Development of a Virtual Reality Milling Machine For Knowledge Learning and Skill Training

Khalid I. Bakkar

Dublin City University

M Eng. **1999**

Development of a Virtual Reality Milling Machine For Knowledge Learning and Skill Training

By

Khalid I. Bakkar, BSc (Eng.)

This thesis is submitted to Dublin City University as the fulfilment of the requirement for the award of Degree of

Master of Engineering

Supervisor: Professor M.S.J. Hashmi

Dublin City University

School of Mechanical & Manufacturing Engineering

September 1999

Dedication

This thesis is dedicated to my Mum and Dad who always wished me to be a good scientist. Sorry, Daddy you are not with us, but I am carrying your spirit all the way to make your dreams come true.

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Master of Engineering is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: Khalid   ID No: 95971670

Khalid I.Bakkar

Date: September 1999

Acknowledgements

Many individuals have come to my assistance during the present work. I offer many thanks to all, and in particular I would like to acknowledge the contributions of:

Prof. M.S.J.Hashmi, my academic supervisor and Head of School of Mechanical & Manufacturing Engineering at Dublin City University (D.C.U) for his continued support, guidance, limitless energy, encouragement, equanimity, and his excellent supervision throughout the course of this project. Like this, kind of a project cannot be carried out smoothly and successfully without his limitless support.

Special thanks are owed to Dr. Majid S. Hamouda, Department of Mechanical & Manufacturing Engineering, University Putra Malaysia (U.P.M). Without his assistance, the quality of this project would be substantially less.

I would also like to thank Mr. Martin Johnson and Mr. Liam Domican for their co-operation and assistance.
Special thanks will go to Ms. Michelle Considine for her help and providing me with all I requested and more.

I would like to express my sincere thanks to my brother Adel and his wife for their continued support, energy and enthusiastic encouragement.

Special thanks are due to wonderful Irish woman to Mrs. Joan McCarthy who fostered and cared for me in Dublin as if she was my own mother. I will be forever indebted to Jahn for her Irish generosity. I owe this person a lot.

All my fellow students and friends within Dublin City University and elsewhere, for their assistance, support and friendship. Special thanks to Brian Cahill, Joseph Stokes, Helen Murphy, Iqbal, Susan, Mohamed El-hemrey Nasser Khamkham, Abdul baset Abuazza, Hussam Al-shek, Wong, Tang, Dr. Gareth O'Donnell, Dr Mahiuddin Ahmed, Dr. Fawzi, Dr. Abdul Arhman, and finally to Dr. Aziz Elniehum.

There are many, individuals who have contributed in major and minor ways to this work. Thanks are due to them, whose names I forget to mention.

I would like to convey my sincere thanks to my family, especially my mom and my elder brother El-haj Saad, without whose support and encouragement this thesis would not have been written. Thanks are due to my siblings for their encouragement and up-lifting confidence in my ability and who continually inspired me from the family.

My wife, Safa, whom the present work, is dedicated, for providing me with a platform for success and so much more. Thanks a lot.

Abstract

Development of a Virtual Reality Milling Machine For Knowledge Learning and Skill Training

By

Khalid I. Bakkar, B.Sc. (Eng.)

Current methods of training personnel on high cost machine tools involve the use of both classroom and hands on practical training. The practical training required the operation of costly equipment and the trainee has to be under close personnel supervision. The main aim of this project is to reduce the amount of practical training and its inherent cost, time, danger, personal injury risk and material requirements by utilising a virtual reality technology.

In this study, an investigation into the use of Virtual reality for training operators and students to use the Milling Machine was carried out. The investigation has been divided into two sections: first the development of Milling Machine in the 3D virtual environment, where the real machine was re-constructed in the virtual space.

This has been carried out by creating objects and assembling them together. The complete Milling machine was then properly modelled and rendered so it could be viewed from all viewpoints.

The second section was to add motion to the virtual world. The machine was made of functions as for the real machine. This was achieved by attaching Superscape Control Language (SCL) to the objects. The developed Milling machine allows the users to choose the material, speed and feed rate. Upon activation, the virtual machine will be simulated to carry out the machining process and instantaneous data on the machined part can be generated.

The results were satisfactory, the Milling Machine was modelled successfully and the machine was able to perform according to task set. Using the developed Virtual Model, the ability for training students and operators to use the Milling Machine has been achieved.

Nomenclature

2-D	Two Dimensional
3-D	Three Dimensional
3 D S Max	3-D Studio Max
ASCII	American Standard Character for Information Interchange
AVR	Advanced Virtual Reality
BVR	Basic Virtual Reality
CAD	Computer Aided Design
CAI	Computer Aided Instruction
CAM	Computer Aided Machining
CSG	Computer Solid Geometry
DCU	Dublin City University
DoF	Degree of Freedom
DOS	Disk Operating System
DXF	Database Exchange Files
EVR	Entry Virtual Reality
GUI	Graphics User Interface
HMD	Head Mounted Display
HRTF	Head Related Transfer Function
HTML	Hyper Text Mark-up Language
HUD	Heads-Up
Hz	Hertz
IVR	Immersion Virtual Reality
LCD	Liquid Crystal Display

MB	M Byte
NASA	National Aeronautics & Space Administration
PC	Personal Computer
RAM	Read Access Memory
SCL	Superscape Control Language
S.E	Shape Editor
SGI	Silicon Graphic
SVGA	Super Video Graphic Adapter
WE	World Environment
WoW	World on Window
VE	Virtual Environment
VGA	Video Graphic Adapter
VR	Virtual Reality
VMM	Vertical Milling Machine
VEOMTL	Virtual Equipment Operator and Maintenance Training Lab

List of Figures

Chapter Number ***Page Number***

Chapter Four

Chapter Five

List of Tables

Table of Contents

Chapter Five: - The Development System, Result and Discussion

Chapter Six: -Conclusion and Suggestion for Further Work

Appendix-A

Appendix-B

Appendix-C

Appendix-D

Appendix-E

Appendix-F

Appendix-G

Chapter One:
Introduction and Justification

1.1 Introduction

In this introductory chapter, I justify the undertaking of the work, identify the aims of the study and outline the method of approach adopted in achieving the set objective. Finally, a summary of the content of the different chapters is provided under the heading " Layout of Thesis".

1.2 Importance Of Applying Information Technology In Engineering Education And Training

Recent advances in the computational and communication capabilities of computers and their associated infrastructure offer great promise for supporting continual improvements in all aspects of undergraduate education and industrial training. These advances also underscore the need for credible research into the practical benefits and limitations of teaching and learning in settings, enhanced by information technology. Information technology tools can enable students to visualise and experiment with complex, real-world scientific problems, promoting exploration and inquiry based modes of learning. They also enable collaboration, interactive learning, and new pedagogical approaches that change the way in which students and faculty interact. One of the promising Information Technology tools is the Virtual Reality.

Since its development in the late 1950's, Virtual Reality (**VR**) has been a useful tool in various fields to simulate different situations or environments that are not achievable by humans in reality or too dangerous to be achieved in real life.

Virtual Reality is becoming increasingly important as a means of representing physical objects. It has had a considerable amount of success in fields such as entertainment, medicine and military training. This includes the simulation of the eruption of volcanoes, the explosion of nuclear bombs, the military strength of one country against another, the training of new military officers or new pilots and other applications. Of course, VR has also been widely applied in various interesting and adventurous games, which was not made possible until the introduction of VR.

Industry is now beginning to see that VR has potential to become a powerful technology for developing interactive training tools. " *Educational systems are going through a difficult period, not only of financial type, and they need continuous content upgrading and improvements in the learning methodologies exploiting the growing power of electronic tools* " [1].

VR technology may offer a reduction in investment in technical schools, where various types of machinery may be represented in a virtual environment and students may learn about a particular machine by using a VR machine. VR machines have the advantage of not requiring the actual machines associated operating costs.

The advantages of developing VR-based learning systems are:

- A reduction in training time can be obtained on physical machines with associated cost savings.

- There will be an increase in the safety level during training as participants cannot be injured on the virtual machine and will be aware of dangers before being introduced to the actual machine.
- Knowledge transfer through self-learning will benefit the learner and company. It will minimise the damage to equipment from inexperienced operators.
- Capital and maintenance costs of providing actual equipment are eliminated.
- A reduction in travel and time to conduct training can be achieved.

These advantages were among the motivating factors behind the present work.

1.3 Aim Of The Study

Current methods of training personnel on high cost machine tools involve the use of both classroom and hands-on practical training. The practical training required the operation of costly equipment and the trainee has to be under close personnel supervision. The main aim of this project is to reduce the amount of practical training and its inherent cost, time, danger, personal injury risk and material requirements by utilising a virtual reality technology.

The objectives of the present study can be summarised as follows:

✓ To develop a training application that will acts as a framework for other virtual machine and will have the capability of mimicking the operating characteristics of actual machines.

✓ To provide an environment for the trainees to be familiar with the theories of the mechanics of machining before entering the actual laboratory,

- ✓ To reduce the familiarisation time needed by the students and trainee,
- ✓ To generate some typical data of the milling machine so that the students can be familiar with the data variation and thus be able to carry out the tests more efficiently in real life.

The Bridgeport Vertical Milling Machine at the Central Workshop, School of Mechanical and Manufacturing Engineering, Dublin City University, has been modelled in the virtual world for future use by the students and trainees.

To achieve the above-stated objectives, the following method of approach was adopted.

