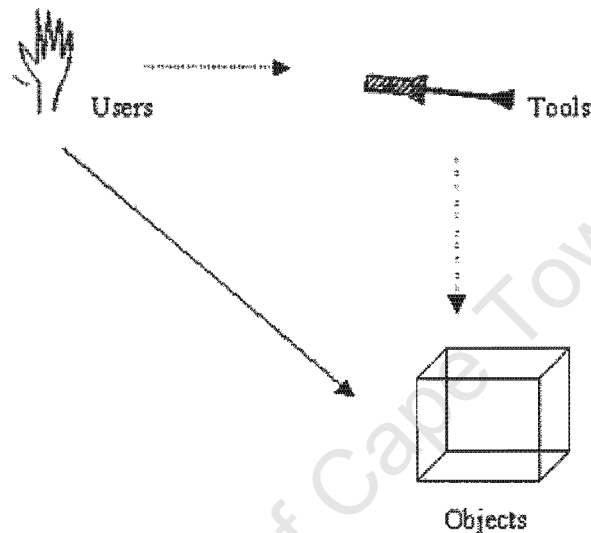


Figure 12: **Interaction in real life.** The dashed line represents indirect interactions and the solid line is direct interaction.

The Tool approach uses the idea of indirect manipulation. Every interaction requires some kind of tool, e.g. a “paintbrush” is used to paint objects, and a “hand” tool is for grabbing objects. The tools are represented as buttons. People can select tools with mouse clicks and then directly apply them onto objects [81], as people use tools in real life. This approach, though using modes in a disguised form, is claimed to be modeless because the tools are part of the environment [46]. Figure 13 is an example of the tool approach for editing user interface.

Different tools can be seen as different modes. The tool approach, thus, is a form of “global mode” or “mode per system”. (Details of “mode per system” are discussed in Section 2.2.4.) However, the tool approach might confuse users if the active tool (or mode) is not clearly indicated.

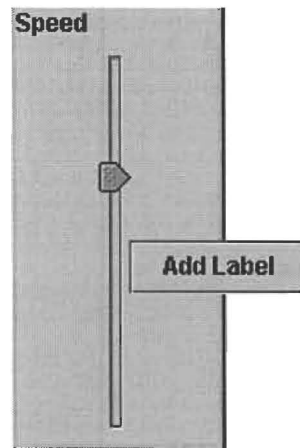


Figure 13: **An example of Tool approach application.** The button, labelled “Add Label” is used as a tool for changing the label of this user interface. When the button moves on top of the user interface, the user interface is set to this mode. When the button is removed from the user interface, the mode is set back to execute.

### 3.3.3 “Mode per Object” Approach

An alternative to the tool approach is adopting a mode per object. This approach allows different objects to be in different modes, meaning that there will be multiple active modes in one environment. The details of “mode per object” approach are presented in Section 2.2.4.

An example of using the “mode per object” approach is in Marsden [46]. A screw is attached on every user interface component. One can unscrew it to edit the user interface [46]. Figure 14 is a screen shot of Marsden’s interface. Thus in one environment some interface components are in design mode, while some are in execution mode. This approach relies on affordances and in the example, a screw is shown on the corner of the interface component to indicate mode. This screw affords to be screwed and unscrewed – indicating whether the interface component is currently editing or not.

In virtual environments, it sounds reasonable to use the “mode per object” approach. For instance, in real life we can write on a paper (mention) while the computer is running with some programs (use). The difficulty of this approach are visual cues indicating state because they consume some space on the virtual objects. Additionally, these visual cues would

diminish the sense of the presence as users are aware of the artificial visual cues which appear on the virtual objects.

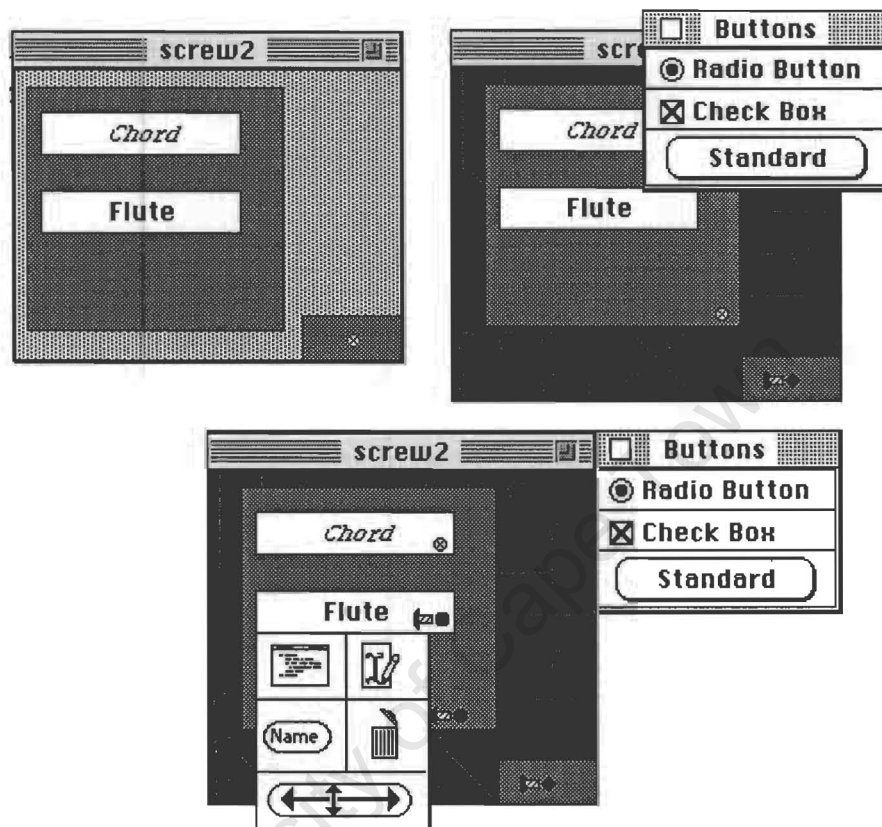


Figure 14: **Marsden's user interface.** When the screw is in, clicking on the buttons causes the function of the button to be triggered (as the left-up window). When the screw is not in, a property sheet appears, as in the right-up window. One can add, delete, or modify buttons on the pane, and one can "use" the buttons as well. Additionally, the screw of the children of this user interface component appears.

### 3.4 Summary

In this chapter, we have described Alice, a 3D interactive authoring tool. We are interested in Alice because it is the kind of virtual environment authoring tool which allows the users to experience the virtual world as they construct it. Furthermore, Alice can be used by users of

any experience level. Additionally, Alice uses several good design principles to make it easy to use and easy to learn.

We then discussed the problems found in the Alice authoring tool from two points of view: presence and human-computer interaction.

The problem with respect to presence is that the experience of presence would be broken in Alice-like authoring tools because users need to change environments for editing and viewing the environment. This kind of virtual environment authoring tool is used mainly by end-users, not virtual environment designers. It allows the users to create a world, and also view and experience the virtual environment. Thus, presence is important.

Additionally, the unnatural interaction methods reduce the sense of presence. Changing environments for designing and viewing is unnatural and it is not consistent with everyday experiences. Therefore presence might diminish.

The other problem in Alice is “mention and use”. “Mention and use” is an old problem in the user interface for programming languages. Virtual environment authoring tools suffer from this problem as well because they are used to create and view virtual environments using the same program. Alice uses the “modal approach” to solve this problem. The system explicitly distinguishes between mention and use. Therefore objects behave differently depending on the active modes. This inconsistency is not desirable in the user interface.

Finally we present some possible solutions to these problems. The single-window approach solves the problem of breaks in presence and the problem of an unnatural (two distinct environments) interaction method as well. This idea is borrowed from everyday life experience – people work on, and use an object in the same environment. The single-window approach, however, requires some mechanism to indicate the user’s action. The tool approach restricts people to work with the currently active tool and interact with objects indirectly. On the other hand, the mode-per-object approach allows the users to apply different tools on different objects at the same time. We need, therefore, to investigate which of these approaches provides the best solution for a novice VR author.

## Chapter 4

# Design and Prototypes of Virtual Environments

Our goal is to build a desktop virtual environment which is useful to a wide range of people, ranging from experts to novices. This virtual environment tool combines the features of VR applications and those of VR authoring tools. Therefore in one system, and in one environment, the users can experience the virtual world as well as being able to edit its contents. Additionally, our virtual environments should reduce, or eliminate, the problems discussed in Chapter 3.

In this chapter we will present the prototype virtual environments. We have built four virtual environments with three different metaphors. One is based on current VR authoring tools; and the other three virtual environments are using the design strategies that we discussed in the previous chapter. The four virtual environments will be used to conduct experiments.

In Section 4.1, we present the basic interaction which is implemented in all virtual environments. There are addition interaction options for the purpose of the experiments. In Section 4.2, we describe the virtual environment, which is a mimic of the traditional virtual environment authoring tools. The interaction method, and the strategies used for changing modes are presented, too. In Section 4.3, we present the virtual environment adopting the tool approach and the single-window approach. The idea behind the design and some issues of the interaction methods are discussed as well. Lastly, in Section 4.4, we present the virtual environment which employs the mode per object approach. This virtual environment, similar to the tool approach virtual environment, reduces the problem of break in presence. All four virtual environments will be used for user testing, which are discussed in Chapter 5 and Chapter 6.

## 4.1 Interaction Options

As suggested in [36], there are four main categories of common user input tasks: create/edit, viewpoint manipulation, and navigation. In this section, we describe create/edit only. Navigation and viewpoint manipulation will be discussed with respect to each prototype.

Creating or editing objects in virtual environments includes selection, translation, rotation, scale, object creation and object instancing [36]. In this research, we will use this grouping and consider only selection, translation, and rotation. Changing colours of objects is considered as an editing task as well.

### Selection

Selection refers the action of specifying or choosing objects for some purpose [47, 8]. In Mine [47], two basic categories of selection techniques are *local*, and *at-a-distance*. In a local selection mode, the object is chosen by moving the mouse cursor, or hand within the object's selection region and performing the per-defined signal to indicate selection [47]. In an at-a-distance selection mode, the users can select objects which are not within the reach of hands using laser beams or spotlights which project from their hand and intersect with the objects in the virtual world [47]. (See Figure 15)

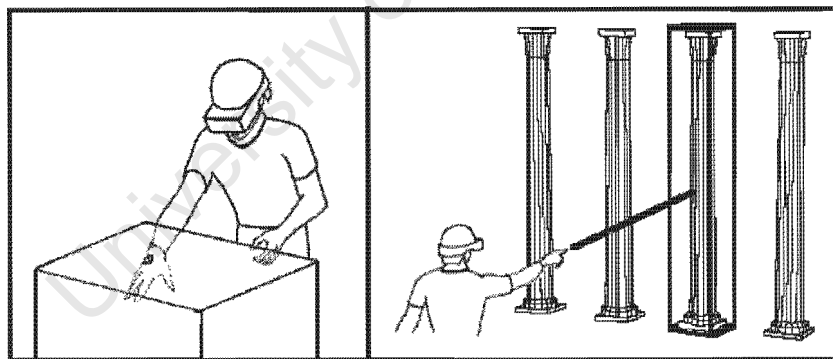


Figure 15: **Selection Techniques as Discussed in Mine [47].** The left hand side is the technique for selecting objects which are within hand reach. The right hand side illustrates the way to select objects at a distance [47].

environment. The users can walk around the circle and they can view all objects from any angle. Figure 16 shows the route that the users can walk. The users can walk from point A to point B and they will always face the centre of the room. As indicated in Figure 16, almost every object is in the user's sight at any point; therefore in the virtual environment, the users cannot turn their head because this action seems to be unnecessary.

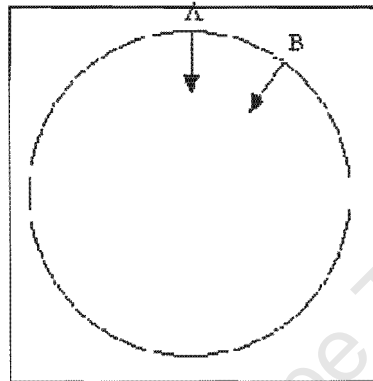


Figure 16: **The navigation in the conventional virtual environment.** The circle in the square is the pre-defined walk path. The users can walk from point A to point B and vice versa. The users always face the direction indicated in the arrows.

We do not consider navigation as a variable in our research, and both virtual environments used in the presence experiment have the same navigation. In addition, we do not need to implement any mechanism to detect whether the user would walk out of the boundary of the room.

#### 4.2.2 Interaction Method

This virtual environment uses a similar interaction method to most virtual environment authoring tools – using separate windows for editing and experiencing the virtual environments. In Alice, the interaction options are listed in a popup menu. The users can trigger the popup menu, which set the system into the chosen mode and brings up another window for editing.

In this virtual environment, we use this same approach to interact with objects. A popup menu appears while the mouse is over an object and the right mouse button is clicked. Figure 17

of the object; the Z axis is the line crossing the centre of the object and coming out of the screen. Therefore the centre of the rotation is the centre of the object.

In most 3D modelling programs, such as 3D Studio Max [1], there are three separate rotations. In our prototypes, we will use three rotations as well. "Pitch" is the name for rotation along the X axis, "yaw" is for rotation along Y axis, and rotation along Z axis is called "roll".

### **Colouring**

There are two ways of colouring objects in our prototypes. One is allowing the users to change only the place where the mouse clicks. This is similar to the way that people paint in real world and it is used in almost every image editing programs, such as Adobe Photoshop [2]. This way of colouring is named "paintbrush" in our prototype virtual environments.

In some drawing applications, a tool can use to fill the bounded object with colour. In *Paint* in Windows 2000 [52], this tool is called "fill with colour". In our prototypes, we have included this tool and it is called "magic wand".

### **Other Interactions**

Besides all these interactions, there are some other interaction options available, such as "use" in the pin approach virtual environment. These addition interactions are discussed in the specific virtual environments.

## **4.2 The Conventional Virtual Environment**

The purpose for this virtual environment is to illustrate the conventional VR authoring tool and to serve as a base for comparisons with the alternative virtual environment authoring tools. In this section, we will present the interaction method and the user interface of the first virtual environment. In addition, we will also discuss the issue of changing the viewpoint and navigational concerns.

### **4.2.1 Navigation and Changing the Viewpoint in the Virtual Environment**

In this virtual environment, we have allowed the users to change their position and their viewpoints as if they are walking in the virtual environment. We have set a pre-defined route in which the users can walk freely. The pre-defined route is a circle inside the virtual

tion of an object, or changing the colour, will be made and revealed in the main window, as shown in Figure 17 (i).

### **4.3 The Tool Approach Virtual Environment**

The tool approach virtual environment is one of our new designs for virtual environments and uses the tool approach, as discussed in Section 3.3.2. The goal of this virtual environment is to allow the users to edit the world within the virtual world, thereby maintaining the sense of presence. We have built two prototypes that use the tool approach to serve in the two experiments we wish to conduct. The differences between the two prototypes are the navigation and the interaction options in the virtual environment, but the interaction strategy (i.e. a single window) is the same in both prototypes. In this section, we present the navigation and changing viewpoint in the two tool approach virtual environments. We then describe the ideas and interaction strategy used in the virtual environments. Finally we present the interaction options in the two virtual environments.

#### **4.3.1 Navigation and Changing the Viewpoint in Virtual Environments**

Please note, that although we built four environments in total, two are based on the tool approach – one is used to compare with the conventional environment and the other to compare with the mode-per-object environment. The former prototype is referred to as “the tool approach VE (a)” and the latter one as “the tool approach VE (b)” in the rest of the document.