1.4 Method of Approach

Background research about virtual reality has been carried out extensively before the software was available. Most of the resources such as books and periodicals about this subject were obtained from the library and the Internet. The functions and operating sequences of the Milling Machine were examined at the initial stage. The appropriate dimensions of the machine were taken for reconstructing the same Milling Machine in the virtual world. Computer Aided Drafting (*CAD*) of the Milling Machine was performed using 3D Studio Max (*3DS-Max*) and the files are saved as *DXF* format. These files were then converted to Virtual Reality Software "Superscape version 5.6".

This actual design of the Virtual world is divided into two main categories:

(i) Building of Bridgeport Milling Machine in virtual world, and

(ii) Developing the Superscape Control Language (*SCL*) for the objects so that these could react and interact with the end-user and other objects in the virtual world.

1.5 Layout of Thesis

This thesis is divided into six chapters. Following this introductory chapter, Chapter 2 gives the importance of using Information Technology in general and Virtual Reality in particular in the area of education and training. The historical developments of virtual reality systems are reviewed and the relevant literatures are given. Also in this chapter, the virtual reality main components are presented and discussed. Chapter 3 describes the Development of a Virtual Milling Machine for Training and educational purposes. Chapter 4 identifies and describes the software and hardware chosen to develop the most realistic presentation of Milling Machine for a training application. The development stages of the Virtual Milling Machine are given in chapter four. Typical results are also given in Chapter 5. Finally, in Chapter 6 conclusions are drawn based on the present work.

Chapter Two: Virtual Reality Literature Review

2.1 Introduction

The aim of this chapter is to provide the fundamental and background information necessary to pursue the subject of virtual reality. Accordingly, the historical development of VR is first given; this followed by main components to construct Virtual World and the fundamental applications of virtual reality in engineering. The main techniques available and the hardware and software aspects are also reviewed and discussed. No attempt is made to deal in depth with any specific applications involving virtual reality. However, numerous references are given for the readers to explore these topics in greater details.

2.2 The Historical Perspective

Like most technologies, Virtual Reality (*VR*) did not suddenly appear. It emerged in the public domain after a considerable period of research and development in industrial, military and academic laboratories. The emergence of VR was closely related to the maturity of other technologies such as real-time computer system, computer graphics, displays, fibre optics and 3-D tracking. Then, at some critical time a crude working system appeared. It is not unusual for these embryonic systems to be of no practical use to anyone apart from their inventors, but they do establish the idea and are a vital stage of creative design. They also provide a base from which to extrapolate the concept into the future, and by keeping an open mind, a variety of paths the technology can follow may be predicted. Predicting the future with any precision, however, is fraught with problems, no matter how qualified and familiar with technology one is. History reminds us that there is a time and place for everything to happen, and, irrespective of how useful, exciting and revolutionary a product appears to be, it will not succeed unless the right conditions exists [1]

VR obviously depended upon the existence of the computer, not to mention the associated field of computer graphics. The latter relies heavily upon television technology, and both the technologies of television and computers would not exist without electricity. So it is interesting to trace back in time, to see how each major technological breakthrough brings us one step towards today's VR systems, as shown Table [2.1].

Table 2.1 The path, which Virtual Reality has followed [2]

1920's	Edwin Link worked on vehicle simulation, arguably the forerunner of Virtual Reality technology.
1940's	Teleoperation technology began. Teleoperation allows a human operator to use a visual display and a " master " manipulator (e.g. a joystick) to manually control a remote " slave" device such as a vehicle or robotic arm
1950's	" Cinerama " was developed using 3-side screens. Cinerama uses 3 Projectors showing three films to fill an immense deeply curved screen. The deeply curved screen created an intense participitory effect for the audience.
1966	Flight Simulation, *NASA*. Present flight simulation techniques attempt to provide a pilot with enough sensory cues to sufficiently fool the pilot into believing that an actual aircraft is being flown.
late 1960's	Development of synthetic computer-generated displays used for virtual environments, pioneered by Dr. Ivan Sutherland.
Mid 1970's	Krueger introduced the term " Artificial Reality ", which is one of the earlier terms of Virtual Reality.
1984	William Gibson published the term " Cyberspace " in his book, " Neuromancer ". The term Cyberspace was later refined to Virtual Reality.
1989	Jaron Lanier, founder of VPL research, introduced the term " Virtual Reality "
1990	Continued research for specific uses of Virtual Reality, such as the entertainment industry e.g. Sega & Nintendo companies.

Presently there are VR applications in the fields of education, entertainment, engineering and even, the medical profession, to name but a few. In the 1970's, Hollywood started to realise the power of VR in the film industry due to its potential to create extraordinary visual scenarios. Films such as " Star Wars ", followed by 'Terminator' and 'Jurassic Park' are just some of the films that benefited immensely from VR and computer graphics in general. Figure 2.1 shows the wide range of disciplines in which VR is being applied to. Presently the technology is available to create applications in many fields the designer's imagination should take of this technology. The potential of using VR in the training sector is enormous as can be seen from the early days of the NASA flight simulator to recent advancements of using VR to train surgeons in the medical profession.

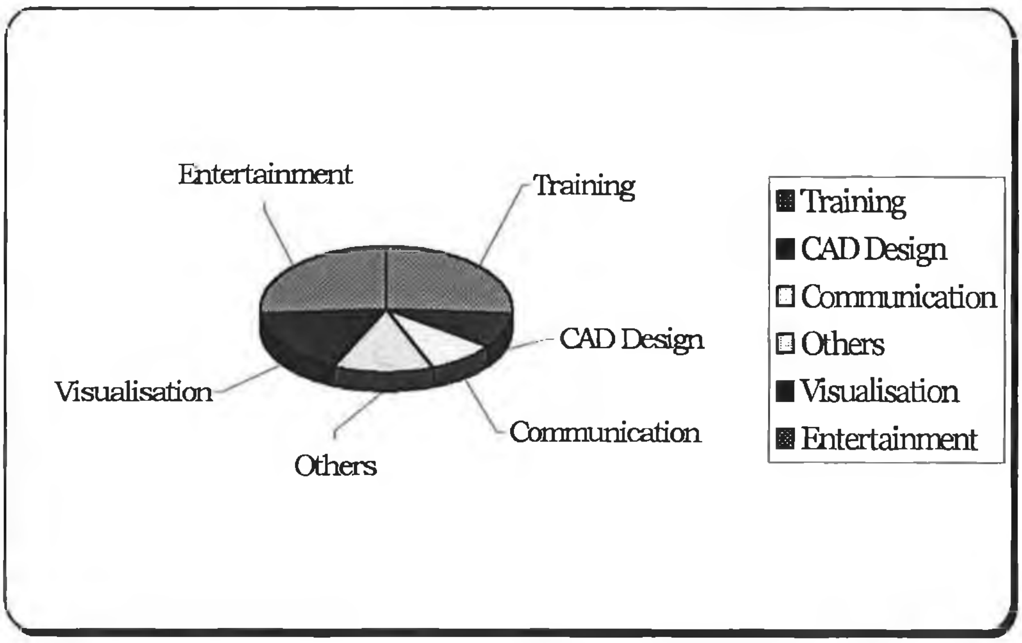


Figure 2.1: Virtual Reality Discipline Segments [2]

2.3 Virtual Reality

Virtual reality can be seen as a logical evolution of the existing human-computer interface. In the early beginning of computing, humans interacted with computers by moving physical switches on the computer itself. In 1960's, human interacted with computers using punch cards. This was followed by, in 1970's; programmers could work interactively via a terminal and keyboard with the microcomputers. Since the development of windows operating environment since 1980's, the primary means of interacting with computers is through the keyboard and mouse, while viewing the display on a monitor. Virtual reality offers a new, unique way to interact with computer data and images, and opens up new opportunities to expand the use of computer technology for engineering. [3,4]

Howard Rheingold, in his book *Virtual Reality* [5], describes VR as:
"... *An environment in which the brain is coupled so tightly with the computer that the awareness of the user seems to be moving around inside the computer-created world the way people move around the natural environment.*"

Another commonly used definition of Virtual Reality is given in the book
"The Silicon Mirage"; it stated that "*Virtual Reality is a way for humans to visualise, manipulate and interact with computers and extremely complex data*" The visualisation part refers to the computer generating visual, auditory or other sensual outputs to the user of a world within the computer. This world may be a CAD model, a scientific simulation, or a view into a database. The user can interact with the world and directly manipulate objects within the world.

Other processes, perhaps physical simulations animate some worlds, or simple animation scripts. Interaction with the virtual world at least with near real time control of the viewpoint, is a critical test for a “Virtual Reality”.

Some people object to the term "Virtual Reality", saying it is an oxymoron. Other terms that have been used are Synthetic Environments, Cyberspace, Artificial Reality, Simulator Technology, etc. VR is the most commonly used and has caught the attention of the media.

Virtual Reality techniques attempt to remove barriers of the computer interface and allow the users to visualise, manipulate and interact with computers and extremely complex data. Basically, VR is a combination of computer graphics and simulation technologies. In its most ambitious form, the goal of VR systems is to create a three-dimensional, interactive and artificial environment that is indistinguishable to the users from a real environment. People immerse themselves in a VR environment with sensory interfaces that allow them to see, feel, hear and interact in the synthetic environment, which may be a CAD model, a scientific simulation, or a view into a database. The user can interact with the world and directly manipulate objects within the world.

The applications being developed for VR run a wide spectrum, from games to architectural and business planning. Many applications are VR worlds that are very similar to our own, like CAD or architectural modelling. Some applications provide ways of viewing from an advantageous perspective not possible with the real world, like scientific simulators and telepresence systems, air traffic control systems.

Other applications are much different from anything we have ever directly experienced before. These latter applications may be the hardest and most interesting systems, for instance, visualising the flow of the world's financial markets and navigating a large corporate information base, etc.

A cursory review of some common engineering literature produces references to the virtual product [6], virtual laboratory [7], virtual workcells [8], virtual factory, virtual corporation, and virtual prototype [9]. This opens new opportunities for engineering design. Just as computer aided design techniques have enhanced the ability of engineers and led to better designs, virtual reality holds a similar promise.

The following are a selection of different applications in which VR has been used to great effect [10].

Architecture

- Walkthroughs to evaluate design decisions and/or present designs to customers.
- Demonstrate how a planned construction fits into the environment in which it is intended to be built.

Conferencing

- Collaborative work over the Internet.
- Virtual Work groups.
- Virtual Lectures and Conferences.

Training

- Civilian and Military Training Simulators
- Driving Simulators
- Flight Simulators
- Training for hazardous and difficult operations
- Nuclear Plant maintenance
- Practice Locating and Fixing Faults in Equipment

Education

- Visualise Concepts (Past and Present)
- Conferencing and Virtual Classrooms
- Medical/Surgery
- Practice Performing Surgery
- Phobia therapy
- Use of VR to teach new skills in a Safe, Controlled, Environment
- Design and Prototyping
- Used to Create Rapid Prototypes rather than make clay Models or full-scale Mock-ups
- Simulate assembly lines. For example, to evaluate the risk of interference or Collision between Robots, Cars, and Car Parts.

Scientific Visualisation

- View Complex Data Sets to gain greater insight and understanding of structure
- View Complex Molecular Structures
- View geological structures

Art

- Virtual Galleries and Museums
- Games
- Sport Simulators
- 3-D navigational interfaces to the Net

2.4 Types of Virtual Reality Systems

A major distinction of VR systems is the mode with which they interface to the user. This section describes some of the common modes used in VR systems [11,12].

2.4.1 Window on World Systems (WoW)

Some systems use a conventional computer monitor to display the visual world. This sometimes is called Desktop VR or a Window on a World (*WoW*).

2.4.2 Video Mapping

A variation of the WoW approach merges a video input of the user's silhouette with a *2D*-computer graphic. The user watches a monitor that shows his body's interaction with the world.

2.4.3 Immersive Systems

The ultimate VR systems completely immerse the user's personal viewpoint inside the virtual world. These "immersive" VR systems are often equipped with a Head Mounted Display (*HMD*) (Figure 2.2(a)). A nice variation of the immersive systems uses multiple large projection displays to create a 'Cave' or room in which the viewer(s) stand.

2.4.4 Telepresence

Telepresence is a variation on visualising complete computer generated worlds. This is a technology, which links remote sensors in the real world, with the senses of a human operator (Figure 2.2(b)). The remote sensors might be located on a robot. Fire fighters use remotely operated vehicles to handle some dangerous conditions. Surgeons are using very small instruments on cables to do surgery without cutting a major hole in their patients. The instruments have a small video camera at the business end. Robots equipped with telepresence systems have already changed the way in which a deep sea and volcanic exploration is done. NASA plans to use telerobotics for space exploration. There is currently a joint US/Russian project researching telepresence for space rover exploration.

2.4.5 Mixed Reality

Merging the Telepresence and Virtual Reality systems gives the Mixed Reality or Seamless Simulation systems. Here the computer generated inputs are merged with telepresence inputs and/or the users view of the real world. A surgeon's view of a brain surgery is overlaid with images from earlier CAT scans and real-time ultrasound. A fighter pilot sees computer generated maps and data displays inside his fancy helmet visor or on cockpit displays.

2.4.6 Fish Tank Virtual Reality

The phrase "fish tank virtual reality" was used to describe a VR system, which combines a stereoscopic monitor display using LCD Shutter glasses with a mechanical head tracker (Figure 2.2(c)). The resulting system is superior to simple stereo-WoW systems due to the motion parallax effects introduced by the head tracker.

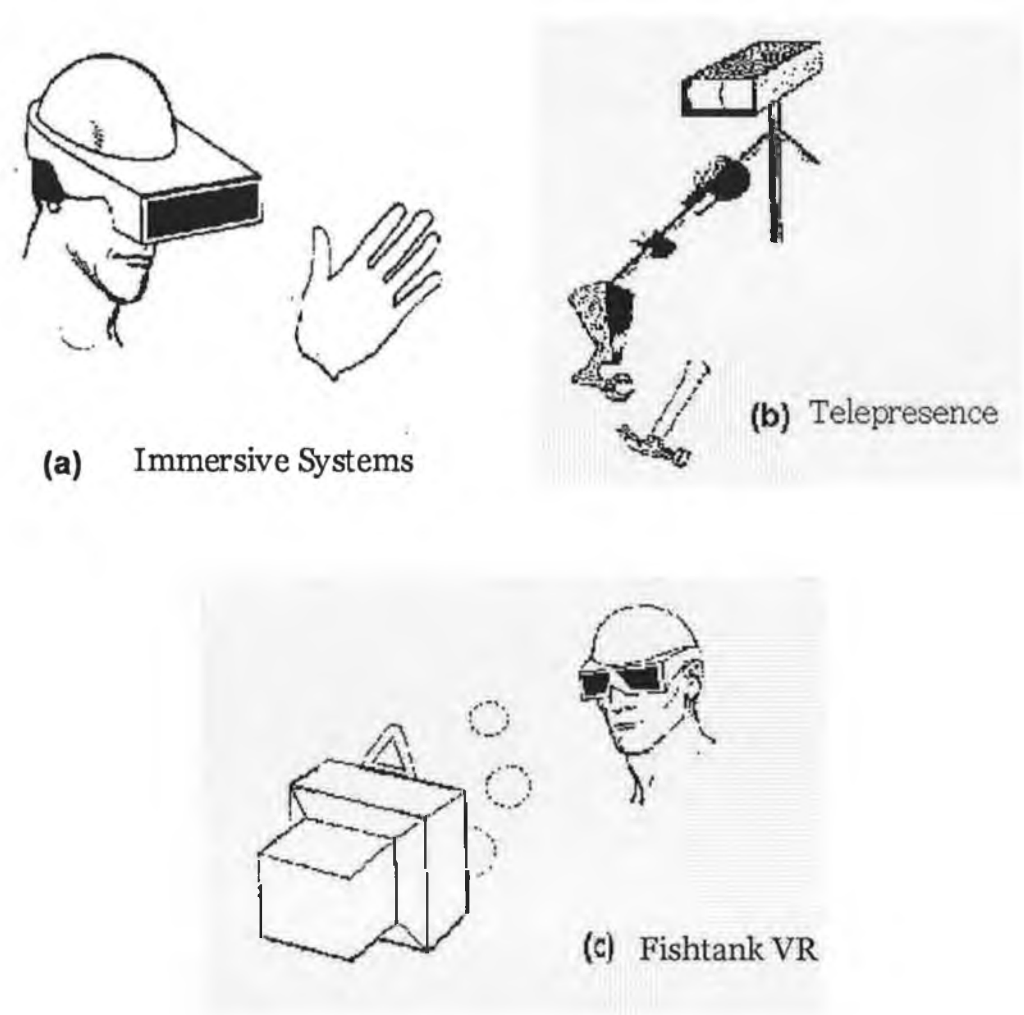


Figure 2.2: (a) Immersive Systems, (b) Telepresence, (c) Fishtank VR

2.5 Aspects of a Virtual Reality Software

Just what is required of a VR program? The basic parts of the system can be broken down into an Input Processor, a Simulation Processor, a Rendering Process, and a World Database. All these parts must consider the time required for processing. Every delay in response time degrades the feeling of 'presence' and reality of the simulation.

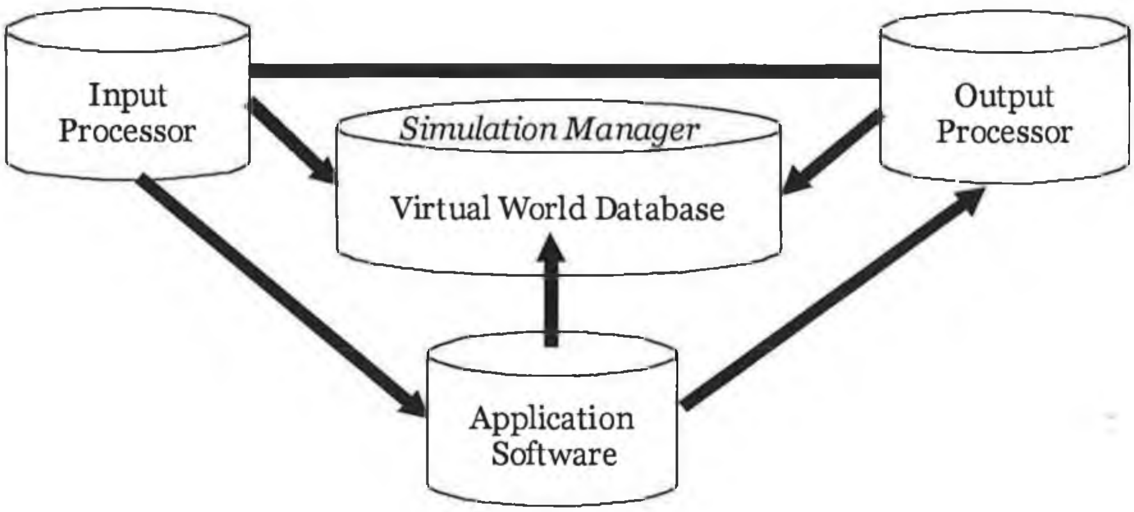


Figure 2.3: VR System Software [3,13]

2.5.1 Input Processes

The Input Processes of a VR program control the devices used to input information to the computer. There are a variety of possible input devices: keyboard, mouse, trackball, joystick, 3-D & 6-D position trackers (glove, wand, head tracker, body

Figure 2.4: the Virtual Hand is Touching an object under the control of the Glove.

suit, etc.). A networked VR system would add inputs received from net. A voice recognition system is also a good augmentation for VR, especially if the user's hands are being used for other tasks. Generally, the input processing of a VR system is kept simple. The object is to get the co-ordinate data to the rest of the system with minimal lag time. Some position sensor systems add some filtering and data smoothing processing. Some glove systems add gesture recognition. This processing step examines the glove inputs and determines when a specific gesture has been made. Thus, it can provide a higher level of input to the simulation.