##### **The Tool Approach VE (a)**

This prototype is used in the presence experiment and it is used to compare the level of presence with the conventional virtual environment, discussed in Section 4.2. Therefore the navigation and the way of changing the viewpoint are the same as the conventional virtual environment. That is, the users can only walk on the pre-defined path as shown in Figure 16; and they always face the direction indicated by the arrows. The navigation keys and the field of views are the same as the conventional virtual environment.

##### **The Tool Approach VE (b)**

The tool approach VE (b) is used in the usability experiment. This is an extended version of the tool approach VE (a). In this virtual environment, the users can freely walk and look

Other interaction techniques are WIM [87] and Voodoo dolls [61] (details are discussed in Section 2.2.1). However, these two techniques require using the replica of virtual objects and we argue that this might diminish the sense of presence.

In our prototypes, we will use the following technique. In the local situation, the users can select objects by moving the mouse cursor to the desired object and clicking to indicate selection. In the at-a-distance situation, the users do not need to move closer to the object to select, but they can select objects which are visible as if the objects are in arm's reach.

### **Translation**

“Translation” is a mathematical term [13] for changing the position of objects. There are three directions of moving objects – along X, Y, and Z axes. Though most people understand the concept of these axes [13], X, Y and Z are not common terms and could increase the cognitive overhead for novice developers. Therefore, the mapping step from the natural language to the coordinate system should be avoided [13]. In Alice, the terms “forward”, and “backward” (moving along Z axes), “left”, and “right” (moving along X axes), and “up” and “down” (moving along Y axes) are used [4]. According to Conway [13], using common language is easier to understand and easier to learn. Therefore in our design, we will use the natural language terms.

We have grouped three directions of translation into two – one is moving in the image plane (which is along X- and Y-axes), and the other is moving in depth (along Z-axis). We do not think that it is necessarily to break the translation into three (one for each axis) as combining the move along X- and Y-axes are easily implemented and distinguishable. The translation on the image plane (moving in X and Y axes) is named “move” in our prototypes and the translation in depth is called “move closer and further”.

### **Rotation**

Rotation in virtual environments can be in three dimensions – rotation along the horizontal line, rotation along the vertical line and rotation along the line which is perpendicular to the horizontal and vertical lines. In Alice [4], each object has its own coordinate system. Therefore an object can, for example, turn left which means the centre of rotation is the centre of the object and it rotates along the vertical axis.

In our prototypes, each object has its own coordinate system for rotation. The X axis is the horizontal line across the centre of the object; the Y axis is the vertical line across the centre

around in the virtual environment. We have implemented a way to avoid the user walking into walls and out of the virtual environment; therefore, the limitation on the freedom of navigation is removed. The users can walk using the arrow keys on the keyboard, and they can rotate their heads with the mouse.

Using mouse and keyboard for navigation is used in several 3D games, such as Quake [64]. In one study [14], it was found that most users feel that using a mouse in conjunction with keyboard is easy to master. The effective rating of the motion tools found in [14] is relatively high. Therefore we will adopt this motion control in the prototypes.

### 4.3.2 Interaction Method

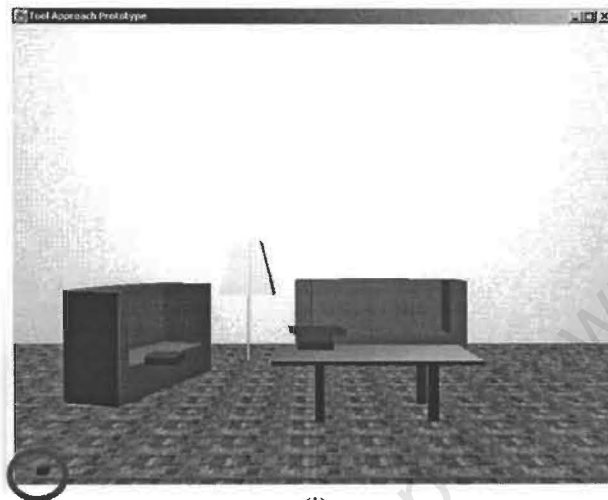
In previous chapters, we have discussed the “mode per system” approach and the “tool approach”. In Alice [4], the “modal approach” is adopted. As discussed in Section 3.1, the modal approach is usable— it provides a clear distinction between “mention” and “use”. The problems of this approach are that the sense of presence cannot be maintained and a lack of feedback of the active mode might confuse users. However, the goal of this research is to design a tool which can maintain the sense of presence in virtual environment authoring tools.

Therefore we employ the tool approach and the single-window approach in the virtual environment. The single-window approach should solve the problem of breaks in presence. The tool approach is one solutions of “mention and use” problem [81]. As the similarity of user interface programming languages and virtual environment authoring tools, we argue that the tool approach should be able to solve the problem of “mention and use”. Besides solving these two problems, the interaction method should be as natural as possible because the naturalness of the interaction method would influence the sense of presence [86, 97]. We, therefore, borrow the natural interaction strategy from real world.

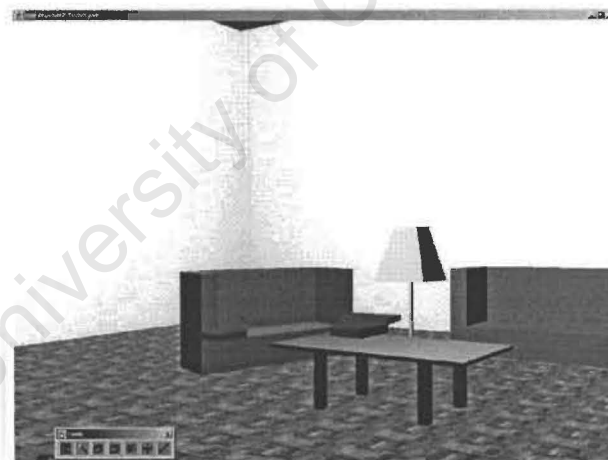
In real life, workers normally carry a toolbox to the working site. When they want to modify or fix an object, they take out an appropriate tool from the toolbox. Once the job is done, they put the tool into the toolbox and walk to another place with the toolbox.

Most virtual environments are quite similar to real life in this respect, and therefore we felt that using the toolbox idea in virtual environments is appropriate. To implement the idea, a toolbox is provided in the virtual environment as a 3D object that can be opened and closed. In real life, the toolbox is not seen until it is actively sought out. It is difficult to implement this in the virtual environment because objects outside the users’ view frustum are difficult to access [60]. Instead, we place the toolbox in a fixed location related to the user’s viewpoint.

Thus the users know where to find the toolbox when it is needed, and time is reduced in searching the entire virtual environment for the toolbox if it is located at a fixed position. Figure 18 (i) and Figure 19 (i) show the toolbox,



(i)



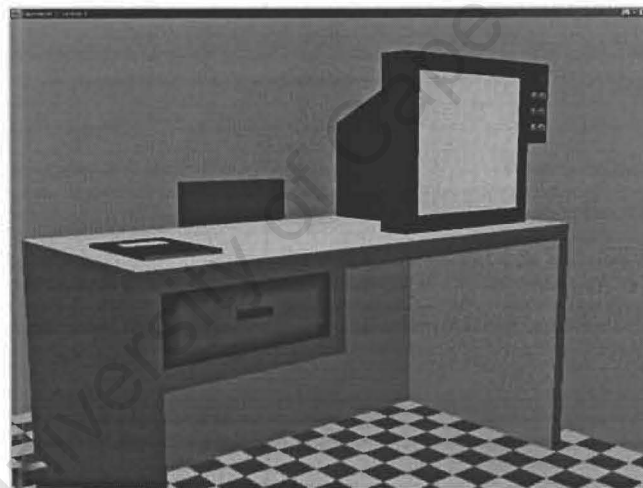
(ii)

Figure 18: **The screen shot of the tool approach VE (a).** (i) the toolbox is circled. It is in a fixed location relative to the users. (ii) After the users walk, the toolbox is still at the left-bottom corner of the screen. The tools arranged in the menu appear while the users click on the toolbox. Tools will disappear while the users close the toolbox or while the users walk in the virtual environment.

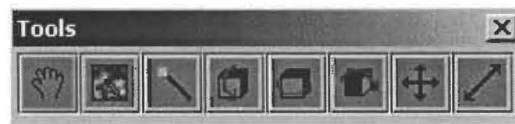
which is circled, and it is placed at a corner of the screen. Moreover the toolbox is a small box at the corner of the users' view frustum, and it will not block the user's view.

While the toolbox is open, the tools appear for selection. These virtual tools are represented as buttons and arranged in a virtual menu, shown in Figure 18 (ii) and Figure 19 (ii). In Houde [29], the different shapes of the mouse cursor are used as an indication of the action performed. In our work, we follow this method – once the desired tool is selected, the shape of the mouse cursor will change accordingly. For instance, while the “paintbrush” is selected, the mouse cursor will change to a “brush” shape. While the toolbox is closed, the virtual menu disappears automatically and the shape of the mouse cursor changes to default (i.e. the arrow shape).

The interaction method of the tool approach VE (b), which is shown in Figure 19, is the same as that of the tool approach VE (a); however the scene of the virtual environment and the number of interactions are different.



(i)



(ii)

Figure 19: A screen shot of the tool approach VE (b). (i) The scene of the virtual environment is different from that of the tool approach (a).

### 4.3.3 Interaction Options

The tool approach VE (a) has the same interaction options as the conventional virtual environment. They are the basic interaction options listed in Section 4.1.

Besides the basic interaction options, there is an additional interaction available in the tool approach VE (b). For the purpose of the usability experiment, the users are required to “use” objects. To use objects means to invoke the function of the objects, such as to close or open a book, or to switch on and off the light. Because the interaction method is using tools, we need a tool for using objects in virtual environment. This tool is called “use” and it is indicated as a hand shape in the tool list. Figure 19 (ii) shows the interaction options and the scene of the tool approach VE (b).

### 4.3.4 Navigation in the Virtual Environment

The VE(b) virtual environment is used in the usability experiment (along with the Pin approach described in the next section). As we are not interested in whether navigation influences presence and usability in virtual environment, we have to make the navigation in both environments. Thus, the users can walk and look around .

We have implemented collision detection to avoid the users walking into and through the walls. When the users attempt to walk into the walls, they will stay in the position; however, there is no error message to notify the users. We do not think it is appropriate since it is unusual to have a message saying that you are bumping the wall. Moreover we do not think that the users would try to walk through the walls as this action is abnormal in reality.

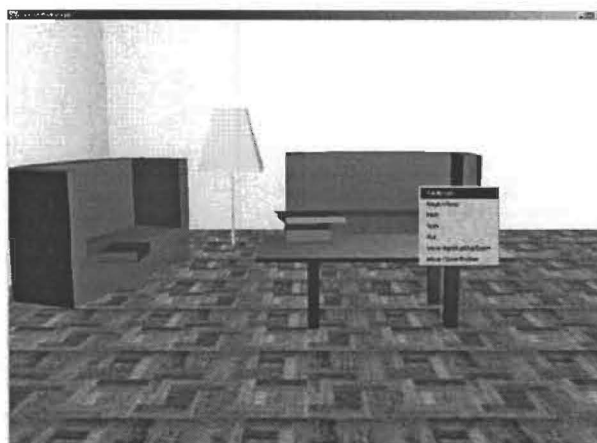
## 4.4 The Pin Approach Virtual Environment

The last virtual environment that we have built is using the mode per object approach. This is to set a particular object in the desired mode while the rest of the environment can be used as usual. A mechanism, or an indication, is required to indicate whether the object is in the editing mode or in the normal mode.

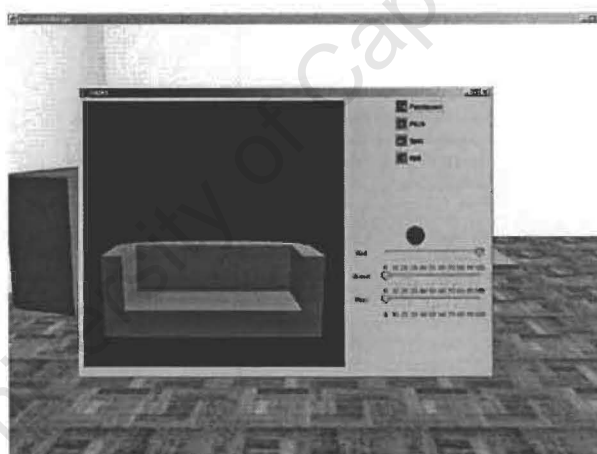
### 4.4.1 Interaction Method

The mode per object approach allows the users to “use” an object while the object is not in editing mode. For this reason, we have created a list of objects to populate the scene, each of

shows the popup menu and the second window (editing window) for editing objects. When the editing window is closed, any change in the editing window, such as changing in the posi-



(i)



(ii)

Figure 17: **The screen shot of the conventional virtual environment.** (i) A popup menu appears when the users want to edit an object, e.g. the chair. The popup menu consists of the interaction options. They are: Paintbrush, Magic wand, Pitch, Spin, Roll, Move right/left/up/down, and Move closer/further. (ii) The window on top is the editing window. In this window, only the interested object, the chair, is displayed. Besides selecting colours, three ways of rotation are available so that the users can paint the part that is not visible at the initial state.

which has a default “use” function. The list of objects is shown on Table 1. For instance, the users can open and close the door, however, they cannot move or rotate the door since the rotation and movement tools are not provided for the door.

Objects	Use (default action)	Tools
Walls, Floor, Ceiling	N/A	Paint/Magic wand only
Chair	N/A	All tools
Desk (with a drawer)	Open Drawer	All tools
Books	Open/Close	All tools
Lamp	Switch on/off light	All tools
Door	Open/Close	Paint/Magic wand only
TV set and the remote control	Switch on/off, change channels	All tools
Pin	Grab and drop	N/A

Table 1: **Objects and the functions associated with them.** The first column is the object in the virtual environment. The second column is the default function associated with it. This default function can be manipulated directly. The last column is the tools that can be applied to the object. For instance, all basic interaction tools can be applied to the desk and the drawer of the desk can be open and closed without any tools. There are no direct manipulations that can be done on walls, floor, and ceiling – only paint and magic wand can be applied to them.

As we mentioned earlier, we need a way to indicate the mode status of each object and an easy-to-understand metaphor to edit objects. Affordances are a possible way of indicating the action that the users can do on the objects. For instance, the close button on top of a window affords pushing and the result of pushing the button is to close the window. However, adding every affordances on objects, such as an affordance for painting, an affordance for rotation and an affordance for moving, in virtual environments will clutter the interface and make the world artificial. Fortunately we learn how to use “affordances” to indicate the way of modification from Marsden’s work [46], in which the screw is used as an affordance for starting editing.

We have employed this idea, and have drawn our metaphor from the way that an artist draws a picture. In real life, an artist would put drawing paper on the drawing board and pin the paper on. The pin is used to fix the paper on the drawing board. It can be also seen as an

indication that the drawing is in process. Therefore, to pin an object and then edit it is the metaphor we use in the virtual environment.

As the metaphor required, a drawing pin is provided in the virtual environment. The drawing pin, similar to the toolbox, is placed at the left-bottom corner of the screen and it is always in this fixed location. The drawing pin will not block the users' view and it is always available despite of the users' position (the system re-replaces the pin in the object so that it is always visible).

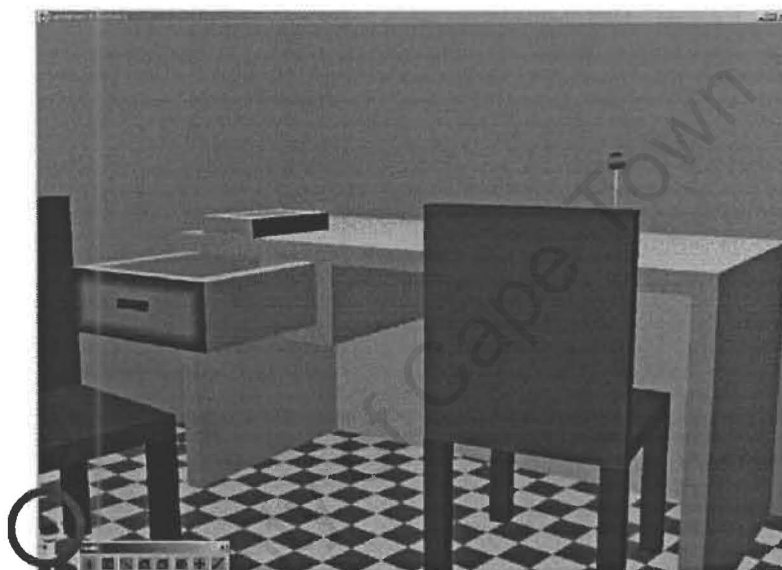


Figure 20: **The pin approach virtual environment.** The drawing pin is at the corner of the screen (circled). The chair is pinned and the pin attached to the chair is in a different colour from the drawing pin at the corner. While the chair is pinned, the users can still invoke the basic function of any unpinned object in the virtual environment.

To pin an object, the user simply drags the drawing pin and drops it on the object. To show that it is being edited, a new drawing pin will appear and attach on that object. A tool list then appears. The user can choose the desired tool and apply it to only that object. The users have to unpin the object in order to use the object, even when there is no tool mode set for that object. In order to avoid confusion between the drawing pin, which is always at the left-

bottom corner of the screen, and the pin attached on object, we use different colours to differentiate class from instantiation.(shown in Figure 20).

We have provided visual feedback to indicate the currently active mode of the pinned objects. While the mouse is moving over the pinned object, the mouse cursor will change the shape according to the status of the object [29]. Further feedback is provided via the texture on the pins, which are attached to the objects. The texture on the knob of the drawing pin will be the same as the active tools. Figure 21 illustrates the idea of changing texture of the drawing pin to indicate the currently active modes of the pinned objects.



Figure 21: **The texture of the drawing pin, which is attached to objects.** (i) When the object is in the paintbrush mode, the texture is changed to a paintbrush. (ii) The object is in the move mode, and the texture on the pin indicated this.

## 4.5 Summary

In this chapter, we presented four virtual environments that we have built. All four virtual environments will be used in the two experiments (two per experiment), discussed in the next two chapters.

We first described the interactions and modifications allowed in the four virtual environments. The basic interactions are: move in the image plane, push and pull (move in the direction perpendicular to the image plane), three ways of rotation, paint where clicked, and paint the whole object with one mouse click.

We then described the four virtual environments one by one. The first virtual environment uses the multiple-window approach. The multiple-window approach is the most commonly used approach in virtual environment authoring tools. There is one window for use

(experiencing virtual environments) and another window for mention (editing virtual environments). The editing environment appears by selecting the desired mode from the pop-up menu. The changes are reflected in the main environment when the editing window is closed.

The second and third virtual environments use the tool approach and the single-window approach. In the virtual environments, the users can edit objects within the virtual world, instead of outside the virtual world as with the conventional virtual environment. To interact with objects, only indirect manipulation is allowed in the two virtual environments. The differences between the two tool approach virtual environments are the freedom of navigation and the number of tools (interactions). One of them is the same navigation and number of interactions as the conventional virtual environment; and the other is the same as the pin approach virtual environment.

The chapter ends with the description of the fourth virtual environment – the pin approach virtual environment. It adopts the mode per object approach. A drawing pin is provided to change and indicate the mode of each object. Additionally, direct manipulation is allowed in the virtual environment.

## Chapter 5

# Presence Experiment

In this experiment, we are investigating whether our improved, single window, virtual environment can score higher presence than the conventional virtual environment. To conduct this experiment, we have built a virtual environment which allows the users to modify objects in the same environment as where they view the virtual world. This is unlike conventional virtual environment authoring tools, where users have to change between different windows or environments to interact with objects.

In this chapter, we, firstly, present the information about the participants that were involved in the experiment in Section 5.1. In Section 5.2, we describe the questionnaires that we used in the experiment for measuring presence and immersive tendencies in individuals. We then describe the equipment used in the experiment, the experimental tasks and procedure in Section 5.3, 5.4, and 5.5 respectively. In Section 5.6, we present the results of the experiment and the results are discussed in Section 5.7. We will end this chapter with a summary of the experiment.

### 5.1 Subjects

We need 30 participants in this experiment for a t-test [31]. The participants were paid volunteers and were students from various faculties in University of Cape Town.

Thirty participants were randomly divided into two groups of 15 people. One group used the conventional virtual environment system and the other used the tool approach virtual environment system. The former group is called “the control group” and the latter group is called “the experimental group” in the rest of the chapter. Table 2 shows the information of participants in each group. Some participants might have more experience with virtual

environments than others. For the purpose of the experiment, we define “expert users” as those who have played first-person shooting games or flight simulation games more than five times.

	Control Group	Experimental Group
<b>Average Age</b>	21	21.6
<b>Male</b>	11	9
<b>Female</b>	4	6
<b>Expert Users</b>	8	8
<b>Novice Users</b>	7	7

Table 2: **Information of subjects participating in the experiment.** The control group used the conventional virtual environment authoring tool and the experimental group used the new design of virtual environment authoring tool.

## 5.2 Questionnaires

The purpose of this experiment is to investigate whether users of the new virtual environment authoring tool can experience higher presence than conventional virtual environment authoring tools. As discussed in Chapter 2, there are several different approaches to measure presence. We chose questionnaires as they are cheap, easy to administrate and it is the most commonly used measuring method. In order to test the level of presence, we use a standard questionnaire.

The questionnaire that we use in the experiment is the presence questionnaire (PQ), developed by Witmer and Singer [97]. The virtual environments that we built do not contain any auditory and haptic information; therefore the questions concerning auditory or haptic feedback are left out in order not to confuse the participants. The PQ can be found in Appendix A.

Another questionnaire used in the experiment is the Immersive Tendencies Questionnaire (ITQ). This questionnaire is used to measure the differences between individuals and how well one can become involved with and focus on the virtual environments. The ITQ was also developed by Witmer and Singer [97]. We use all thirty-two questions in the experiment. The ITQ is shown in Appendix B.

### 5.3 Equipment and Prototype Systems

The machine that we use for the experiment is:

- AMD K7 processor 500 MHz
- 392 Mb memory
- GeForce 256 display card with 32 MB memory
- Running Windows 2000.

The prototypes that we used in the experiment were the conventional virtual environment and the tool approach VE (a) (refer to Section 4.2 and 4.3 for more details). Both virtual environments provide the same scene, navigation, and interactions. The only difference is that the former uses multiple windows (one is editing and one is viewing) and the latter uses a single window for interaction and viewing.

### 5.4 Experimental Tasks

Both virtual environments consist of a virtual living room with some furniture and other objects. The objects and furniture are a table, two chairs, a lamp and three books. The users can interact with objects as well as navigate (though navigation is limited). We set several tasks that we thought one might want to do in a virtual living room. Some tasks are simple, involving only one interaction tool, such as "Move the table to the right bottom corner of the room." Some tasks are more complex, and they require the users to use several different tools to perform the tasks. An example of this kind of tasks is "Put a green blob on the wall behind the lamp" which requires the users to use both the colouring tool and the movement tool. The tasks for both virtual environment systems were the same. The list of the experimental tasks can be found in Appendix E.

### 5.5 Experimental Procedure

Each participant was assigned to one session for the experiment and each session was about forty minutes long. The participants were led to an office, separate from the laboratory. They were asked to fill in the ITQ before they use the system. Once they finished the ITQ, an appropriate user manual of how to use the virtual environment and the task sheet were handed to them. The user manual for the control group is in Appendix C and the user manual for the experimental group is in Appendix D. The participants read the manual and then, before the

actual experiment began, the experimenter briefly demonstrated how to interact and navigate in the virtual environment. The participants could ask questions regarding the interaction and navigation in the virtual environments.

The participants were then left alone in the room to perform the tasks on the task sheet. The participants experienced and performed tasks for about twenty minutes or until they finished all the tasks. After experiencing the virtual environment, the participants answered another questionnaire – the PQ. After they filled in the questionnaire, the experimenter gave a short interview regarding the interfaces and experiences of the virtual environment.

## 5.6 Analysis of Results

In this section, we present the results obtained in the experiment. We, firstly, describe the hypotheses and the variables associated with it, that we intend to measure. We briefly discuss the procedure of gathering data and then we present the statistical analysis of the obtained data.

### 5.6.1 Variables and Hypotheses

We intend to test whether the single-window virtual environment can maintain presence and give a greater sense of presence than the conventional virtual environment authoring tool.

The variables that we are interested in are:

- **Presence (P) score:** Presence is measured by the PQ. The score calculated from the questionnaire indicates how involved one is in the virtual environment.
- **Immersive Tendencies (IT) Score:** IT is measured by the ITQ. This score indicates how easily one is involved and immersed in a given medium.

In the experiment, we expect that the P score of the experimental group should be higher than that of the control group. As discussed in Section 3.3.1 and Section 4.3, the tool approach prototype virtual environment, which uses the single-window approach, should give higher presence than the conventional virtual environment. Moreover, the IT score should be correlated with the P score as suggested by Witmer and Singer [97].

### 5.6.2 Gathering Data

We used a shortened version of the PQ with only the relevant questions included for the experiment. There are eighteen questions used in calculating P scores, although the participants answer all twenty-four questions. (This is because, as suggested at the end of the scoring instructions, questions from eighteen to twenty-four are new items and are not analysed yet.) Each question in the PQ is a seven-point scale. To get a P score, we simply add all scores from question 1 up to question 18. Item 11 is not scored as suggested in the scoring instruction and item 13, 16, and 17 are reverse scored.

Subject	Presence (P) Score	Immersive Tendencies (IT) Score
1	92	151
2	73	143
3	84	172
4	44	145
5	86	139
6	100	175
7	74	132
8	69	162
9	97	171
10	53	115
11	92	174
12	81	143
13	77	118
14	103	137
15	77	157
<b>Mean</b>	80.13	148.93
<b>Standard Deviation</b>	16.49	19.42

Table 3: The P score and IT score of participants in the control group. All data for the control group is obtained.

Each question in the ITQ is also on a seven-point scale. Similar to the PQ, IT scores are calculated by adding up all questions. There is no weighting of items and there are no

reversed score questions in ITQ. Although Witmer and Singer [97] suggested that it is better to use only analysed items, we use all items to analyse the result since the reliability of the ITQ (the Cronbach's alpha will be discussed in Section 5.6.3) is acceptably high.

### The Control Group

15 data values were collected. The highest P score is 104 while the lowest P score is 44. The highest IT score is 175 and the lowest IT score is 115. The mean of P scores is 80.13 and the standard deviation is 16.49. The mean of IT scores is 148.93 and the standard deviation is 19.42. Table 3 shows the scores for each subject and the mean and standard deviation of P and IT scores.

Subject	Presence (P) Score	Immersive Tendencies (IT) Score
1	72	-
2	100	133
3	87	148
4	83	137
5	104	158
6	93	172
7	74	133
8	101	126
9	93	178
10	93	162
11	120	203
12	90	159
13	76	141
14	78	111
15	-	-
<b>Mean</b>	90.28	150.84
<b>Standard Deviation</b>	13.36	24.64

Table 4: The P score and IT score of each participant in the experimental group. Subject 1 missing the IT score and subject 15 missing both data.

### The Experimental Group

There are 15 subjects using the experimental virtual environment system; however, there are only 14 P scores and 13 IT data points collected. The IT score is missing for one subject and both IT and P scores are missing for another subject. The minimum P score is 72 and the maximum P score is 120. The minimum IT score is 111 and the maximum IT score is 203. The mean value of P scores is 90.28 and the standard deviation is 13.36. The mean value of IT scores is 150.84 and the standard deviation is 24.64. Table 4 shows the scores for each subject in the experimental group.

### 5.6.3 Statistical Analysis

#### Reliability of Presence Questionnaire and Immersive Tendencies Questionnaire

Cronbach's alpha coefficient is used to measure the internal consistency of a questionnaire. The internal consistency of a scale refers to the degree to which the items measure a single factor only. Cronbach's alpha coefficient ranges between zero and one; one indicates perfect agreement between items, and zero no agreement between items. We need to determine whether the questionnaire is reliable before any further analysis can be done.

With 29 observations, we obtain a high Cronbach's alpha value (0.895) for PQ. With 28 data points for the IT score, we obtain a Cronbach's alpha value of 0.814 which is acceptably high.

#### Correlations Between Presence Score and Immersive Tendencies Score

It has been suggested by Witmer and Singer [97] that one who becomes easily involved in some media should have a higher sense of presence while experiencing virtual environments. This means there is a correlation between P scores and IT scores. We have conducted a correlation test on the P scores and the IT scores. It shows that the correlation coefficient of P scores and IT scores is 0.51, which is significant with  $p < 0.05$ . The correlation matrix (n=28) is presented in Table 5.

	P score	IT score
P score	1.00	<b>0.506401</b>
IT score	<b>0.506401</b>	1.00

Table 5: Correlations between P scores and IT scores. Significant correlations ( $p < 0.05$ ) is in bold.

We have performed the correlations between the P scores and the IT scores for the two groups separately. Interestingly, we find that the correlation between the P score and the IT score for the control group is 0.27 which is not significant at  $p < 0.05$ . However, the correlation between P score and IT score for the experimental group is 0.63 which is significant at  $p < 0.05$ . Table 6 (ii) shows the correlation matrices between P score and IT score for two groups.

	P score (Control)	IT score (Control)
P score (Control)	1.00	0.272925
IT score (Control)	0.272925	1.00

(i)

	P score (Experimental)	IT score (Experimental)
P score (Experimental)	1.00	<b>0.634556</b>
IT score (Experimental)	<b>0.634556</b>	1.00

(ii)

Table 6: **Correlation matrix of P and IT scores.** (i) The correlation between P score and IT score for the control group. (ii) Correlation between P score and IT score for the experimental group. The significant level is  $p < 0.05$ , in bold.

### T-test of Presence Score and Immersive Tendencies Score

We performed a t-test on the IT scores between the two groups. The result of the t-test on the IT scores shows that there is no significant difference [ $t = 0.229$ ;  $p > 0.05$ ] between the two groups. Table 7 shows the result of the t-test on the mean of the IT score between the control and the experimental group.

	t-value	df	p	F-ratio variances	p variances
Experimental IT score vs. Control IT score	0.229536	26	0.820248	1.607125	0.394134

Table 7: **Result of t-test on IT score of two groups.** It shows no significant difference ( $p < 0.05$ ) between them.

We also performed a t-test on the means of the P scores between the experimental and the control group. The result shows that there is no significant difference [ $t = 1.813$ ;  $p > 0.05$ ] between two groups. Table 8 shows the result of the t-test.

	t-value	df	p	F-ratio variances	p variances
Experimental P score vs. Control P score	1.81334	27	0.08091	1.823468	0.454902

Table 8: **Result of t-test on P scores of two groups.** It is not significant at  $p < 0.05$ .