2.5.2 Simulation Process

The core of a VR program is the simulation system. This process knows about the objects and the various inputs. It handles the interactions, the scripted object actions, simulations of physical laws (real or imaginary) and determines the world status. This simulation is basically a discrete process that is iterated once for each time step or frame. It is the simulation engine that takes the user inputs along with any tasks programmed into the world such as collision detection, scripts, etc. and determines the actions that will take place in the virtual world.

2.5.3 Rendering Processes

The Rendering Processes of a VR program are those that create the sensations that are output to the user. There would be separate rendering processes for visual, auditory, haptic (touch/force), and other sensory systems. Each renderer would take a description of the World State from the simulation process or derive it directly from the World Database for each time step.

2.5.3.1 Visual Renderer

The major consideration of a graphic renderer for VR applications is the frame generation rate. It is necessary to create a new frame every 1/20 of a second or faster. 20 frames per second (*fps*) is roughly the minimum rate at which the human brain will merge a stream of still images and perceive a smooth animation. Visual renderers for VR use other methods such as a 'painter's algorithm', a *Z-Buffer*, or other Scanlinge oriented algorithm. The visual rendering process is often referred to as a rendering pipeline. This refers to the series of sub-processes that are invoked to create each frame. A sample-rendering pipeline starts with a description of the world, the objects, lighting and camera (eye) location in world space. A first step would be eliminating all objects that are not visible by the camera. This can be quickly done by clipping the object bounding box or sphere against the viewing pyramid of the camera. Then the remaining objects have their geometry transformed into the eye co-ordinate system (eye point at origin). Then the hidden surface algorithm and actual pixel rendering is done.

2.5.3.2 Auditory Rendering

A VR system is greatly enhanced by the inclusion of an audio component. This may produce mono, stereo or 3D audio. Research into 3D audio has shown that there are many aspects of our head and ear shape that effect the recognition of 3D sounds. It is possible to apply a rather complex mathematical function (called a Head Related Transfer Function or "HRTF") to a sound to produce this effect.

2.5.3.3 Haptic Rendering

Haptics is the generation of touch and force feedback information. There have been very few studies done on the rendering of true touch sense (such as liquid, fur, etc.). Almost all systems to date have focused on force feedback and kinesthetic senses. These systems can provide good clues to the body regarding the touch sense, but are considered distinct from it.

2.5.4 World Space

The virtual world itself needs to be defined in a 'world space'. By its nature as a computer simulation, this world is necessarily limited. The computer must put a numeric value on the locations of each point of each object within the world.

Usually these 'co-ordinates ' are expressed in Cartesian dimensions of ***x, y,*** and ***z*** (length, height, and depth). It is possible to use alternative co-ordinate systems such as spherical but Cartesian co-ordinates are the norm for almost all applications. Conversions between co-ordinate systems are fairly simple (if time consuming).

2.5.4.1 World Co-ordinates

A major limitation on the world space is the type of numbers used for the co-ordinates. Some worlds use floating-point co-ordinates. This allows a very large range of numbers to be specified, with some precision lost on large numbers. Other systems used fixed-point co-ordinates, which provides uniform precision on a more limited range of values. The choice of fixed versus floating point is often based on speed as well as the desire for a uniform co-ordinate field. One method of dealing with the limitations on the world co-ordinate space is to divide a virtual world up into multiple worlds and provide a means of transiting between the worlds. This allows fewer objects to be computed both for scripts and for rendering.

2.5.4.2 World Database

The storage of information on objects and the world is a major part of the design of a VR system. The primary things that are stored in the World Database (or World Description Files) are the objects that inhabit the world, scripts that describe actions of those objects or the user (things that happen to the user), lighting, program controls, and hardware device support.

2.5.4.3 Storage Methods

There are a number of different ways the world information may be stored, a single file, a collection of files, or a database. The multiple file method is one of the more common approaches for VR development packages. Each object has one or more files (geometry, scripts, etc.) and there is some overall 'world' file that causes the other files to be loaded. Some systems also include a configuration file that defines the hardware interface connections. Sometimes the entire database is loaded during program start-up, other systems only read the currently needed files. A real database system helps tremendously with the latter approach. An Object Oriented Database would be a great fit for a VR system, but the anther is not aware of any projects currently using one. The data files are most often stored as ASCII (human readable) text files. However, in many systems binary computer files replace these. Some systems have all the world information compiled directly into the application.

2.5.4.4 Objects

Objects in the virtual world can have geometry, hierarchy, scripts, and other attributes. The capabilities of objects have a tremendous impact on the structure and design of the system. In order to retain flexibility, a list of named attribute/values pairs is often used. Thus, attributes can be added to the system without requiring changes to the object data structures. These attribute lists would be addressable by name (i.e. cube mass => mass of the cubic object). They may be a scalar, vector, or expression value. They may be addressable from within the scripts of their object. They might be accessible from scripts in other objects.

2.5.4.5 Position/Orientation

An object is *positionable* and *orientable*. That is, it has a location and orientation in space. Most objects can have these attributes modified by applying translation and rotation operations. These operations are often implemented using methods from vector and matrix algebra.

2.5.4.6 Hierarchy

An object may be part of an object part *Hierarchy* with a parent, sibling, and child objects. Such an object would inherit the transformations applied to its parent object and pass these on to its siblings and children. Hierarchies are used to create jointed figures such as robots and animals. They can also be used to model other things like the sun, planets and their moons in a solar system.

2.5.4.7 Bounding Volume

Additionally, an object should include a *Bounding Volume*. The simplest bounding volume is the Bounding Sphere, specified by a centre and radius. Another simple alternative is the Bounding Cube. This data can be used for rapid object culling during rendering and trigger analysis. Objects whose bounding volume is completely outside the viewing area need not be transformed or considered further during rendering. Collision detection with bounding spheres is very rapid. It could be used alone, or as a method for culling objects before more rigorous collision detection, algorithms are applied.

2.5.4.8 Object Geometry

The modeling of object shape and geometry is a large and diverse field. Some approaches seek to very carefully model the exact geometry of real world objects. Other methods seek to create simplified representations. Most VR systems sacrifice detail and exactness for simplicity for the sake of rendering speed. The simplest objects are single dimensional points. Next come the two dimensional vectors. Many CAD systems create and exchange data as 2D views. This information is not very useful for VR systems, except for display on a 2D surface within the virtual world. There are some programs that can reconstruct a 3-D model of an object, given a number of 2-D views. The sections below discuss a number of common geometric modeling methods. The choice of method used is closely tied to the rendering process used. Some renderers can handle multiple types of models, but most use only one, especially for VR use. The modeling complexity is generally inversely proportional to the rendering speed. As the model gets more complex and detailed, the frame rate drops.

2.5.4.9 3-D PolyLines and PolyPoints

The simplest 3-D objects are known as PolyPoints and PolyLines. A PolyPoint is simply a collection of points in space. A Polyline is a set of vectors that form a continuous line.

2.5.4.10 Polygons

The most common form of objects used in VR systems are based on flat polygons. A polygon is a planar, closed multi-sided figure. They maybe convex or concave, but some systems require convex polygons. The use of polygons often gives objects a faceted look. This can be offset by more advanced rendering techniques such as the use of smooth shading and texture mapping. Some systems use simple triangles or quadrilaterals instead of more general polygons. This can simplify the rendering process, as all surfaces have a known shape. However, it can also increase the number of surfaces that need to be rendered.

2.5.4.11 Primitives

Some systems provide only Primitive Objects, such as cubes, cones, and spheres. Sometimes, these objects can be slightly deformed by the modeling package to provide objects that are more interesting.

2.5.4.12 Solid Modeling and Boolean Operations

Solid Modeling (Computer Solid Geometry, *CSG*) is one form of geometric modeling that uses primitive objects. It extends the concept by allowing various additions, subtractions, Boolean and other operations between these primitives. This can be very useful in modeling objects when you are concerned with doing physical calculations, such as center of mass, etc. However, this method does incur some significant calculations and is not very useful for VR applications. It is possible to convert a CSG model into polygons.

2.5.4.13 Curves and Patches

Another advanced form of geometric modeling is the use of curves and curved surfaces (patches). These can be very effective in representing complex shapes, like the curved surface of an automobile, ship or beer bottle. However, there is significant calculation involved in determining the surface location at each pixel, thus curve based modeling is not used directly in VR systems. It is possible; however, to design an object using curves and then compute a polygonal representation of those curved patches. Various complexity polygonal models could be made from a single high resolution 'metaobject'.

2.5.4.14 Dynamic Geometry (morphing)

It is sometimes desirable to have an object that can change shape. The shape might simply be deformed; such as a bouncing ball or the squash/stretch used in classical animation, ('cartoons') or it might actually undergo metamorphosis into a completely different geometry. The latter effect is commonly known as 'morphing' and has been extensively used in films, commercials and television shows. Morphing can be done in the image domain *(2D morph)* or in the geometry domain *(3D morph)*. The latter is applicable to VR systems. The simplest method of doing a 3D morph is to pre-compute the various geometry's and step through them as needed. A system with significant processing power can handle real time object morphing.