## 5.7 Discussion of Results

We have shown that the participants are randomly assigned between two groups since there is no difference in the mean value of Immersive Tendencies Questionnaire of both groups. That means that any difference in presence must be due to the differences in the system, not because of the differences of individuals.

According to the factors suggested by Witmer and Singer [97] and Slater and Steed [79], the experimental VR system (using single-window approach) should give greater presence (because the interaction method used in the system is more natural than that used in the control VR system). Additionally, the experimental VR system reduces the disruption of presence while one interacts and modifies objects. However, the difference in the mean of the presence score shows that the experimental VR system **does not** significantly improve presence. We argue that it might be because the difference in presence is too small to be picked up by the presence questionnaire. Moreover, the time of the break in presence is short; therefore it might not affect the experience of presence greatly. It might be significantly different if the users are forced to change environments which are hugely different, such as experiencing virtual environments using immersive systems and editing using a desktop system.

Although the PQ is one of the best available measures of presence, it is possible to argue that the presence questionnaire is not suitable for this kind of research. A similar study by Johns, Daya, and Sellars [35] had similar results – the environment which should generate a greater sense of presence did not have significant mean difference from the low-presence environment. Furthermore, the presence questionnaire is highly reliant on how the user answers the question. For instance, one of the participants scores with only extreme scores (i.e. 1 and 7) and the middle score (i.e. 4).

Additionally, the questionnaire is dependent on the user's prior experiences [22]. For instance, one question in the Presence Questionnaire is: "How natural did your interactions with the environment seem?" Scoring 1 means extremely artificial and scoring 7 means completely natural. In this case, using a mouse to interact with virtual objects is not natural when comparing the interaction method with real life. However, one might score 1 because he or she is not familiar with computer systems and one might score 7 since he or she knows that using a mouse to interact in a virtual environment is the most commonly used method.

One way to check the validity of the PQ is to test the correlation between the user's immersive tendencies and presence. Theoretically, presence should be correlated with immersive tendencies [97], and the overall presence score and immersive tendencies scores support the theory. Nevertheless we have found that the presence score is significantly correlated with the IT score in the experimental group but not significantly correlated in the control group. The correlations between the IT score and the P score seems to hold only in the high presence virtual environments. A similar result was also found by Johns, Daya, and Sellars [35].

## 5.8 Summary

We conducted an experiment to test whether the tool approach virtual environment can create higher presence than the conventional virtual environment authoring tool. Our new design uses the single-window approach which is more natural than the several-window approach used in the traditional VR authoring tools. In this chapter, we describe the methods, subjects, and results of the experiment.

Our experimental findings:

- The Cronbach's alpha coefficient for Immersive Tendencies Questionnaire is 0.895 and the Cronbach's alpha value of Presence Questionnaire is 0.814. The Cronbach's alpha coefficient of PQ and ITQ are acceptably high. This means that both questionnaires are reliable.
- The t-test on Immersive Tendencies scores between the two groups is not significant with  $t = 0.229$ . It shows that there is no difference in subjects between the two groups; therefore the differences found in the experiment are due to the differences of the systems.

- The correlation between Presence scores and Immersive Tendencies scores for all participants is significant [ $r = 0.51$ ]. This result is consistent with the theory proposed by Witmer and Singer [97] – the easier one can feel involved and immersed in the virtual environment, one can experience higher presence.
- The correlation between Presence scores and Immersive Tendencies scores for the control group is 0.27. However, the correlation between Presence scores and Immersive Tendencies scores for the experimental group is significant [ $r = 0.63$ ]. Our results suggest that the correlation between Presence scores and Immersive Tendencies scores is not consistent across all systems. This result is also found in Johns, Daya, and Sellars's research [35].
- The t-test of Presence scores between the two groups is not significant [ $t = 1.813$ ;  $p > 0.05$ ]. It suggests that there is no significant difference between the single-window approach and the multiple-window approach for virtual environment authoring tools. We argue that this might be because the Presence Questionnaire cannot pick up the small difference in presence. Additionally, we think that the Presence Questionnaire might not be suitable for this type of research and the user's prior experiences might affect the way they answer the questions.

## Chapter 6

# Usability Experiment

In this chapter, we investigate whether there are differences in interface usage behaviour when the interface strategies used in the virtual environments are different. We conduct this experiment using the tool approach virtual environment (b) and the pin approach prototypes (refer to Section 4.3 and 4.4 for details).

We begin this chapter with the aim of this experiment in Section 6.1. The participants' information is described in Section 6.2. Then we will describe the tasks that they have to do during the experiment session in Section 6.3. This is followed by the experimental procedure in Section 6.4 and the equipment and prototypes (Section 6.5) used in this experiment. We then describe the methods we used for measuring usability in Section 6.6. The results of this experiment are presented in Section 6.7 and Section 6.8. We also propose recommendations about the applications suitable for each interaction approach.

### 6.1 Aims

From the presence experiment, we have learned that the single-window virtual environment certainly does not generate less presence than the multiple-window approach found in most virtual environment authoring tools. Furthermore, we believe that a single-window system may increase presence, but in a way that is not detectable by current measures. We shall therefore keep the single environment interface and see if we can improve the usability of such a system.

We, therefore, have built two prototypes; both of them use the single-window approach, but the strategies of managing "modes" are different. We wish to investigate if different ways of interacting with objects will cause users to behave differently. We believe that the users will

change their ways of working, depending on the system they are using. For instance, with the tool approach prototype system, we expect the users to work in the fashion of “tool by tool” and users might work “object by object” in the pin approach prototype. In the experiment, we will test whether our hypothesis holds. Additionally, we will investigate which way of using modes is more useful.

Please note, that as interaction in the conventional environment is identical to the tools environment, there is no value in doing a separate usability comparison between the pin environment and the conventional environment.

## 6.2 Participants

As suggested by Neilson [49], only five users are needed for a formative usability test. In our experiment, there were nine participants involved. These participants were paid volunteers and were students from various faculties in University of Cape Town. We expect the end-users of these prototype systems to be non-experts in computer programming and computer graphics – they are interested in creating and editing their own virtual environments. However, they must be familiar with, and know how to use, standard computer input and output devices.

The participants were all familiar with standard input devices (mouse and keyboard) and they were not familiar with any virtual environment editing software, such as 3D Studio Max. Some of them were studying computer science courses, however, none of them had taken any courses in computer graphics.

The experiment has a *between-groups* design. The participants were divided randomly into two groups and each was assigned to one of the prototypes. Four participants used the tool approach VE (b) and five participants used the pin approach VE (Details of the two prototypes are in Section 4.3 and Section 4.4 respectively). Table 9 shows the information regarding participants.

	Participant	Gender	Courses
<b>Tool Approach Prototype</b>	1	M	Commerce
	2	M	Commerce (Business Science)
	3	M	Science (Engineering)
	4	F	Science
<b>Pin Approach Prototype</b>	1	M	Commerce (Business Science)
	2	M	Science
	3	M	–
	4	M	Science
	5	M	Science

Table 9: **Information about Participants in this Experiment.** The courses information of participant 3 of pin approach prototype group is missing (although we can assume they had not studied computer graphics or 3D modelling as they would not have been allowed to participate in the experiment).

### 6.3 Experimental Tasks

There are two virtual rooms in the two prototypes. One room is a storage room, which contains all the furniture at the beginning of the experiment. The other room is a living room with a lamp inside. The users can walk freely in the virtual environment. However, they can only walk from storage room to living room through the door, and vice versa. Figure 22 shows the layout of the virtual environment.

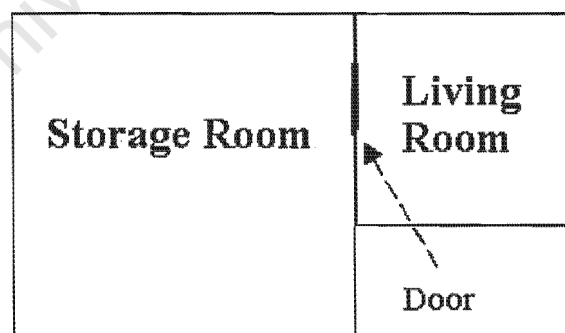


Figure 22: **The top view of the virtual environment.** The left room is the storage room which initially contains furniture and objects. Only a lamp is in the living room.

room at the start of the experiment. The users are required to move all objects from the storage to the living room through the door, which is the thicker line between two rooms indicated by the arrow.

The task is to arrange the virtual living room according to the image in one of the virtual books. The users need to move all the furniture and objects to the other room (the living room), through the door. There are three books in the virtual environment. The books contain the images of three different arrangements of the room. There are three channels on the TV. On each channel, there are four images of the particular arrangements from different viewpoints. The three books are marked differently at the back. One book is marked a “L”, one is marked a “1” and the last one does not have a mark. The users are asked to find the book with “1” at the back, and arrange the room accordingly. The task description for the experiment is in Appendix F.

Figure 23 shows the arrangement of the room which the users need to make. The top view of the room is in the book marked “L”, and different viewpoints can be found on the TV.

We use the living room and storage room scenario as this is a real-world task familiar to users. Additionally, compared to manipulating some boxes, manipulating virtual furniture is more realistic. By placing instructions in books and on the television, we ensure that users are required to perform “use” actions in the environment – if users were given the plan on physical paper, they would not need to “use” the objects and hence comparison of the two interaction techniques would be pointless.

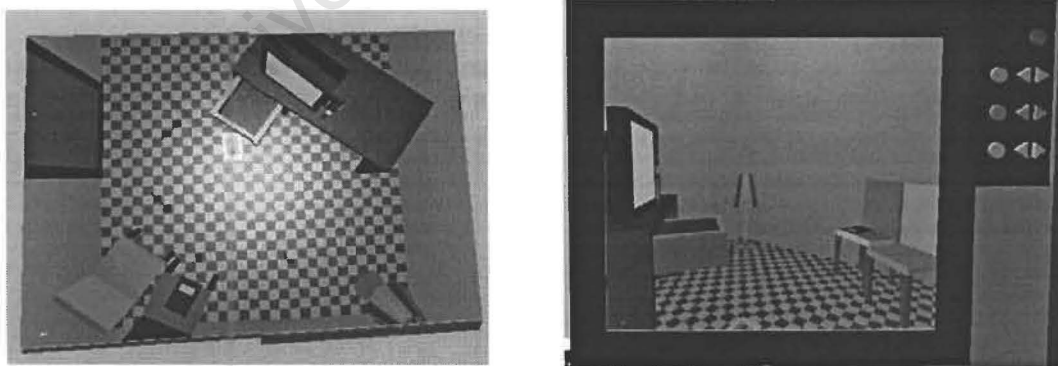


Figure 23: A set of arrangements of the room. The left is the top view image of an arrangement in a book and the right is the corresponding side view in one channel

of the TV. The red button on the TV is the power button and the colour buttons are channel buttons. The arrows next to each button are for changing images within channel.

## 6.4 Experimental Procedure

Each participant was assigned to one session for the experiment. The participant was led to a small room, separate from the laboratory. The participant had no prior knowledge about what we wanted to test or the tasks of the experiment. The participant was shown how to interact in the virtual environment. They were then given the instruction sheet about what they needed to do. The participant used the prototype for about thirty minutes but none of them knew the time limit at the beginning of the experiment. Within the time limits, the participant performed the task at their own pace and they were not limited in the sequence of the tasks. The main reason that we limited the time is that the task is quite complex and the users might need quite a long while to finish everything. We think that thirty minutes should be long enough for us to observe the users' behaviour without them becoming too bored with the task.

The experimenter initially stayed in this room for three to five minutes. During this time, the experimenter helped the participant, if the participant was not sure about the tasks, or if the participant was not able to move an object due to the collision detection between the object and the walls. The experimenter then left the participant alone in the room to complete the task.

## 6.5 Equipment and Prototype Systems Used

Both prototype systems use a desktop virtual environment with standard input devices and standard output devices. The equipment used is:

- Dual Intel Pentium 4 Xeon 1.7 GHz
- 1.2 GBytes of RIMM
- The video card is a Geforce II GTS
- Two 21 inch CRT monitors
- A digital video camera

One monitor is used by the users for performing the task and another monitor is connected to the system for video recording the users' actions.

Both prototype systems used in this experiment were implemented using Java 3D. One prototype system uses the tool approach for editing objects in the virtual environment, while the other uses the pin approach.

## 6.6 Measurement of Usability

We used two ways to measure the usability for both prototype systems. One is a usability questionnaire, and the other is an observation technique. In this section, we will discuss these two techniques for measuring usability.

### 6.6.1 Questionnaire

The subjects answered a questionnaire after they experienced each virtual environment. The questionnaire we used is the Computer System Usability Questionnaire. It is freely available on-line from AMC [56]. The Computer System Usability Questionnaire (CSUQ) was developed by J. R. Lewis of IBM and measures on a 7-point Likert scale. The overall reliability of this questionnaire is very high ( $r = 0.95$ ) [38]. There are nineteen questions, which are based around three themes [38]:

- System usefulness
- Information Quality
- Interface quality

System usefulness refers to the users' perception of ease of use, ease of learning, effectiveness of completing the tasks and subjective feeling about the system. Information quality includes the error handling methods, and the information provided on-line. Lastly, interface quality is about the presentation of information and the input and output devices [38].

Here are some sample questions.

- Overall, I am satisfied with how easy it is to use this system.
- I believe I became productive quickly using this system.

The questionnaire is available in Appendix G.

### 6.6.2 Observation

We intend to find out whether people work differently with different interaction methods in virtual environment authoring tools. To achieve this goal, we observed the users' behaviour by video recording their actions on the screen monitor.

We set up another monitor, which is connected to the computer. The display on this monitor is the same as the monitor that the participant looks at. We record the users' actions on the monitor during the experimental session using a video camera. We analysed the tapes after the experiment to try to find some behavioural pattern or unusual occurrences while the participants were in the virtual environment.

## 6.7 Observation Results

From post experiment video analysis, we have identified some behavioural patterns among the participants within the systems and across systems. In this section, we will describe the behaviour of the participants in the tool approach prototype, the pin approach prototype and then some common behaviour that both groups of subjects exhibited

### Tool Approach Prototype System

With the tool approach prototype, we have found some patterns that the participants do while performing the task. These are summarised as following:

- Of all four participants, one participant did not move all the furniture across the room. The other three managed to move all the furniture and books across the room and painted some objects.
- The users tried all tools on one object at the beginning.
- The users seemed to be familiar with the controls after a short while. At the end, they worked faster than they worked at the beginning of the experiment.
- The participants moved objects to the other room in any order and put the objects in any position. Once all objects were in the living room, they then put the objects in the correct location. The last thing they do was painting.
- Some participants would put the book in the same orientation as the living room. That is, if they are standing in the storage and facing the living room, then they will place

the book vertically and rotate it in such a way that the door is at the left bottom of the book.

- After using the system for a short while, the users understood that tools were applied to all objects. In other words, if they wanted to use the currently used-tool on other object, they did not have to click on the tool again.

### **Pin Approach Prototype System**

With the pin approach prototype, we have identified some patterns of performing the tasks.

- Of all five participants, two of them did not move all the furniture across the room. The other three managed to move all the furniture and books across the room and only one of them did not start painting objects.
- When editing an object, no matter if the object was pinned or not, the participant would pin the object before starting.
- The participants would move objects to the correct position one by one. Once all objects are in the correct position, then they would paint the objects according the image in the book.
- Most participants did not fully understand the function of the drawing pin and would tend to invoke the functionality of the object while the object was pinned. Some participants only took a few mouse clicks to learn how to invoke the functionality of the objects; some took longer (about ten minutes). Eventually all participants learned before the end of the experiment. Once they learned it, they made fewer mistakes, and took fewer tries.

### **Overall Findings**

There are some common behavioural patterns found in both systems.

- Even though the participants were informed that it was possible to walk through objects, they would try to not walk through objects.
- The participants would move objects away from each other if they collided.
- Most participants were confused with moving the object up and down (along Y-axis) and push and pull (along Z-axis). Even with exploring in the virtual environment for a while, some subjects were still confused with these two operations.

- To put objects one on top of the other, the users used all manipulation tools (rotation, movement, and push and pull) to make sure that the objects were lying flat on each other.
- The way of choosing colour – using red, green and blue sliders – is not a good idea. Most subjects took a long time to find the desired colours.

## 6.8 Questionnaire Results

We asked the participants to answer the usability questionnaire (CSUQ). In this section, we will describe the statistical results of the questionnaires. The questionnaire allows users to fill in their comments about the system; we will present the comments in this section as well.

### 6.8.1 Statistical Analysis

#### Gathering Data

We add up all items in the questionnaire and get the total score for each participant. Table 10 contains the total score for each participant and the mean for each group.

There are four data points collected for the tool approach prototype. The mean of the tool approach prototype is 92.5 and the standard deviation is 5.97.

There are four valid data points collected for the pin approach prototype, although there were five participants involved. One is ignored because all items in the questionnaires were scored with seven (the maximum score). The mean for the pin approach prototype is 69.25 and the standard deviation is 9.912. The table below (Table 10) shows the individual scores and the mean and standard deviation for each group. The data collected will be used to perform a t-test.

	<b>Participants</b>	<b>Score</b>
<b>Tool Approach Prototype</b>	1	85
	2	99
	3	91
	4	95
	<b>Mean</b>	92.5
	<b>Std. Dev.</b>	5.97

	Participants	Score
<b>Pin Approach Prototype</b>	1	61
	2	82
	3	69
	4	61
	<b>Mean</b>	69.25
	<b>Std. Dev.</b>	9.912

Table 10: **The data collected from the CSUQ.** The Score of the questionnaire for each participant and the mean and standard deviation for each group.

### T-Test on the Mean of the Tool Approach System and the Pin Approach System

We conducted a t-test on the mean scores of the tool approach system and the pin approach system. The result shows that the t value is 4.191. There is a significant difference between the mean of the tool approach and that of the pin approach at  $p < 0.05$ . Table 11 shows the result of the t-test. This result shows that the tool approach virtual environment is better than the pin approach virtual environment.

	t-value	P	p variances
Tool approach vs. Pin approach	4.191	0.0057	0.4274

Table 11: **The result of t-test.** The p value is 0.0057, which is significantly different at  $p < 0.05$ .

### 6.8.2 Users' Comments

Each subject could enter 5 positive comments and 3 negative comments on the questionnaire. The comments were about the general issues, such as interaction with mouse, and the manipulation in the virtual environments. We have summarized the comments.

#### Positive Comments

- The subjects feel that using the mouse to navigate, and to interact in the virtual environment is easy to understand.

- Collision detection on the walls is good because it let them know if the object is against the walls.
- The manipulation method is intuitive, as if the users were in the real world.
- They have control of the environment.

### Negative Comments

- There is no **Undo** function.
- There is no **Zoom** function.
- Pieces of furniture can pass through one another and people can walk into objects.
- It is difficult to find the correct colours.

## 6.9 Discussions

In this section, we discuss the results found from the observation, as well as the statistical results. The main findings we have are:

- From the statistical results, it seems that the tool approach prototype is better than the pin approach prototype in terms of usability. This is also true from observations since the users are more familiar with the interaction metaphor. There are significant differences between the tool approach prototype and the pin approach prototype.
- We expected that the users work “tool by tool” in the tool approach prototype and “object by object” in the pin approach prototype. However, there are no behavioural differences (as we expected) although there are differences in the results of the questionnaires.
- The participants who used the pin approach prototype would re-pin the objects, even if they could see the pin.
- The participants who used the pin approach prototype more frequently invoked the function of objects than those who used the tool approach prototype.
- In both systems, the users reacted to the virtual environments as if they were in the real world.

### The Tool Approach vs. the Pin Approach

Initially we thought that the pin approach prototype would be more useful than the tool approach prototype. One reason is that the ability to work on different objects with different modes is more efficient because users do not need to change modes constantly. For instance, one can change the mode of a chair to “paint” and the table to “rotation” and not change the mode of a book. In this way, the user can paint the chair, and rotate the table, and refer to the plan in the book without changing modes once the modes are set as desired. Another advantage of the pin approach prototype over the tool approach prototype is that the pin approach is somehow more intuitive for the “use” functionality than the tool approach – the users do not need to use the “hand tool” to use an object in the pin approach.

Our observation of the users’ actions shows that the users are more familiar with the tool approach than the pin approach. The reason could be that in the real world, people work on a tool basis. Also, as global mode is more common in interactive software, and our subjects were all computer literate, this style of interaction may be more intuitive for them. This seemed to be borne out in our observation of re-pinning – users were trying to impose a global mode way of thinking on to a mode per object environment.

### **Difference in Working Style**

We also expected the users to behave differently in the two different prototypes. We thought users would tend to work in a tool-by-tool fashion in the tool approach, and a object-by-object fashion in the pin approach. The reason we assumed this is that it would be the most efficient way of working with the different interfaces.

Interestingly, we did not observe this in the participants. The only pattern we found is that the participants would move objects across the room in any order, whether to the correct location or not, and then paint the objects to the correct colours. This happened regardless of environment type.

One explanation for this is that the experiment session was quite short; only about thirty minutes. This may not be enough time to change the way of working. Additionally, the participants may need longer to discover an efficient way of working with the interface in the virtual environment. We do not suggest that if the users had to work with one of the prototypes for a while, then they would develop a new way of working; we just suggest that it is not justified to say that the pin approach is not an appropriate interface metaphor.

### **Re-pin objects**

As mentioned above, the participants would pin an object before editing, even when the object was already pinned. This way of working could require less cognitive effort – the participants do not need to remember the active mode of the object. Instead, they simply re-pin the object and select the relevant tools. Moreover, the action “pin and then edit” is not thought of as changing modes, rather a procedure for editing objects in the virtual environment.

We do, however, provide some visual feedback when the participants use different tools on the objects to lessen cognitive load. This visual feedback includes the change of the shape of the mouse cursor and the texture of the in-use-pin. These help the users to recall the modes for each object. We argue that the participants were not aware of the change of the shape of the mouse cursor and the texture on the in-used-pin because the virtual environment itself is a rich environment. There are too many interesting things happening in the virtual environment for the user to notice. As a result, the changes become so minor that the participants hardly notice.

#### **Invoking the functions of objects**

As discussed in Section 4.4 and above, it is easier to invoke the functionality of the object in the pin approach prototype than in the tool approach prototype. This was supported by the users' behaviour. The participants using the pin approach more frequently invoked the functionality of the objects compared to the tool approach (in which the users only invoke when necessary). Thus, the participants who experience the pin approach virtual environment tend to explore more than those who use the tool approach virtual environment.

#### **Reacting in Virtual Environments as in Reality**

In both systems, the subjects avoid walking into objects and when one object collided into another, they moved one away from the other. Our virtual environments are not photo-realistic and some physical laws, such as gravity, and collisions between users and objects are ignored. Nevertheless, the participants attempt to model the virtual environment as the real world, and obey the laws of reality without prompting.

### **6.10 Recommendations**

From the findings and discussion, we will suggest some applications that are suitable for the global-mode approach and some applications that are suitable for the pin approach.

### **The Tool Approach Virtual Environment Authoring Tools**

In the real world, people work in a way similar to the tool approach. Though the overload of accessing tools may not be less than for the pin approach virtual environment, we believe that the tool approach is easier to learn than the pin approach. This is important for novice users – they do not have to learn new interactive methods and they can get used to the interface in a short time. Additionally, in most editing environments the tool approach is employed, and therefore it has the higher familiarity than the pin approach prototype.

### **The Pin Approach Virtual Environment Authoring Tools**

There is a default function for each object (refer to Section 4.4 for more details). From the discussion above, the pin approach virtual environment authoring tools encourage the users to explore the environment and use these functions. Therefore, this approach is suitable for those applications that require the users to explore the environment or search for information hidden in virtual objects, such as in educational applications.

This approach is also suitable for collaborative virtual environments. The drawing pin can serve as a lock – while the drawing pin is on an object, others can see that the object is currently being edited. As the texture of the pin would change depending on the tools applied on that object, others in the environment can also know which operation is used on that object as well. These applications include architecture building, or design discussion meetings.

## **6.11 Summary**

We have discovered how people might behave in virtual environment authoring tools with different interaction methods. The findings and some explanations are:

- The tool approach is better than the pin approach in usability. This is because the tool approach is a “global mode” approach, which is similar to the way people work in the real life and in other software.
- People do not adjust their behaviour for efficient performance. The experiment session was quite short and it is not necessary for a person to change their way of working in a short period. Moreover, people may still not find an efficient way of working with a certain user interface.

- People would re-pin an object even if it was not necessary. We argue that it is an easier way of working – people do not need to remember the active modes for each object.
- People do not seem to notice iconic visual feedback in virtual environments as readily as in desktop environments perhaps due to the rich visual nature of the virtual environment – it is hard to notice minor changes.
- Subjects who use the pin approach prototype invoke the function of objects more frequently than those who use the tool approach prototype.
- Even though the objects in the virtual environment are not realistic and there is lack of physical laws, people behave and react as if in reality.

We also give some suggestions about the suitable applications for each interaction approach.

- The tool approach is better for novice users and infrequent users. The method is similar to our everyday experience and therefore the users can learn it easily.
- The pin approach is better in applications where users are encouraged to interact and explore and are only required to undertake moderate amounts of editing or customisation. The pin approach is also suitable for collaborative virtual environments, in which people can build virtual objects or a virtual world together.

We also performed statistical analysis on the questionnaires. We found that the t value for the t-test is 4.191. This shows that there is significant difference between the mean of the tool approach and the mean of the pin approach. It shows that the usability of the tool approach system is significantly better than that of the pin approach system.

## Chapter 7

# Conclusion

The focus of this research is to build a virtual environment which allows users to create and edit a virtual world within the virtual world itself.

Besides this, the interaction methods used in the virtual environments should be easy to use, easy to learn, and as natural as possible. Therefore the interface should become transparent, thereby improving the usability of the virtual environments. Another issue that we are concerned with is the sense of presence in virtual environment authoring tools. The typical virtual environment authoring tool that we are interested in are those which allow a wide range of users (including novices) to use them. This kind of VR authoring tool, such as Alice, is not only an authoring tool; they are also VR applications, in which one can experience virtual environments. Many VR researchers believe that it is important to maintain the sense of presence in such VR systems and our virtual environments should generate that sense of presence.

### 7.1 Delivered Virtual Environments

To achieve our goals, we implemented four virtual environments using three different interaction methods. One interaction method is the conventional approach – using several different windows for different interaction purposes. “Mention” and “use” are thus explicitly distinguished and pre-defined by the system. The second interaction method is using the “tool approach” in which we use virtual tools, such as paintbrush and magic wand, to edit and interact with objects. This approach adopts a mode per system for solving the problem of “mention and use”. There is only *one* environment, and therefore the users can edit objects within the same virtual world. The last interaction metaphor is the “drawing pin”. A drawing pin is used to indicate the mode of the particular object. This metaphor allows the users to edit

some part of the virtual environment while some other part of it can be used as usual. This metaphor uses the mode per object approach to deal with “mention and use”. Similar to the tool approach virtual environment, there is one environment which is used for editing and for experiencing the virtual environment.

We argue that the conventional approach of having different edit and use environments disturbs the experience of virtual environments. The system forces the users to change between different windows and hence the sense of presence in the virtual environment cannot be maintained – but it may be the case that users maintain presence in the wider application. Furthermore the method of using several environments is not natural since in real life people do not change to another world for interacting and back for viewing the world.

In addition, the novel interaction metaphors are borrowed from real life experience. They are hopefully, therefore, easy to learn and easy to use.

## **7.2 Experiments for this Research**

We conducted two experiments. One tests whether our novel interaction metaphor can generate greater sense of presence than that of the conventional virtual environments. The other experiment tests whether there is any difference in usability between the two approaches – the mode per system and the mode per object approaches.

### **7.2.1 Results Obtained from the Presence Experiment**

In this experiment, we are interested in presence particularly. The conventional approach (which is the multiple-window and the modal approach) virtual environment authoring tool asks the users explicitly to distinguish between design and viewing environments. We believe that the sense of presence is important in this kind of virtual environment authoring tool because these tools combine both features of VR authoring tools and VR applications. Our novel approach, the tool approach, is used in this experiment. The tool approach virtual environment uses only one environment for design and viewing. Additionally, the interaction metaphor is borrowed from real life. Therefore it should generate greater presence because of the reduction of breaks in presence and the naturalness of the interaction method.

From the experiment, we have found that the mean presence score of the tool approach virtual environment is greater than that of the conventional approach virtual environment. However,

there is no statistically significant difference between them. This means that both virtual environments can generate more or less the same level of presence.

According to the presence literature, the tool approach virtual environment should generate greater presence than the conventional approach virtual environment. Our finding, however, shows that the tool approach virtual environment **does not** generate significantly greater presence than the conventional virtual environment. We argue that it might be because the duration of breaks in presence are short and the differences between the design and experience environments are subtle. Additionally, the presence questionnaire might not be suitable for this kind of research because it is highly reliant on how the user answers the questions (one of the subjects scored with only extreme scores and the middle score). Moreover, the questionnaire is dependent on the users' prior experience and no user has had previous experience with our system.

Although the result is not expected and it does not agree with the literature, it is an important finding – the tool approach virtual environment does not generate less presence than the conventional virtual environment. This means that in this kind of virtual environment, the conventional approach is not the only solution. The tool approach virtual environment can be as good as the conventional approach virtual environment in terms of presence. Additionally, we believe that the tool approach should be better than the conventional approach in terms of usability since the tool approach solves some usability problems which are found in tools like Alice.

### **7.2.2 Results Obtained from the Usability Experiment**

This experiment aimed to investigate the usability of our virtual environments and the users' behaviour with the different interaction metaphors in the virtual environment. It also investigated which solution of the "mention and use" problem is better in virtual environments. The two virtual environments in question are the tool approach virtual environment and the pin approach virtual environment – there is no point in comparing the multiple windows environment as it is interactionally equivalent to the tool approach. We suggested, prior to the experiment, that people might work in a tool-by-tool fashion with the tool approach virtual environment – that is, for instance, painting all objects to the desired colours and then moving all objects to the correct locations. In the pin approach virtual environment, people might work in a object-by-object fashion. This seems to be the most efficient way of performing tasks. Moreover the pin approach virtual environment allows the users to use virtual objects more easily and reduces the number of times the mode is changed. The pin approach virtual environment, therefore, should be easier to use and to learn.