2.5.4.15 Swept Objects and Surface of Revolution

A common method for creating objects is known as Sweeping and Surfaces of Revolution. These methods use an outline or template curve and a backbone. The template is swept along the backbone creating the object surface (or rotated about a single axis to create a surface of revolution). This Method may be used to create either curve surfaces or polygonal objects. For VR applications, the sweeping would most likely be performed during the object modeling (creation) phase and the resulting polygonal object stored for real time use.

2.5.4.16 Lights

Lighting is a very important part of a virtual world (if it is visually rendered). Lights can be ambient (everywhere), or located. Located lights have position and may have orientation, colour, intensity and a cone of illumination. The more complex the light source, the more computation is required to simulate its effect on objects.

2.5.4.17 Cameras

Cameras or viewpoints may be described in the World Database. Generally, each user has only one viewpoint at a time (two closely spaced viewpoints for stereoscopic systems). However, it may be useful to define alternative cameras that can be used as needed.

2.5.4.18 *Scripts and Object Behaviour*

A virtual world consisting only of static objects is only of mild interest. Many researchers and enthusiasts of VR have remarked that interaction is the key to a successful and interesting virtual world. This requires some means of defining the actions that objects take on their own and when the user (or other objects) interacts with them. This is referring to generically as the World Scripting. The scripts are divided into three basic types: Motion Scripts, Trigger scripts and Connection Scripts. Scripts may be textual or they might be actually compiled into the program structure.

2.5.4.19 *Motion Script*

Motion scripts modify the position, orientation or other attributes of an object, light or camera based on the current system tick. A 'tick' is one advancement of the simulation clock. Generally, this is equivalent to a single frame of visual animation. (VR generally uses Discrete Simulation methods) for simplicity and speed, only one motion script should be active for an object at any one instant. Motion scripting is a potentially powerful feature, depending on how complex we allow these scripts to become. Care must be exercised since the interpretation of these scripts will require time, which influences the frame and delay rates. Additionally, a script might be used to attach or detach an object from a hierarchy. For example, a script might attach the user to a CAR object when he wishes to drive around the virtual world. Alternatively, the user might 'pick up' or attach an object to him.

2.5.4.20 *Simple Animation*

A simpler method of animation is to use simple formulas for the motion of objects. A very simple example would be "Rotate about Z axis once every 4 seconds". This might also be represented as "Rotate about Z 10 radians each frame". A slightly more advanced method of animation is to provide a 'path' for the object with controls on its speed at various points. These controls are sometimes referred to as "slow in-out". They provide a much more realistic motion than simple linear motion. If the motion is fixed, some systems can pre-compute the motion and provide a 'channel' of data that is evaluated at each time instance. This may be a simple lookup table with exact values for each frame, or it may require some sort of simple interpolation.

2.5.4.21 Graphical User Interface/Control Panels

A VR system often needs to have some sort of control panels available to the user. The world database may contain information on these panels and how they are integrated into the application. Alternatively, they may be a part of the program code. There are several ways to create these panels. There could be 2D menus that surround a WoW display, or are overlaid onto the image. An alternative is to place control devices inside the virtual world. The simulation system must then note user interaction with these devices as providing control over the world. One primary area of user control is control of the viewpoint (moving around within the virtual world). Some systems use the joystick or similar device to move. Others use gestures from a glove, such as pointing, to indicate a motion command. The user interface to the Virtual World might be restricted to direct interaction in the 3D world. However, this is extremely limiting and requires many 3D calculations. Thus, it is desirable to have some form of 2D graphical user interface to assist in controlling the virtual world. These 'control panels' of the world appear to occlude portions of the 3D world, or perhaps the 3D world would appear as a window or viewpoint set in a 2D screen interface. The 2D interactions could also be represented as a flat panel floating in 3D space, with a 3D effector controlling them.

2.6 Aspects of Virtual Reality Hardware

There are a number of specialised types of hardware devices that have been developed or used for Virtual Reality applications [13].

2.6.1 Image Generators

One of the most time consuming tasks in a VR system is the generation of the images. Fast computer graphics opens a very large range of applications aside from VR, so there has been a market demand for hardware acceleration for a long while. There are currently a number of vendors selling image generator cards for PC level machines, many of these are based on the Intel i860 processor. These cards range in price from about $2000 up to $10,000. Silicon Graphics Inc. has made a very profitable business of producing graphics workstations. SGI boxes are some of the most common processors found in VR laboratories and high-end systems. SGI boxes range in price from under $10,000 to over $100,000. The simulator market has produced several companies that build special purpose computers designed expressly for real-time image generation. These computers often cost several hundreds of thousands of dollars.

2.6.2 Manipulation and Control Devices

One essential element for interaction with a virtual world, is a means of tracking the position of a real world object, such as a head or hand. There are numerous methods for position tracking and control. Ideally a technology should provide three measures for position (X, Y, Z) and three measures of orientation (roll, pitch, yaw). One of the biggest problems for position tracking is latency, or the time required for making the measurements and pre-processing them before input to the simulation engine. The simplest control hardware is a conventional mouse, trackball or joystick. While these are two-dimensional devices, creative programming can use them for 6D controls. There are a number of three and six dimensional mice/trackball/joystick devices being introduced to the market at this time. These add some extra buttons and wheels that are used to control not just the X &Y translation of a cursor, but its Z dimension and rotations in all three directions. The Global Devices 6D Controller is one such 6-D joystick it looks like a racket ball mounted on a short stick. You can pull and twist the ball in addition to the left/right & forward/backward of a normal joystick. Other 3D and 6D mice, joystick and force balls are available from Logitech, Mouse System Corp. among the others. One common VR device is the instrumented glove. A Basic Patent in the USA covers the use of a glove to manipulate, objects in a computer as shown in Fig 2.5. Such a glove is outfitted with sensors on the fingers as well as an overall position/orientation tracker. There are a number of different types of sensors that can be used such as; fiber optic sensors for finger bends and magnetic trackers for overall position.

Figure 2.5: A typical VR Data Glove.

2.6.3 Position Tracking

Mechanical armatures can be used to provide fast and very accurate tracking. Such armatures may look like a desk lamp (for basic position/orientation) or they may be highly complex exoskeletons (for more detailed positions). The drawbacks of mechanical sensors are the encumbrance of the device and its restrictions on motion. Exos Systems builds one such exoskeleton for hand control. It also provides force feedback. Shooting Star system makes a low cost armature system for head tracking. Fake Space Labs and LEEP Systems make much more expensive and elaborate armature systems for use with their display systems. Ultrasonic sensors can be used to track position and orientation. A set of emitters and receivers are used with a known relationship between the emitters and the receivers. The emitters are pulsed in sequence and the time lag to each receiver is measured. Triangulation gives the position.

Drawbacks of ultrasonic are low resolution, long lag times and interference from echoes and other noises in the environment. Logitech and Transition State are two companies that provide ultrasonic tracking systems. Magnetic trackers use sets of coils that are pulsed to produce magnetic fields. The magnetic sensors determine the strength and angles of the fields. Limitations of these trackers are a high latency for the measurement and processing, range limitations, and interference from ferrous materials within the fields. However, magnetic trackers seem to be one of the preferred methods. The two primary companies selling magnetic trackers are Polhemus and Ascension.

Optical position tracking systems have been developed. One method uses a ceiling grid LEDs and a head mounted camera. The LEDs are pulsed in sequence and the camera image is processed to detect the flashes. Two problems with this method are limited space (grid size) and lack of full motion (rotations). Another optical method uses a number of video cameras to capture simultaneous images that are correlated by high-speed computers to track objects. Processing time (and cost of fast computers) is a major limiting factor here.

2.6.4 Stereo Vision

Stereo vision is often included in a VR system. This is accomplished by creating two different images of the world, one for each eye. The images are computed with the viewpoints offset by the equivalent distance between the eyes. There are a large number of technologies for presenting these two images. The images can be placed side-by-side and the viewer asked (or assisted) to cross their eyes. The images can be projected through differently polarised filters, with corresponding filters placed in front of the eyes. Anaglyph images use red/blue glasses to provide a crude (no colour) stereovision. The two images can be displayed sequentially on a conventional monitor or projection display. Liquid Crystal shutter glasses are then used to shut off alternate eyes in synchronisation with the display. When the brain receives the images in rapid enough succession, it fuses the images into a single scene and perceives depth. A high display-swapping rate (minimum 60Hz) is required to avoid perceived flicker. Another alternative method for creating stereo imagery on a computer is to use one of several split screen methods. These divide the monitor into two parts and display left and right images at the same time. One method places the images side by side and conventionally oriented. It may not use the full screen or may otherwise alter the normal display aspect ratio. A special hood viewer is placed against the monitor which helps to position the eyes correctly and may contain a divider so each eye sees only its own image. Most of these hoods, such as the one for the V5 of Rend386, use fresnel lenses to enhance the viewing. An alternative split screen method orients the images so the top of each points out the side of the monitor. A special hood containing mirrors is used to correctly orient the images.

2.6.5 Head Mounted Display (HMD)

One hardware device closely associated with VR is the Head Mounted Device. These use some sort of helmet or goggles to place small video displays in front of each eye, with special optics to focus and stretch the perceived field of view. Most HMDs use two displays and can provide stereoscopic imaging. Others use a single larger display to provide higher resolution, but without the stereoscopic vision. Most lower cost HMDs ($3000-10,000 range) use LCD displays, while others use small CRTs, such as those found in camcorders. The more expensive HMDs use special CRTs mounted along side the head or optical fibers to pipe the images from non-head mounted displays ($60,000 and up). A HMD requires a position tracker in addition to the helmet. Alternatively, the display can be mounted on an armature for support and tracking (a Boom display).