From the observations, we found that there were no major behaviour differences in how the participants performed the tasks. People work in a similar pattern in both virtual environments – they move the furniture first and then they will paint the objects as the last task. First of all, the results are not expected – the users do not behave differently with different interaction methods. One explanation is that the experiment session was short (about 30 minutes). This may not have been enough time to change their way of working. The participants may need a longer time to discover an efficient way of working with the interface in the virtual environment.

From the questionnaires, we found that the tool approach virtual environment has higher usability scores than the pin approach virtual environment. This result is also shown from the observation that three out of four subjects in the tool approach virtual environment achieved painting, whereas only two out of five subjects achieved painting in the pin approach virtual environment. One possible reason for this is because the tool approach interaction method is more familiar than the pin approach interaction method. In real life, people work on a tool basis. Furthermore, the “global mode” is more common in interactive software and our subjects were all computer literate, and therefore this style of interaction may be more intuitive for them.

Furthermore, we also find from the users’ actions that people use a “less effort” way of working when they encounter a new system. In the pin approach virtual environment, the users would pin an object before editing, even when the object was already pinned. This way of working could require less effort because the participants do not need to remember the active mode, and whether the object is pinned or not. Moreover, the action of “pin and then edit” is thought of as a procedure of editing objects rather than an action for changing modes.

Although the results of this experiment are not what we expected, it gives important suggestions for designing interaction methods in virtual environments – when choosing an interaction method in virtual environments, the best method is the one which is consistent with real life experience or which is a familiar method. Additionally, because people do not bother to learn new ways of working with a new system at the beginning, the natural interaction method helps the users to master the system. This agrees with HCI and usability designing principles. Furthermore, people might build their own model of how the things work in the system when they encounter a novel interaction method. This helps the users to understand and to learn the system, and the usability of the system is higher because the interface becomes transparent.

### 7.3 Recommendations

We have suggested some uses for the different interaction metaphors. The tool approach virtual environment is easy to learn and easy to use. Therefore the tool approach virtual environment is useful for novice users. Users are familiar with tools in real life and adopting this strategy in virtual environment reduces the time needed to learn the interface.

Another suggestion is for the pin approach virtual environment. Because the drawing pin is visible and the appropriate texture is shown on the knob of the drawing pin, this approach can be used in collaborative virtual environments. The example applications are shared spaces for the conference meetings which require interaction with the design. Additionally, to use objects in the pin approach virtual environment is trivial so the pin approach virtual environment encourages the users to explore the environment. This approach, therefore, can also be used in applications intended for the users to explore and learn about the environments.

In this research, we have used two different methods of user testing – using the questionnaire technique and using the observation technique. With the questionnaire technique, we can obtain the suggestions and opinions directly. However, this method is highly dependent on how the users answer the questions and their understanding and interpretation of the questions. For instance, in the presence experiment, one of the subjects answered the questions with 1, 4, and 7 (the extreme scores and the middle scores), and in the usability experiment, one of the participants scored all usability questions with 7 (highly agree). The observation technique, on the other hand, does not get the information of how the users feel about the system directly. It gives experimenters insight into how the users work with the system and some facts, such as if the users have problems with certain features of the system, cannot be hidden, even though the users report differently from what they really think of the system. However, observation is costly. For similar work, we suggest using both techniques because with two techniques, the researchers can get more information about how the system performs, how the users feel, and how they work in the system.

### 7.4 Future Work

Our prototypes of virtual environment authoring tools can do only a few tasks compared to available VR authoring tools. Our prototypes include only some simple tasks such as moving and painting objects. A future version should be extended to allow users to add new objects, to add animation and to add events into the virtual environments. The system can also include

the ability to create a new object (using existing software, such as 3D Studio Max) into the virtual environments.

Secondly, the presence experiment did not show a significant difference between the multiple-window approach and the single-window approach. It would be necessary to repeat the presence experiment with larger data set and using the improved virtual environment authoring tools. This improved virtual environment authoring tools would allow the users to create new objects, and to specify events and animation. The systems would be immersed for experiencing the virtual world and a desktop system for editing and creating the world. We could design the tasks to be more complex, not only re-arrange the furniture in the virtual rooms. For example, we could ask users to create furniture for their virtual place. With the new design of the experiment, we expect to find that the single-window approach does improve presence and it generates more enjoyment for the users when designing their own virtual places as users are forced to switch from edit to use environments more frequently.

As suggested in Chapter 6, the pin approach virtual environment can be used in collaborative virtual environments and for exploring virtual environments. A future study could test whether this approach is suitable in collaborative virtual environments – in our experience, no other VE authoring application provides this type of collaborative authoring interface. We would use observation techniques, and usability questionnaires. We would observe how the users work in the virtual environments, and what they feel about the method. Because the pin approach is drawn from real life experience, it should be easy to learn and understand the metaphor. We are therefore expecting to find that our method requires less time to learn and it is easy to master and collaborate in the virtual environments. Furthermore, we would attempt to increase the effectiveness of the pin by using different shaped pins to represent different editing operations. This would be computationally more expensive than the current solution of differing textures, but may better help users remember which editing task is associated with which object.

We have also suggested that the pin approach encourages the users to explore the virtual environments. Therefore, this approach can be used for educational purposes. Another future application could be to use the pin approach virtual environment as an educational tool. We would test how well people learn with this kind of environment – does this method help people to learn, or does this method distract from learning?

## Appendix A

# Presence Questionnaire

PRESENCE QUESTIONNAIRE (Modified version)  
(Witmer & Singer, Vs. 3.0, Nov. 1994)

Characterize your experience in the environment, by marking an "X" in the appropriate box of the 7-point scale, in accordance with the question content and descriptive labels. Please consider the entire scale when making your responses, as the intermediate levels may apply. Answer the questions independently in the order that they appear. Do not skip questions or return to a previous question to change your answer.

### WITH REGARD TO THE EXPERIENCED ENVIRONMENT

1. How much were you able to control events?  
*1 Not at All... 4 Somewhat... 7 Completely*
2. How responsive was the environment to actions that you initiated (or performed)?  
*1 Not responsive... 4 Moderately responsive... 7 Completely responsive*
3. How natural did your interactions with the environment seem?  
*1 Extremely artificial... 4 Borderline... 7 Completely natural*
4. How much did the visual aspects of the environment involve you?  
*1 Not at all... 4 Somewhat... 7 Completely*
5. How compelling was your sense of objects moving through space?  
*1 Not at all... 4 Moderately compelling... 7 Very compelling*

6. How much did your experiences in the virtual environment seem consistent with your real world experiences?

*1 Not consistent... 4 Moderately consistent... 7 Very consistent*

7. Were you able to anticipate what would happen next in response to the actions that you performed?

*1 Not at All... 4 Somewhat... 7 Completely*

8. How completely were you able to actively survey or search the environment using vision?

*1 Not at All... 4 Somewhat... 7 Completely*

9. How closely were you able to examine objects?

*1 Not at all... 4 Pretty closely... 7 Very closely*

10. How well could you examine objects from multiple viewpoints?

*1 Not at all... 4 Somewhat... 7 Extensively*

11. How well could you move or manipulate objects in the virtual environment?

*1 Not at all... 4 Somewhat... 7 Extensively*

12. How involved were you in the virtual environment experience?

*1 Not involved... 4 Mildly involved... 7 Completely engrossed*

13. How much delay did you experience between your actions and expected outcomes?

*1 No delays... 4 Moderate delays... 7 Long delays*

14. How quickly did you adjust to the virtual environment experience?

*1 Not at all... 4 Slowly... 7 Less than one minute*

15. How proficient in moving and interacting with the virtual environment did you feel at the end of the experience?

*1 Not proficient... 4 Reasonable proficient... 7 Very proficient*

16. How much did the visual display quality interfere or distract you from performing assigned tasks or required activities?

*1 Not at all... 4 Interfered somewhat... 7 Prevented task performance*

17. How much did the control devices interfere with the performance of assigned tasks or with other activities?

*1 Not at all... 4 Interfered somewhat... 7 Interfered greatly*

18. How well could you concentrate on the assigned tasks or required activities rather than on the mechanisms used to perform those tasks or activities?

*1 Not at all... 4 Somewhat... 7 Completely*

19. How completely were your senses engaged in this experience?

*1 Not engaged... 4 Mildly engaged... 7 Completely engaged*

20. To what extent did events occurring outside the virtual environment distract from your experience in the virtual environment?

*1 Not at all... 4 Moderately... 7 Very much*

21. Overall, how much did you focus on using the display and control devices instead of the virtual experience and experimental tasks?

*1 Not at all... 4 Somewhat... 7 Very much*

22. Were you involved in the experimental task to the extent that you lost track of time?

*1 Not at all... 4 Somewhat... 7 Completely*

23. Were there moments during the virtual environment experience when you felt completely focused on the task or environment?

*1 None... 4 Occasionally... 7 Frequently*

24. How easily did you adjust to the control devices used to interact with the virtual environment?

*1 Difficult... 4 Moderate... 7 Easily*

## Appendix B

# Immersive Tendencies Questionnaire

### IMMERSIVE TENDENCIES QUESTIONNAIRE

(Witmer & Singer, Version 3.01, September 1996)

Indicate your preferred answer by marking an "X" in the appropriate box of the seven point scale. Please consider the entire scale when making your responses, as the intermediate levels may apply. For example, if your response is once or twice, the second box from the left should be marked. If your response is many times but not extremely often, then the sixth (or second box from the right) should be marked.

1. Do you easily become deeply involved in movies or tv dramas?  
*1 Never... 4 Occasionally... 7 Often*
2. Do you ever become so involved in a television program or book that people have problems getting your attention?  
*1 Never... 4 Occasionally... 7 Often*
3. How mentally alert do you feel at the present time?  
*1 Not Alert... 4 Moderately... 7 Fully Alert*
4. Do you ever become so involved in a movie that you are not aware of things happening around you?  
*1 Never... 4 Occasionally... 7 Often*
5. How frequently do you find yourself closely identifying with the characters in a story line?  
*1 Never... 4 Occasionally... 7 Often*

6. Do you ever become so involved in a video game that it is as if you are inside the game rather than moving a joystick and watching the screen?

*1 Never... 4 Occasionally... 7 Often*

7. What kind of books do you read most frequently? (CIRCLE ONE ITEM ONLY!)

*Spy novels; Fantasies; Science fiction; Adventure novels; Romance novels;  
Historical novels; Westerns; Mysteries; Other fiction; Biographies; Autobiographies;  
Other non-fiction*

8. How physically fit do you feel today?

*1 Not fit... 4 Moderately Fit... 7 Extremely fit*

9. How good are you at blocking out external distractions when you are involved in something?

*1 Not very good... 4 Somewhat good... 7 Very good*

10. When watching sports, do you ever become so involved in the game that you react as if you were one of the players?

*1 Never... 4 Occasionally... 7 Often*

11. Do you ever become so involved in a daydream that you are not aware of things happening around you?

*1 Never... 4 Occasionally... 7 Often*

12. Do you ever have dreams that are so real that you feel disoriented when you awake?

*1 Never... 4 Occasionally... 7 Often*

13. When playing sports, do you become so involved in the game that you lose track of time?

*1 Never... 4 Occasionally... 7 Often*

14. How well do you concentrate on enjoyable activities?

*1 Not at all... 4 Moderately well... 7 Very well*

15. How often do you play arcade or video games? (OFTEN should be taken to mean every day or every two days, on average.)

*1 Never... 4 Occasionally... 7 Often*

16. Have you ever gotten excited during a chase or fight scene on TV or in the movies?

*1 Never... 4 Occasionally... 7 Often*

17. Have you ever gotten scared by something happening on a TV show or in a movie?  
*1 Never... 4 Occasionally... 7 Often*
18. Have you ever remained apprehensive or fearful long after watching a scary movie?  
*1 Never... 4 Occasionally... 7 Often*
19. Do you ever become so involved in doing something that you lose all track of time?  
*1 Never... 4 Occasionally... 7 Often*
20. On average, how many books do you read for enjoyment in a month?  
*1 None... 4 Three... 7 More*
21. Do you ever get involved in projects or tasks, to the exclusion of other activities?  
*1 Never... 4 Occasionally... 7 Often*
22. How easily can you switch attention from the activity in which you are currently involved to a new and completely different activity?  
*1 Not so easily... 4 Fairly easily... 7 Quite easily*
23. How often do you try new restaurants or new foods when presented with the opportunity?  
*1 Never... 4 Occasionally... 7 Frequently*
24. How frequently do you volunteer to serve on committees, planning groups, or other civic or social groups?  
*1 Never... 4 Occasionally... 7 Frequently*
25. How often do you try new things or seek out new experiences?  
*1 Never... 4 Occasionally... 7 Often*
26. Given the opportunity, would you travel to a country with a different culture and a different language?  
*1 Never... 4 Maybe... 7 Absolutely*
27. Do you go on carnival rides or participate in other leisure activities (horse back riding, bungee jumping, snow skiing, water sports) for the excitement of thrills that they provide?  
*1 Never... 4 Occasionally... 7 Often*
28. How well do you concentrate on disagreeable tasks?  
*1 Not at all... 4 Moderately well... 7 Very well*

29. How often do you play games on computers?

*1 Not at all... 4 Occasionally... 7 Frequently*

30. How many different video, computer, or arcade games have you become reasonably good at playing?

*1 None... 4 Three... 7 Six or more*

31. Have you ever felt completely caught up in an experience, aware of everything going on and completely open to all of it?

*1 Never... 4 Occasionally... 7 Frequently*

32. Have you ever felt completely focused on something, so wrapped up in that one activity that nothing could distract you?

*1 Not at all... 4 Occasionally... 7 Frequently*

33. How frequently do you get emotionally involved (angry, sad, or happy) in news stories that you see, read, or hear?

*1 Never... 4 Occasionally... 7 Often*

34. Are you easily distracted when involved in an activity or working on a task?

*1 Never... 4 Occasionally... 7 Often*

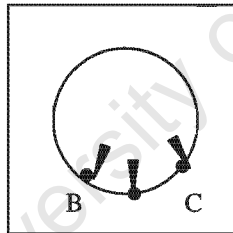
## Appendix C

# User Manual of Control Group for Presence Experiment

There are two modes in this system, viewing mode and editing mode.

### Viewing Mode:

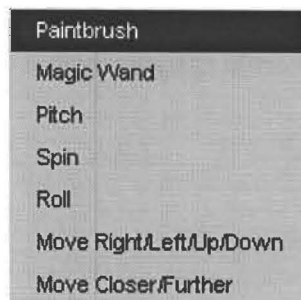
This is the default mode. In this mode, a left-mouse click allows you to move in the room in a predefined route.



A left-mouse click and drag rightward will move you from A to C. A left-mouse click and drag leftward will move you from A to B. The arrow indicates the view that you will see.

A right-mouse click on any object in the scene will pop up a menu. No other actions are available.

## APPENDIX C. USER MANUAL OF CONTROL GROUP FOR PRESENCE 95 EXPERIMENT



There are seven (7) options in the menu that you can choose. Selecting any of these options will change the appropriate editing mode.

### **Editing Mode:**

Another window with only the select object will pop up for editing. Any changes on the object will change on the viewing window only when the editing window is closed.

### **Paintbrush**

This allows you to change the colour of the selected object.

### Choosing options

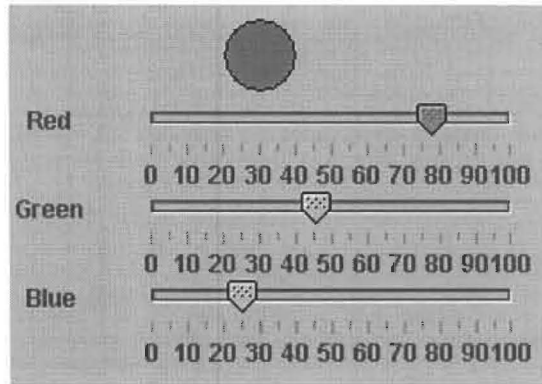
There are 4 operations that you can do in this sub-mode: paint, rotation 1, rotation 2, and rotation 3 (see above for more details). The chosen one is indicated by a rectangle around the text. To choose other option, simply click on either the text, or the image.



### **Paint:**

Three sliders (red, green, and blue) are shown on the right hand side of the editing window. These three colours are the basic components of all colours. The number under each slider is the percentage of that component used.

APPENDIX C. USER MANUAL OF CONTROL GROUP FOR PRESENCE 96 EXPERIMENT



For example, the colour shown in the circle above the sliders (which is brown) is blended with 77% of red, 46% of green and 25% of blue.

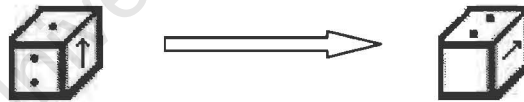
However, when you close the editing window, only the change in colour will result in the viewing window.

**Pitch** 

To trigger this action, you need to click on the object and hold. Then if the mouse moves downward, the box will be rotated in this way:



and if the mouse moves upward, the box will be rotated as



**Spin** 

To trigger this action, you need to click on the object and hold. Then if the mouse moves right, the box will be rotated in this way:



and if the mouse moves upward, the box will be rotated as

APPENDIX C. USER MANUAL OF CONTROL GROUP FOR PRESENCE 97  
EXPERIMENT



**Roll** 

To trigger this action, click the mouse on one object and hold. **No matter which direction the mouse moves, the box will rotate in the following way:**



**Magic Wand** 

This allows you to change the colour of the whole object with one mouse click. See Paint section for how to use different options in this sub-mode and how to select colour.

**Pitch** 

Pitch is the only tool that you can use (see paint section for details), and any changes in position will result in the viewing window.

**Spin** 

Spin is the only tool that you can use (see paint section for details), and any changes in position will result in the viewing window.

**Roll** 

Roll is the only tool that you can use (see paint section for more details), and any changes in position will result in the viewing window.

**Move Right/Left/Up/Down** 

This allows you to move the object up/down and right/left in the scene. To trigger this action, the mouse is clicked on an object and holds. If the mouse is dragged left, the object will move left and the same for right, up and down.

## APPENDIX C. USER MANUAL OF CONTROL GROUP FOR PRESENCE 98 EXPERIMENT

### **Move Closer/Further**

This allows you to move the object closer to you or further to you. While the mouse is clicked on an object and dragged downward, the object will move closer to you (it will become smaller), and upward, it will move further to you (it will become bigger).

### **NOTE:**

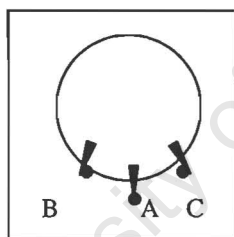
In this scene, walls and the floor cannot be manipulated at all. And the floor is not able to change colour. Therefore the menu for room will only consist of paintbrush and magic wand. The gravity is not simulated and objects can collide into each other.

University of Cape Town

## Appendix D

# User Manual of Experimental Group for Presence Experiment

It is possible to change view of the scene. To do so, a left-mouse click allows you to move in the room in a predefined route.

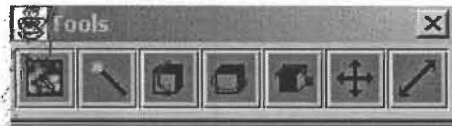


A left-mouse click and drag rightward will move you from A to C. A left-mouse click and drag leftward will move you from A to B. The arrow indicates the view that you will see.

A toolbox is provided for editing any objects in the scene. By editing, it means both manipulation (move, roll, etc) and change colours. There are two ways of changing colours – using paintbrush and using magic wand – and there are several manipulation tools – pitch, spin, roll, move up/down/left/right, and move closer/further.

To start editing, all you have to do is click on the toolbox (which is situated at the left bottom corner of the screen). A floating menu will appear with all editing options. Simply click on the desired option to select.

## APPENDIX D. USER MANUAL OF EXPERIMENTAL GROUP FOR PRESENCE 100 EXPERIMENT

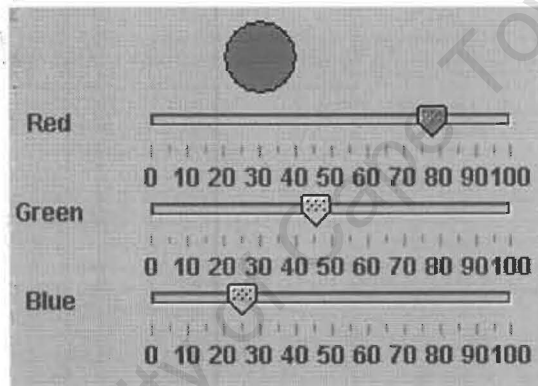


However, while the list of tools in the toolbox is shown (the floating menu is shown), you are not allowed to move in the room. In order to change view, you need to close the menu – simply by click on the X at the right top corner of the floating menu.

The following is the details of how to use all these tools.

### Paintbrush :

Paint brush allows you to change the colour of the selected object.



Three sliders (red, green, and blue) are shown on the right hand side of the editing window. These three colours are the basic components of all colours. The number under each slider is the percentage of that component used. For example, the colour shown in the circle above the sliders (which is brown) is blended with 77% of red, 46% of green and 25% of blue.

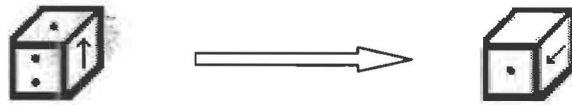
### Magic wand :

Similar to paint brush, however, magic wand changes the colour of the whole object with one mouse click.

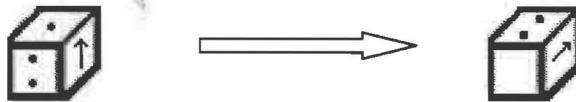
### Pitch :

To trigger this action, you need to click on the object and hold. Then if the mouse moves downward, the box will be rotated in this way:

APPENDIX D. USER MANUAL OF EXPERIMENTAL GROUP FOR PRESENCE 101  
EXPERIMENT

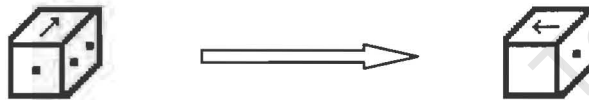


and if the mouse moves upward, the box will be rotated as



**Spin** 

To trigger this action, you need to click on the object and hold. Then if the mouse moves right, the box will be rotated in this way:



and if the mouse moves upward, the box will be rotated as



**Roll** 

To trigger this action, the click the mouse on one object and hold. **No matter which direction the mouse moves, the box will rotate in the following way:**



**Move** 

This allows you to move the object up/down and right/left in the scene. To trigger this action, the mouse is clicked on an object and holds. If the mouse is dragged left, the object will move left and the same for right, up and down.

## APPENDIX D. USER MANUAL OF EXPERIMENTAL GROUP FOR PRESENCE 102 EXPERIMENT

### **Move closer/further** :

This allows you to move the object closer to you or further to you. While the mouse is clicked on an object and dragged downward, the object will move closer to you (it will become smaller), and upward, it will move further to you (it will become bigger).

### **NOTE:**

In this scene, walls and the floor cannot be manipulated at all. And the floor is not able to change colour. Therefore the menu for room will only consist of paintbrush and magic wand. The gravity is not simulated and objects can collide into each other.

University of Cape Town

## Appendix E

# Tasks for Presence Experiment

1. The table is a weird colour. Please change it to purple.
2. Draw a red line on the left-hand-side wall and a green line on the other (back) wall. Move yourself so that you can see the other two walls. Then draw a blue line on one wall and a yellow line on the other.
3. Move yourself back to a position that you can see all furniture in the room.
4. Move the table to the right bottom corner of the room.
5. Move the lamp closer to you (about 2 tiles).
6. One side of the lamp is red. Rotate the lamp so the red side is facing you.
7. Move all books onto the table and place them on end. Then paint the side of any two books that you can see to ruby, and milk-white.
8. Change the colour of the faces of the lamp cover to pink, purple, brown, and light green respectively.
9. Place the lamp in such a way that both brown and purple face could be seen.
10. Change the chair on the left to be your favourite colour.
11. Put a green blob on the wall behind the chair (which is facing you).
12. Put a green blob on the wall behind the lamp.
13. Change the holder of the lamp to dark blue.
14. Put a red blob on the wall behind the chair which you coloured with your favourite colour.
15. Draw an orange 'L' on each wall.
16. Change the colour of the table to light blue but make the colour of legs in black (remember there are 4 sides for each leg).
17. Draw a white rectangle on the table. After you finish, place the table in upright position.
18. Move the chair with your favorite colour next to the other chair.

19. Put one book on a chair (either one), one on the table, and one on floor under the table. Place them flatly.
20. Change the colour of the book on the table to gray.
21. Change the colour of the side of the book (the one on a chair) to gold. (white part to gold).
22. Draw a white dot on the gray book.
23. Draw two yellow lines on the book which is on the floor.

University of Cape Town

## Appendix F

# Experimental Tasks for the Usability

## Experiment

**Objective:** To arrange the living room to the designed setting.

**Objects:**

- There are one desk, two chairs, three books and a TV set placed in the storage room.
- A lamp is in the living room.

**Things to Note:**

- The light in the living will be automatically switched off once you step out of the room.
- You cannot move objects into walls, or the floor. **NOTE: THIS DETECTION DOES NOT APPLY TO STORAGE ROOM.** Be careful not to move objects outside the walls of the storage room.
- Gravity is not simulated in this virtual environment – objects can float in the air.

**Instruction:**

Your task is to arrange the room according to the setting in the book with a “1” at the back. Different viewpoints of the setting can be found in the furniture.

## Appendix G

# Computer System Usability Questionnaire

Based on: Lewis, J. R. (1995) *IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use*. *International Journal of Human-Computer Interaction*, 7(1): 57-78.

Please rate the usability of virtual environment.

- Try to respond to all the items.
- For items that are not applicable, use: N/A

1. Overall, I am satisfied with how easy it is to use this system.

Disagree 1 2 3 4 5 6 7 Agree

2. It was simple to use this system.

Disagree 1 2 3 4 5 6 7 Agree

3. I can effectively complete my work using this system.

Disagree 1 2 3 4 5 6 7 Agree

4. I am able to complete my work quickly using this system.

Disagree 1 2 3 4 5 6 7 Agree

5. I am able to efficiently complete my work using this system.

Disagree 1 2 3 4 5 6 7 Agree

6. I feel comfortable using this system.

Disagree 1 2 3 4 5 6 7 Agree

7. It was easy to learn to use this system.

Disagree 1 2 3 4 5 6 7 Agree

8. I believe I became productive quickly using this system.

- Disagree 1 2 3 4 5 6 7 Agree
9. The system gives error messages that clearly tell me how to fix problems.  
 Disagree 1 2 3 4 5 6 7 Agree
10. Whenever I make a mistake using the system, I recover easily and quickly.  
 Disagree 1 2 3 4 5 6 7 Agree
11. The information (such as online help, on-screen messages, and other documentation) provided with this system is clear.  
 Disagree 1 2 3 4 5 6 7 Agree
12. It is easy to find the information I needed.  
 Disagree 1 2 3 4 5 6 7 Agree
13. The information provided for the system is easy to understand.  
 Disagree 1 2 3 4 5 6 7 Agree
14. The information is effective in helping me complete the tasks and scenarios.  
 Disagree 1 2 3 4 5 6 7 Agree
15. The organization of information on the system screens is clear.  
 Disagree 1 2 3 4 5 6 7 Agree
16. The interface of this system is pleasant.  
 Disagree 1 2 3 4 5 6 7 Agree
17. I like using the interface of this system.  
 Disagree 1 2 3 4 5 6 7 Agree
18. This system has all the functions and capabilities I expect it to have.  
 Disagree 1 2 3 4 5 6 7 Agree
19. Overall, I am satisfied with this system.  
 Disagree 1 2 3 4 5 6 7 Agree

List the most **negative** aspect(s):

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_

List the most **positive** aspect(s):

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_
4. \_\_\_\_\_
5. \_\_\_\_\_

# Bibliography

1. 3ds max. Discreet. <http://www.discreet.com/products/3dsmax>. (last visited June 2002)
2. Adobe Photoshop. <http://www.adobe.com/products/photoshop/main.html>. (last visited June 2002)
3. Aesthetic Solutions VR/VRML Newsletter Archive November 1997, 1, November 1997. URL: [http://www.aesthetic.com/newsletter/newsletter\\_nov97.html](http://www.aesthetic.com/newsletter/newsletter_nov97.html). (last visited June 2002)
4. Alice Home Page. URL: <http://www.alice.org>. (last visited June 2002)
5. Autodesk – AutoCAD. <http://www3.autodesk.com/adsk/section/0,,616714-123112,00.html> (last visited June 2002)
6. Bangay, S., and L. Preston. An Investigation into Factors Influencing Immersion in Interactive Virtual Environments. In G. Riva and K. Wiederhold and E. Molinari editors, *Virtual Environments in Clinical Psychology and Neuroscience*. Ios Press, 1998.
7. Beier, K. P. UM-VRL: Virtual Reality; A Short Introduction. URL: <http://www-vrl.umich.edu/intro/index.html>.
8. Bowman, D. A. Interaction Techniques for Immersive Virtual Environments: Design, Evaluation, and Application. *Human-Computer Interaction Consortium (HCIC)*, 1998.
9. Bowman, D. A., and L. F. Hodges. User Interface Constraints for Immersive Virtual Environment Applications. Technical Report TR95-26, Graphics, Visualization, and Usability Centre, College of Computing, Georgia Institute of Technology, 1995.
10. Bowman, D. A., and C. A. Wingrave. Design and Evaluation of Menu Systems for Immersive Virtual Environments. In *Proceedings of IEEE Virtual Reality*, pages 149-156. 2001.
11. Brooks, R. P. Jr. What's Real About Virtual Reality? *IEEE Computer Graphics and Applications*, 19(6): 16-27, 1999.
12. Bryson, S. Interaction of Objects in a Virtual Environment: a two point paradigm. Technical Report RNR-91-009. NASA Ames Research Center, 1991.