Figure2.6: A typical VR Head Mounted Display (HMD).

2.6.6 Levels of VR Hardware Systems

The following defines a number of levels of VR hardware systems. These are not hard levels, especially towards the more advanced systems.

Entry VR *(EVR)*: The 'Entry Level' VR system takes a stock personal computer or workstation and implements a WoW system. The system may be based on an IBM clone (MS-DOS/Windows) machine, an Apple Macintosh, or perhaps a Commodore Amiga. The DOS type machines (IBM PC clones) are the most prevalent. There are Mac based systems, but few very fast rendering ones. Whatever the base computer it includes a graphic display, a 2-D input device like a mouse, trackball or joystick, the keyboard, hard disk & memory.

Basic VR *(BVR)*: The next step up from an EVR system adds some basic interaction and display enhancements. Such enhancements would include a stereographic viewer (*LCD* Shutter glasses) and an input/control device such as the Mattel PowerGlove and/or a multidimensional (3-D or 6-D) mouse or joystick.

Advanced VR *(AVR)*: The next step up the VR technology ladder is to add a rendering accelerator and/or frame buffer and possibly other parallel processors for input handling, etc. The simplest enhancement in this area is a faster display card. For the PC class machines, there are a number of new fast VGA and SVGA accelerator cards. These can make a dramatic improvement in the rendering performance of a desktop VR system. Other more sophisticated image processors based on the Texas Instruments TI34020 or Intel i860 processor can make more improvements that are dramatic in rendering capabilities. The i860 in particular is in many of the high-end professional systems. The Silicon Graphics Reality Engine uses a number of i860 processors in addition to the usual *SGI* workstation hardware to achieve stunning levels of realism in real time animation. An AVR system might also add a sound card to provide mono, stereo or true 3D audio output. Some sound cards also provide voice recognition. This would be an excellent additional input device for VR applications.

Immersion VR (IVR): An Immersion VR system adds some type of immersive display system: a HMD, a Boom, or multiple large projection type displays (Cave). An IVR system might also add some form of tactile, haptic and touch feedback interaction mechanisms. The area of Touch or Force Feedback (known collectively as Haptics) is a very new research arena.

2.7 Virtual Environments

The recent surge in VR technologies gives a new impetus to development of new and better training solutions. With the use of a virtual environment (VE), interaction between student and machine is possible in a whole new way. Virtual Environments are made up of 3-D graphical images that are generated with the intention of interaction between the user and objects in that environment [2]. Presently, many VE applications are developed for situations, which appear to be hazardous. An immersive VE enable the users to observe responses in the form of sensation of touch, sight, force and sound. Table 2.2 illustrates the technologies applied to VEs today.

Table 2.2 Technologies applied to VEs today [14,15]

Technology	***Description***	***Features***
Simulators	Projected display, Sound (and Vibration) and replica of physical surroundings (e.g. cab or flight deck)	Often expensive, usually dedicated to specific applications, high quality experience
Head Mounted Display (HMD)	Screens and lenses fitted in goggles or helmet, giving stereoscopic, binocular display; frequently have earphones for auditory environment; head and trackers allow continual updating of display for user movement and orientation.	Range from cheap to relatively expensive; use with range of sophistication in VE software and graphics engine.
Head Coupled display	CRT monitor and controls supported on universally jointed stand. The monitor is held and moved as if it was a large, heavy pair of binoculars.	Improved graphics, fast tracking, increased comfort; expensive.
Mixed environment	Use of HMD with some replication of 'hard' features of environment (e.g. seat, steering wheel).	Approaching a flexible simulator.
Augmented Reality display	Information from computer system overlaid onto view of a real world, for instance 'see through' displays on windscreen or helmet visor.	Probably not a virtual environment
Artificial Reality	Video cameras capture participant body movements that are included within large display of the generated virtual environment.	Inflexible.
Desktop or (Monitor)	Virtual environment displayed on desktop screens; control via variety of '3-D' input devices.	Improved graphics quality, flexible and user comfort over HMDs, possibly at the expense of 'Presence'. Range of software and hardware options from very cheap to very expensive. Can have HMDs fitted for necessary applications.
Wall Mounted	As for desktop but display enlarged and projected on wall.	Greater sense of immersion that for desktop; less display quality unless very expensive. Inflexible
Spatially immersive display (SID).	As for Wall Mounted, but across several walls, ceiling.	As for Wall Mounted

Training should provide people with the knowledge and skills to meet the demands of their jobs and achieve the goal of the organisations to which they belong to [16]. To use the concept of a VE as training method involves, as in any other training method, knowledge of how people learn. The student visual understanding of a particular situation better, reading the procedure from a manual.

Zeltzer (1991) created a model for Characterising virtual environments. His unit cube model identifies three essential components, which all VR systems must have, as well as three dimensions or properties that can be used to compare virtual environments. The three essential components are; (i) A set of models objects or processes, (ii) A means of modifying the state of these models, and (iii) A range of sensory modalities to allow a participant to experience the virtual environment.

The three properties that Zeltzer proposes for measuring and comparing virtual environments are.

(i) Autonomy – The extent to which objects can respond to events and stimuli (both from each other, the environment and the user).

(ii) Interaction – The degree of access to the parameters or variables of an object,

(iii) Presence – a measure of the fidelity of sensory cues that give the sense of physical presence. Zeltzer's model takes into account a large number of VE systems, from immersive to non-immersive systems. "Interactivity is considered to be the core element of VR", [17].

The illusion of immersion in a virtual environment is created through the operation of three technologies, which provide a functional breakdown as an alternative to the preceding abstract analysis. These are (1) Sensors, such as head position or hand shape sensors, to measure operator's body movements, (2) Effectors, such as a stereoscopic displays or headphones, to stimulate the operator's senses and, (3) special-purpose hardware and software to Interlink the sensors and effectors to produce sensory experiences resembling those encountered by inhabitants immersed in a physical environment as shown in figure 2.7. In a virtual environment this linkage is accomplished by a simulation computer. In a head-mounted teleoperator display the linkage is accomplished by the robot manipulators, vehicles, control systems, sensors and cameras at a remote worksite. Though the environment experienced with a teleoperator display is real and that experienced via the virtual environment simulation is imaginary, digital image processing allows the merging of both real and synthetic data making intermediate environments of real and synthetic objects also possible. Truly remarkable displays will be possible fusing sensor data geographic databases. The successful interaction of a human operator with virtual environments presented by head and body referenced sensory displays depends upon the fidelity with which the sensory information is presented to the user. The situation is directly parallel to that faced by the designer of a vehicle simulator. In fact, since virtual environments extend flight simulation technology to cheaper, accessible forms, developers can learn much from the flight simulation literature [18].

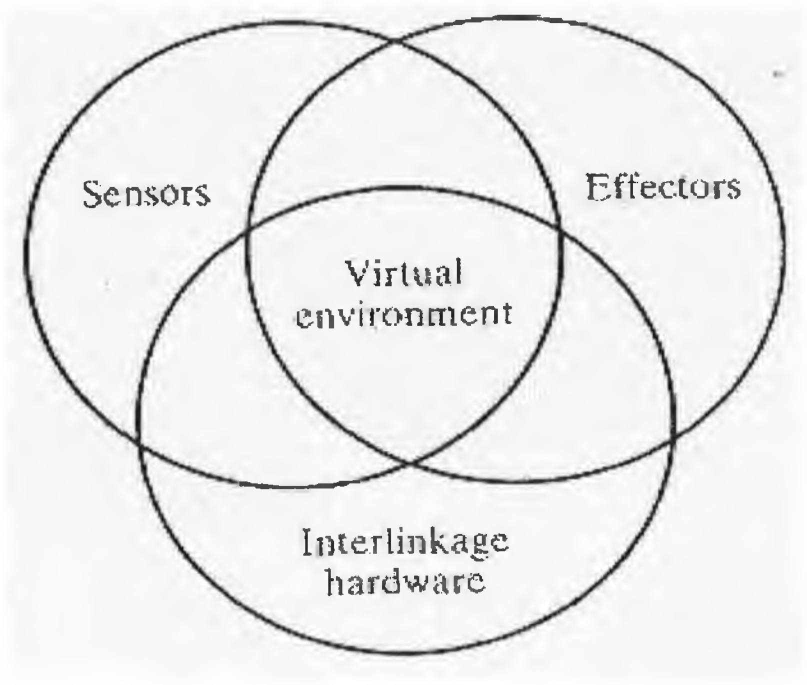
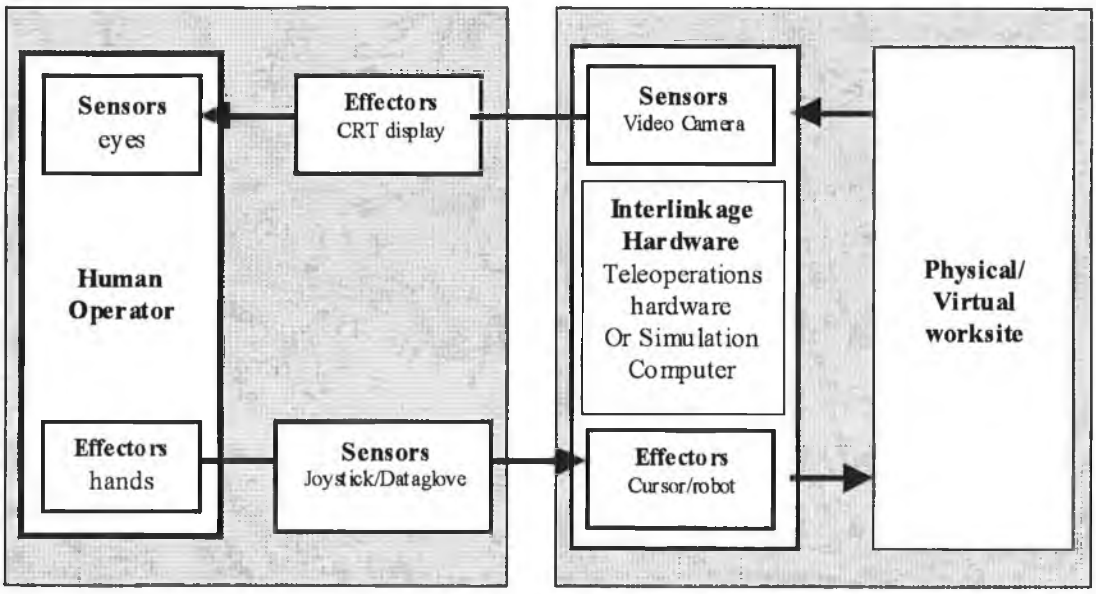


Figure 2.7 (a) & (b): Technological breakdowns of virtual environments.