13. Conway, M. J. Alice: Easy-to-Learn 3D Script for Novices. Dissertation: University of Virginia, 1998.
14. Dalgarno, B. and J. Scott. Motion Control in Virtual Environments: A Comparative Study. In V. Paelke, and S. Volbracht (editors), *Workshop on Usability Centred Design and Evaluation of Virtual 3D Environments*, 2000, University of Paderborn, Germany.
15. Deol, K. K., A. Sutcliffe, and N. Maiden. A design advice tool presenting usability guidance for virtual environments. In *User Centered Design and Implementation of Virtual Environments Workshop. Special Issue of International Journal of Human-Computer Studies*, 1999.
16. The DIVE Home Page. <http://www.sics.se/dive/> (last visited June 2002)
17. Dix, A., J. Finlay, G. Abowd, and R. Geale. *Human-Computer Interaction*, chapter 11, pages 406 – 442. Prentice Hall Europe, second edition, 1998.
18. Ellise, S. R. Presence of Mind: A Reaction to Thomas Sheridan's "Further Musings on the Psychophysics of Presence". *Presence: Teleoperators and Virtual Environments*, 5(2): 247-259, 1996.
19. EON Reality, Inc. <http://www.eonreality.com>. (last visited June 2002)
20. Eysenk, M. W., and Keane, M.T. (1995). *Cognitive Psychology: A Student's Handbook*. Hove, U.K.: Psychology Press.
21. Fencott, P. C. Presence and the Content of Virtual Environments. *Proceedings of the Second International Workshop on Presence*, 1999. URL: <http://web.onyxnet.co.uk/Fencott-onyxnet.co.uk/pres99/pres99.htm>. (last visited June 2002)
22. Freeman, J., S. E. Avons, D. E. Pearson, and W. A. IJsselsteijn. Effects of Sensory Information and Prior Experience on Direct Subjective Ratings of Presence. *Presence: Teleoperators and Virtual Environments*, 8(1) 1 – 13, 1999.
23. Gaver, W. W. Technology Affordances. In *Conference on Human Factors and Computing System*, pages 79 – 84, 1991.
24. Genesis3D. <http://www.genesis3d.com>. (last visited June 2002)
25. Heeter, C. The Look and Feel of Direct Manipulation. *HYPERNEXUS: Journal of Hypermedia and Multimedia Studies*, Summer, 1991.
26. Heeter, C. Being There: The Subjective Experience of Presence. *Presence: Teleoperators and Virtual Environments*, 1(2): 262-271, 1992.
27. Held, R. M., and N. I. Durlach. Telepresence. *Presence: Teleoperators and Virtual Environments*, 1(1): 109-112, 1992.

28. Herndon K. P., A. van Dam, and M. Cleicher. Workshop on the Challenges of 3D Interaction. *SIGCHI Bulletin*, 26(4), October 1994.
29. Houde, S. Interactive Design of an Interface for Easy 3-D Direct Manipulation. In Conference Proceedings on Human Factors in Computing Systems, pages 135 – 142, 1992.
30. Houde, S. and Sellman, R. In Search of Design Principles for Programming Environments. *CHI'94 Proceedings*, 1994 pp230.
31. Howell, D.C. *Statistical Methods for Psychology*. Pacific Grove, CA: Duxbury, 2002.
32. IJsselsteijn, W. A., and H. de Ridder. Measuring Temporal Variations in Presence. *Presence in Shared Virtual Environments Workshop*, 1998.
33. IJsselsteijn, W. A., H. Ridder, J. Freeman, and S. E. Avons. Presence: Concept, determinations and measurement. *Proceedings of the SPIE, Human Vision and Electronic Imaging V*, 2000. URL: <http://www.ip0.tue.nl/homepages/wiussels/SPIE-HVEI-2000.pdf>. (last visited June 2002)
34. Isdale, J. What Is Virtual Reality? A Web-Based Introduction Version 4. URL: <http://vr.isdale.com/WhatIsVR/frames/WhatIsVR4.1.html>. (last visited June 2002)
35. Johns, C., M. Daya, and D. Sellars. The Effects of Presence on Small Group Collaborations and Interactions in Collaborative Virtual Environment. Honours Project, Department of Computer Science, University of Cape Town, 1999.
36. Kalawsky, R. S. JTPA Project 305, Human Factor Aspects of Virtual Design Environments in Education: Project Report. Advanced VR Research Centre, Loughborough University. February, 2000.
37. Kaur, K., A. Sutcliffe, and N. Maiden. Improving Interaction with Virtual Environments. *IEE Colloquium on "The 3D interface for the information worker"*, 1998.
38. Keinonen, T. One-dimensional usability – influence of usability on consumers' product preference. UIAH publication A21. University of Art and Design Helsinki, 1998
39. Kooper, R., B. Wills, K. Hamilton, D. Allison, and L. F. Hodges. COOL-VR: a Virtual Environments Toolkit. URL: <http://citeseer.nj.nec/370959.html>. (last visited June 2002)
40. Lindeman, R. W., J. L. Sibert, and J. K. Hahn. Towards Usable VR: An Empirical Study of User Interfaces for Immersive Virtual Environments. *Proceedings of CHI*, pages 64-71, 1999. URL: [citeseer.nj.nec.com/26648.html](http://citeseer.nj.nec.com/26648.html). (last visited June 2002)
41. Lombard, M. Resources for the Study of Presence. URL: <http://www.presence-research.org>. (last visited June 2002)

42. Lombard, M., and Ditton, T. At the Heart of It All: The Concept of Presence. *Journal of Computer Mediated Communication*, 3(2), 1997. URL: <http://www.ascuse.org/jcmc/vol3/issue2/lombard.html>. (last visited June 2002)
43. Lombard, M., and T. B. Ditton. Measuring Presence: A Literature-based Approach to the Development of a Standardized Paper-and-pencil Instrument. *The Third International Workshop on Presence*, 2000. URL: <http://www.presence-research.org>. (last visited June 2002)
44. Lynch, P. J. Visual Design for the User Interface Part 1: Design Fundamentals. *Journal of Biocommunications*, 21(1), pages 22-30, 1994.
45. Mania, K., and A. Chalmers. Between Real and Unreal: Investigating Presence and Task Performance. The Design of Pilot Study. *Proceedings of the Second International Workshop on Presence*, 1999. URL: <http://www.cs.bris.ac.uk/~mania/presence-workshop/submission.htm>, (last visited June 2002)
46. Marsden, G. Overcoming Design and Execute Modes in User Interface Design Environments. *HCI 95 people and Computers*, pages 133-137, 1995.
47. Mine, M. R. Virtual Environment Interaction Techniques. Technical Report TR95-018, Department of Computer Science University of North Carolina, 1995.
48. Mine, M. R., F. P. Brooks Jr., and C. H. Sequin. Moving Objects in Space: Exploiting Proprioception In Virtual-Environment Interaction. *Proceeding of ACM SIGGRAPH*, pages 19-26, Los Angeles, California, USA, August, 1997.
49. Nielson, J. Why You Only Need to Test With 5 Users. *useit.com: Jakob Neilson's Website*, 19, March, 2000. <http://www.useit.com/alertbox/> (last visited June 2002)
50. Norman, D. A. *The Psychology of Everyday Things*. BasicBooks, 1988.
51. Nunez, D. Virtual Reality Course: Presence and Environmental Perception. Department of Computer Science, University of Cape Town. April, 2001.
52. Paint, Window 2000. <http://www.microsoft.com/windows2000>. (last visited June 2002)
53. Pavlovic, V., R. Sharma, and T. S. Huang. Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7): 677-695, 1997.
54. Pausch, R., T. Burnette, A. C. Capehart, M. Conway, D. Cosgrove, R. DeLine, J. Durbin, R. Gossweiler, S. Koga, and J. White. A Brief Architectural Overview of Alice, a Rapid Prototyping System for Virtual Reality. *IEEE Computer Graphics and Applications*, 1995.

55. Pausch R., and C. Forlines. Web3D RoundUp: Looking Backwards and Forwards. Alice: Model, Paint & Animate - Easy-to Use Interactive Graphics for the Web. *ACM SIGGRAPH Computer Graphics Newsletter*, 34(2), May 2000.
56. Perlman, G. Web-Based User Interface Evaluation With Questionnaires. <http://www.acm.org/~perlman>. (last visited June 2002)
57. Pierce, J. S. Alice in a Squeak Wonderland. In M. Guzdial, and K. Rose editors, *Squeak: Open Personal Computing and Multimedia*, Prentice Hall, 2001.
58. Pierce, J. S., A. Forsberg, M. J. Conway, S. Hong, R. Zeleznik, and M. R. Mine. Image Plane Interaction Techniques in 3D Immersive Environments. In Symposium on Interactive 3D Graphics, pages 39-43, 1997.
59. Pierce, J. S., S. Audia, T. Burnette, K. Christiansen, D. Cosgrove, M. Conway, K. Hinckley, K. Monkaitis, J. Patten, J. Shochet, D. Staack, B. Stearns, C. Sturgill, G. Williams, and R. Pausch. Alice: Easy to Use Interactive 3D Graphics. *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST97)*, pages 77-78, 1997
60. Pierce, J. S., M. Conway, M. van Dantzich, and G. Robertson. Toolspaces And Glances: Storing, Accessing And Retrieving Objects In 3D Desktop Applications. *Symposium on Interactive 3D Graphics*, pages 163-168, 1999.
61. Pierce, J. S., B. C. Stearns, and R. Pausch. Two Handed Manipulation of Voodoo Dolls in Virtual Environments. *Symposium on Interactive 3D Graphics*, pages 141-145, 1999.
62. Poupyrev, I., M. Billinghurst, S. Weghorst, and T. Ichikawa. The Go-Go Interaction Techniques: Non-linear Mapping for Direct Manipulation in VR. In *Proceedings of ACM Symposium on User Interface Software and Technology (UIST '96)*, pages 79-80, 1996.
63. Python. <http://www.python.org>. (last visited June 2002)
64. The Quake. <http://www.quake.com>. (last visited June 2002)
65. Rice, J. A. *Mathematical Statistics and Data Analysis*. Second Edition. Wadsworth Inc, 1995.
66. Schloerb, D. W. A Quantitative Measure of Telepresence. *Presence: Teleoperators and Virtual Environments*, 4(1): 64-80, 1995.
67. Schubert, T., F. Friedmann, and H. Regenbrecht. Embodied Presence in Virtual Environments. In Ray Paton and Irene Neilson, editors, *Visual Representations and Interpretations*, pages 268-278. London: Springer-Verlag, 1999.

68. Schubert, T., F. Friedmann, and H. Regenbrecht. Decomposing the sense of presence: Factor Analytic Insights. *Proceedings of the Second International Workshop on Presence*, 1999.
69. Schuemie, M. J., and C. A. P. G. van der Mast. Presence: Interacting in VR? *Proceedings Twente Workshop on Language Technology*, pages 213-217, 1999.
70. Sheridan, T. B. Defining Our Terms. *Presence: Teleoperators and Virtual Environments*, 1(2): 272-274, 1992.
71. Sheridan, T. B. Musings on Telepresence and Virtual Presence. *Presence: Teleoperators and Virtual Environments*, 1(1): 120-125, 1992.
72. Sheridan, T. B. Further Musings on the Psychophysics of Presence. *Presence: Teleoperators and Virtual Environments*, 5(2): 241-246, 1996.
73. Short, J., E. Williams, and B. Christie. *The Social Psychology of Telecommunications*. Wiley, London, 1976.
74. Slater, M., M. Usoh, and A. Steed. Depth of Presence in Virtual Environments. *Presence: Teleoperators and Virtual Environments*, 3(2): 130-144, 1994.
75. Slater, M., M. Usoh, and A. Steed. Taking Steps: The Influence of a Walking Technique on Presence in Virtual Reality. *ACM Transactions on Computer-Human Interaction*, 2(3): 201-219, 1995.
76. Slater, M., V. Linakis, M. Usoh, and R. Kooper. Immersion, Presence, and Performance in Virtual Environments: An Experiment with Tri-Dimensional Chess. *ACM Virtual Reality Software and Technology (VRST)*, pages 163-172, 1996.
77. Slater, M., and S. Wilbur. A Framework for Immersive Virtual Environments (FIVE): Speculations on the Role of Presence in Virtual Environments. *Presence: Teleoperators and Virtual Environments*, 6(6): 603-616, 1997.
78. Slater, M. Measuring Presence: A Response to the Witmer and Singer Presence Questionnaire. *Presence: Teleoperators and Virtual Environments*, 8(5): 560-566, 1999.
79. Slater, M., and A. Steed. A Virtual Presence Counter. *Presence: Teleoperators and Virtual Environments*, 9(5): 413-434, October 2000.
80. Smith, E. C., C. Irby, R. Kimball, B. Verplank and E. Harslem. Designing the Star User Interface. In *BYTE* 7:242-282, 1982.
81. Smith, R. B., D. Ungar, and B Chang. Chapter 5: The Use-Mention Perspective on Programming for the Interface. In *Languages for Developing User Interfaces*, pages 79-89, USA: Jones and Bartlett Publishers, Inc. 1992.

82. Smith, S. P., and D. J. Duke. Binding Virtual Environments to Toolkit Capabilities. *EUROGRAPHICS 2000*, 18(3): C81-C89, 2000.
83. Smith, S., T. Marsh, D. Duke, and P. Wright. Drowning in Immersion. *Proceedings of UK-VRSIG 98. UK Virtual Reality Special Interest Group*, 1998. URL: <http://www.dcs.ex.ac.uk/ukvrsig98/paplist1.htm>. (last visited June 2002)
84. Stanney, K. M., R. R. Mourant, and R. S. Kennedy. Human Factors Issues in Virtual Environments: A Review of the Literature. *Presence: Teleoperators and Virtual Environments*, 7(4): 327-351, 1998.
85. Stapleton, L. J., and P. J. Costello. *A Survey of Virtual Reality Research in the UK-1997*. Advisory Group on Computer Graphics.
86. Steuer, J. Defining Virtual Reality: Dimensions Determining Telepresence. *Journal of Communications*, 42(4): 73-93, 1992.
87. Stoakley, R., M. J. Conway, and R. Pausch. Virtual reality on a WIM: Interactive worlds in miniature. *Conference Proceedings on Human Factors in Computing Systems*, pages 265-272, 1995.
88. Sutherland, I.E. The Ultimate Display. In *Proceeding of IFIP Congress*, pages 506-508, 1965.
89. Thie, S., and J. van Wijk. A General Theory on Presence. *BT Presence Workshop*, 1998.
90. Tromp, J., A. Bullock, A. Steed, A. Sadagic, M. Slater, and E. Frecon. Small Group Behavior Experiments in the Coven Project. *IEEE Computer Graphics and Applications*, pages 53-63, 1999.
91. Usuh, M., E. Catena, S. Arman, and M. Slater. Using Presence Questionnaires in Reality. *Presence: Teleoperators and Virtual Environments*, 9(5): 497-503, 2000.
92. Weiss, P., and A. S. Jessel. Virtual Reality Applications to Work. *Work*, 11(3): 277-293, 1998.
93. Welch, R. B., T. T. Blackmon, A. Liu, B. A. Mellers, and L. W. Stark. The Effects of Pictorial Realism, Delay of Visual Feedback, and Observer Interactivity on the Subjective Sense of Presence. *Presence: Teleoperators and Virtual Environments*, 5(3), 1996.
94. Welch, R. B. How Can We Determine if the Sense of Presence Affects Task Performance? *Presence: Teleoperators and Virtual Environments*, 8(5): 574-577, 1999.
95. Whitelock, D., and A. Jelfs. Understanding the Role of Presence in Virtual Learning Environments.

96. Willans, J. S. and M. D. Harrison. A toolset supported approach for designing and testing virtual environment interaction techniques. *International Journal of Human-Computer Studies*, 55(2):145-165, 2001.
97. Witmer, B. G., and M. J. Singer. Measureing Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoperators and Virtual Environments*, 7(3): 225-240, 1998.
98. Wloka, M. M. Interacting with Virtual Reality. In J. Rix, S. Haas, and J. Teixeira, editors, *Virtual Environments and Product Development Processes*, Chapman & Hall, 1995.
99. A Survey of Virtual Reality Research in the UK. URL: <http://www.agocg.ac.uk/reports/virtual/38/survey97.htm> (last visited June 2002)
100. Zeltzer, D. Autonomy, Interaction, and Presence. *Presence: Teleoperators and Virtual Environments*, 1(1): 127-132, 1992.