Virtual environments are simulators that are generally worn rather than entered. They are personal simulators. They are intended to provide their users with a direct sense of presence in a world or space other than the physical one in which they actually are. Their users are not in a cockpit that is within a synthetic environment, they are in the environment themselves.

Though this illusion of remote presence is not new, as mentioned earlier, the diffusion and rapid drop in the cost of the basic technology has raised the question of whether such displays can become practically useful for a wide variety of new applications ranging from video games to laparscopic surgical simulation.

2.8 Virtual Reality and Engineering

Virtual reality holds the potential to revolutionise the way in which humans interact with computers. Engineers rely on computers to help in building, testing and verifying designs. Virtual reality offers a new and innovative way to interact with complex data and designs. Some of the areas where VR can contribute to increase engineering productivity are in the areas of design, prototyping, design for maintenance and assembly, factory planning, networked design, and concurrent engineering.

2.9 Influence of VR in Mechanical design

Virtual reality (VR) allows a person to experience phenomena that appear to be real but exist only in the computer. Thus, the user (or designer) “feel” immersed, as if he or she is actually “in” the three-dimensional space [11]. Using VR based systems; a user or designer can visualise a final product, assemblies, sub-assemblies or components. The moving objects and all in-between situations of the simulated world can be viewed with the presentation techniques of VR, like stereo viewing, fly by or walkthrough.

Using VR for product design could provide design flexibility by allowing the exploration of various options and the opportunity to play "what if" experiences where mistakes are less expensive. Significant cost savings could be achieved in system development and production because many of the problems would be identified and corrected before the actual physical product construction. The virtual reality technology would enable developers to refine designs before commitments are made, bringing users into the design process much earlier, and allow engineers to solve problems in a collaborative, group setting.

A newly developed process can be seen in computer screen first, when this is just finished in design stage. In this way, the VR simulated system can often help to find potential mistakes in the initial stages and correct as many discrepancies as possible before the expensive real model is actually built. In addition, virtual model built in a computer will be conveniently applied to a lot of purpose. At the design stage of models, designers can evaluate the feasibility such as dimensions, ergonomics design, and other human needs.

The development of a virtual design is expected to be valuable for the education and training of engineering personnel. Using virtual reality techniques, it is possible to generate the three dimensional graphics in real-time, concurrently with simulation generation, allowing the user to move around, and interact with the simulated process in a natural and intuitive manner.

2.10 Case Studies: Virtual Engineering

Ford: The Ford automotive company has set up a development division called Ford Alpha Simultaneous Engineering. Vehicle parts, which are represented in a CAD system, are transferred to a VR system in a way that enables the user to manipulate the virtual part. The user can then attempt to assemble the parts into the virtual vehicle. This system will also check for collisions between parts and the vehicle. The developed system is expected to be useful to evaluate the human ergonomics of various assembly operations.

Boeing: The Research and Technology organisation of Boeing Computer Services is actively involved in VR technology. Boeing uses a concept known as Augmented Reality. Augmented Reality is a term, which refers to the ability to see-through a computer-generated display. The generated images are superimposed on reality. This accomplished by projecting a computer image onto a half-silvered mirror, which user looks through. This technique provides a very effective and intuitive way of "annotating" reality. The Boeing team is using a headset configured for augmented reality, which they call a HUD set (Heads-Up, see-through, Head-Mounted Display). Virtual Reality is now used to help in the assembly process of Boeing planes.

Porsche: Stuttgart, Germany is the location of the world's first virtual crash simulation laboratory. Porsche is using this test facility as an important step to increase safety. Faster time to market and higher development efficiency are further benefits of this innovative technology. The constant increase in complexity of crash tests for cars, due to higher safety standards and new materials, requires more simulations that in detailed. A significant improvement is obtained using VR for a more intense understanding of numerical crash simulation results. At some point, the real crash of the prototype is inevitable but VR will lead to enormous timesaving and a reduction in the number of crash test [2].

2.11 Virtual Design Verses Conventional Design

The design process for Virtual Reality applications has two driving requirements:

- The virtual environment and its interface should be tailored to the task.
- Stringent performance constraints must be met for the benefit of virtual reality to be realised [18].

The first requirement is in recognition of the fact that immersing the user in a three-dimensional computer-generated environment presents many opportunities not easily found in conventional “Desktop” three-dimensional graphics. Indeed much of the hype surrounding virtual reality is the recognition that one can “ do anything one can imagine ” in VR. While this is a highly hyped statement, it contains a grain of truth: virtual reality affords the opportunity to completely tailor the virtual environment to the task at hand. How to use this freedom effectively raises issues of overall design, design of the user interface, and important issues of human factors. The second requirement refers to the fact that the virtual environment must run with a certain minimal speed in order to be usable. Roughly, everything must happen at least ten frames per second and the system must respond to the user within a tenth of a second.

***Data Visualisation*:** Data Visualisation uses VR for viewing data in 3-D in order to get better ideas of the meaning of it. Examples, such as viewing molecules in 3-D, allow chemists to view the molecules from all angles and model the behaviour of chemical bonding. In addition, physicists can model various problems in virtual reality to help understand the problems of, among others, aerodynamics. Boeing, the American aircraft manufacturer, built a virtual reality wind tunnel, to demonstrate the unseen effects of wind passing over their models. Building aircraft models in virtual reality allows many different designs to be prototyped without the overheads and difficulty of building physical models and trying to experimentally determine their characteristics in a real wind tunnel [3,19].

Astronaut Training: VR has recently been used on the Hubble telescope space mission. One of the astronauts' tasks was to attempt to polish the surface of the telescopes massive lens. Space walks have been done many times before, but this mission required the manual control of equipment at the lens. Using VR, the astronauts manipulated virtual equipment through the same joysticks they were to use in the mission. The virtual equipment responded to the manipulation, as if in a zero-gravity environment, teaching how inertia effects differ in space. NASA declared the experimental training for the mission a success.

Virtual Design: VR presents the opportunity to design in a 3D environment. In the future, engineers will be able to mold and stretch 3D surfaces, 3D holes, fillet 3D corners, etc. without having to contend with a 2D display (traditional monitor). Designers will be able to work in the same area as the design itself, watching and evaluating the design as it takes shape. Virtual design will also give new insights into the interpretation of analysis results such as stress, fluid, and thermal analysis among others. Visualising the analysis results on a 3D model which is in the virtual design space will significantly enhance the engineer's ability to pinpoint trouble spots and areas of the design that are of interest. Analysis errors will become apparent when viewed in the virtual environment. In addition, unexpected results will also be discovered by virtue of using this display/interaction environment. Coupling VR technologies with super computers for a calculation purpose opens the possibility of watching virtual crash tests results [20, 21].

Virtual Prototyping: Most engineering applications of VR at present time are focused on the development of virtual prototypes. Ergonomic assessment of visibility, reachability, accessibility, clearance, comforts and aesthetics is generally performed on a physical prototype. If these same kinds of assessments can be performed on a virtual prototype, significant savings will be realised in the design of new machines.

The intent is not to eliminate physical prototypes altogether, but reduce the number of prototypes that must be built before production of the new design in scheduled. Reducing the number of prototypes will save money, and also decrease the time to market for new designs.

Engineers at the Ford Motor Company are developing virtual prototypes to answer questions such as "Will this part fit? Does this engine come together? Can I reach this knob or dial in the interior" [9]. While, Caterpillar Inc., the world's largest

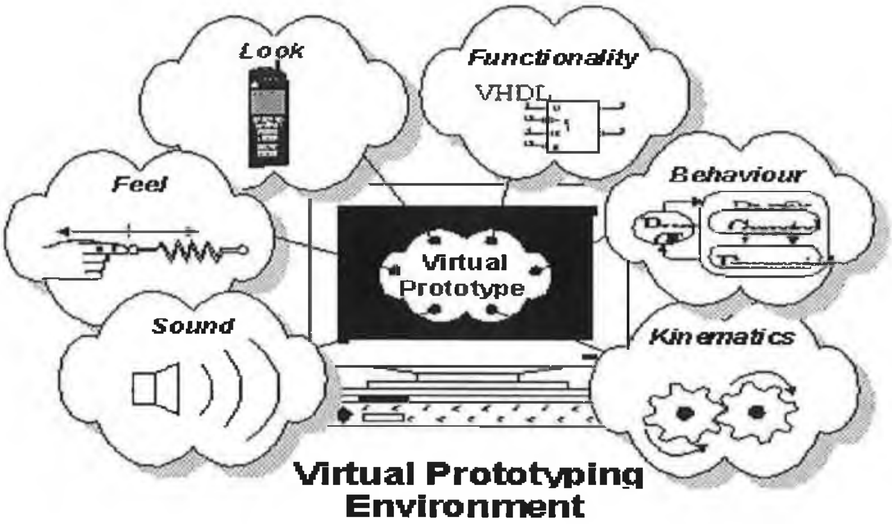


Figure 2.8: Simulated features in typical VP for consumer electronics & telecommunication domainn **[28]**

Manufacturer of earth moving and construction equipment, is using a CAVE (Cave Automatic Virtual Environment) to investigate assessing operator visibility [12]. This type of assessment is very difficult to perform using existing 3D modeling tools. In addition, building several physical prototypes is very expensive. Virtual reality Allows the designer to sit in the driver's seat and look around to evaluate the degree of visibility available. In the future, the designer will be able to reach out and alter the computer model, by moving the virtual surfaces, to produce a design that allows for increased visibility. With virtual designs, if it is easy to alter a design, more iteration will be performed, which will result in more designs that are robust.

Maintenance Planning in Design: Another area where virtual reality will become important is in design for maintenance planning. Manufacturers and designers have become aware of the need to plan for maintenance access to complicated designs. Concurrent engineering design principles advocate bringing design, manufacturing, marketing, sales, and maintenance experts into the design process to provide input to the design. This has resulted in additional demands placed on the design, including the desire for designs, which are easily maintained. In determining, where to place individual components of a design, engineers are increasingly looking for arrangements that provide for easy maintenance. A virtual model of the design can be used to plan and verify the maintenance tasks. Design iterations can be performed on the virtual model to improve maintainability of the product.

The U.S. Army Tank Automotive and Armaments Command is experimenting with bringing maintenance solders into the design facility and soliciting their opinions on design changes needed to improve maintainability [9]. This can be done without the use of expensive prototypes when using a virtual environment. Engineers at Lockheed used VR to plan the positioning of the corrective optics to be installed in the Hubble Space Telescope [23]. Although this was not a planned maintenance, VR was able to contribute significantly to planning the "fix".

Assembly Planning in Design: Similarly, virtual reality can be used for assembly planning. In a virtual environment, users can interact with the computer models in the same way as they would interact with the real models. Assemblies can be taken apart and re-assembled. Difficulties due to assembly tasks will be discovered early in the design process and costly redesign will be avoided. In essence, virtual reality can be used as a dry run through for the assembly operations early in the design process where design changes are less costly to implement.

Factory Planning: Currently, 3D-factory simulation and modeling software is commercially available. Individual machine tools can be modelled and placed on a 3D-factory floor. Material flow simulations can be coupled with the computer models to simulate the flow of the parts through the factory. Virtual reality will provide an enhancement to these existing capabilities by allowing the operator to be immersed in the factory along with the computer models of the machines. Again, reach studies and ergonomic assessments can be performed in the virtual environment. Machines can be easily rearranged to provide for operation that is more efficient.

Concurrent Engineering: Virtual reality provides enhanced visualisation capabilities that will improve concurrent engineering practices. Engineers are accustomed to looking at multiview drawings and visualising the three-dimensional shape of the design. Other members of the concurrent engineering design team are not as accustomed to reading these drawings and have more trouble visualising the shape and function of the final design. Even 3-D computer models are sometimes difficult to understand. Viewing computer models in a virtual environment will be similar to viewing physical prototypes of the design. People can walk around and look under the model, open the doors, move the designs to verify its motion, etc. This will provide non-engineers with a computer model that more closely mimics a real model than current 3-D capabilities.

Networked Virtual Design: Networked virtual reality opens up many possibilities for engineers. The U.S. government has been researching applications of networked virtual battlefields for several years. Michael Zyda and others have been actively researching networked VR for use in large-scale virtual battlefield [24].

Engineers can apply this networking in another area: networked virtual design. Members of the concurrent design team in various locations can share the same virtual design space, modify, and discuss the same virtual objects. Marketing members and sales members from across the country will be able to enter the virtual environment and converse with others on the design team.

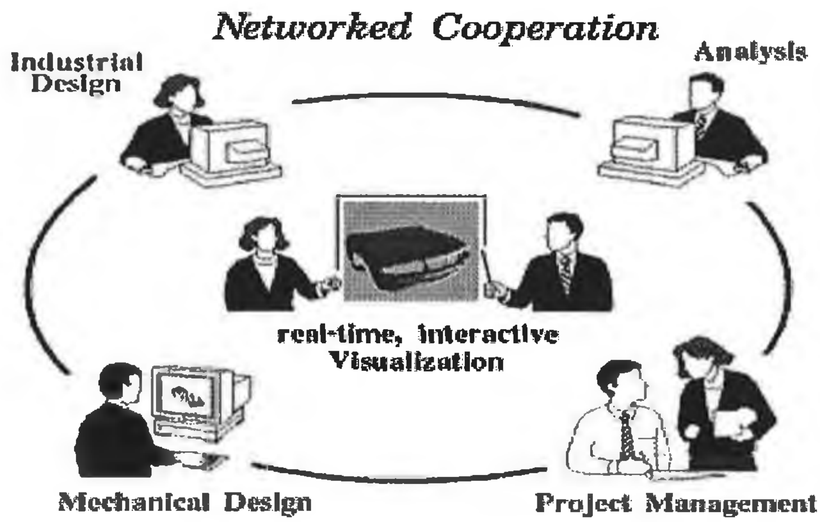


Figure 2.9: Network System

Virtual Reality and Simulation Technology: Computer simulation is the discipline of designing a model of an actual or theoretical system, executing the model on a computer and analysing the output. Simulation is used to develop a level of understanding of the interaction of parts in a system. Computer graphics is the driving force towards a true representation of the system to be simulated. Simulation is often used because the level of understanding achieved is seldom achievable with other disciplines. A simulation is made up a model and system.

- A system is an entity, which maintains its existence through the mutual interaction of its parts, whereas,
- A model is a simplified representation of the parts that make up the system.

Simulators, such as the flight simulator, were first built for the aircraft industry and the U.S. Air Force. The flight simulator dates back to 1929 when Edwin A.Link patented the first ground-based flight trainer (Stark, 1992). Stark cited 1934 as the first time the U.S.Army used simulators to train their pilots. Students Pilots learned how to manoeuvre aeroplanes by manipulating controls in specially built cockpits. These cockpits were initially removed from the aeroplanes and mounted on moveable platforms that tilted and rolled, based on the pilots' actions on the controls.

Since that time, flight simulators have advanced well beyond basic instrument and radio navigation trainers. Today, simulators enable pilots to 'feel' the simulated emergency in motion-based systems and conduct air-to-air combat in visually based systems [25]. A major limitation of these early simulators was that they lacked visual feedback. This changed when video displays were coupled with the model cockpits and now the video displays have been replaced with highly detailed virtual world.

2.12 Safety Considerations

Virtual Reality is not always beneficial to its users. Researchers have shown that the use of Virtual Reality itself has caused many side effects. These side effects include Repetitive Stress injury, Immersion injury, transmittable disease, Cybersickness, and psychological effects. Repetitive Stress injury can be a fairly common occurrence. One develops this injury through repetitive motions, such as using a joystick, or keyboard controls.

Immersion injury is a result of VR. A user experience immersion injury when the user becomes engrossed in the program, and as a result, becomes disoriented with the surroundings. Consequently, the user tries unsafe acts such as running while wearing a head monitor device (HMD). Transmittable diseases can be a result from VR. Diseases such as Hemophilus influenza and multiple viruses are transmitted through HMD's and hand controllers that have multiple users.

Cyber sickness is very similar to motion sickness. Common symptoms are headaches, drowsiness, and disorientation. Lastly, psychological effects can also be created using VR. Although the psychological effects have no lasting effect on the user, effects while using VR can be anywhere from acrophobia to claustrophobia [20,26].

Leslie Harris of Susquehanna University has studied another side effect. Harris hypothesises that VR has a " reality-making " effect on users. To state this more clearly, our past experiences determine what we perceive to be real, and our actions within an environment are determined as a result of our experiences. For instance, Harris cites an example of car driving. In VR, one can manipulate a car in order to drive it by pushing buttons, or moving joysticks. Those who are frequent users of VR begin to develop a new " driving script".

The driving script allows a user to become adjusted to the use of buttons to drive the virtual car. The user soon begins to accept the VR version of driving as normal. When the user tries to drive a car in the real world, the user experiences disorientation. No longer does the user have buttons or joysticks to control the car, but a steering wheel and pedals. The disorientation of a user is a dangerous possibility of Virtual Reality [11,27].

2.13 Conclusion

In conclusion, it is evident that Virtual Reality will have enormous effects on society, as we know it. The possibilities are endless, and Virtual Reality will no doubt be able to simulate almost every aspect of life. VR is an invention that will open many doors for Scientists, education, training, shopping, travelling, medicine, entertainment, business and industry will all be affected by the mass emergence of VR. Who knows what the future will bring? We will just have to wait and see!

Chapter Three: Development of Virtual Milling Machine for Training & Educational Purposes

3.1 Introduction

Technology in industry is changing rapidly; therefore, new and better ways of training people are required. Virtual Reality (VR) has a lot of potential for use in the training sector, from simulating surgical techniques to operating workshop machines. This chapter presents a review of present training philosophies and techniques and discusses the development stages of the Virtual Milling Machine (*VMM*) will used for training and educational purposes.

3.2 Education, Training and Learning

Education, Training and learning are concepts that should be given priority in an organisation, in order to develop the skills of the employees. Employee development in the process of an employee going through learning experience. If an employee has learned something then that employee has the ability to do something, which was not previously within his/her capability. This learning ability is achieved though education and training. Training can be thought of as a means of making better use of human resources in an organisation. Training is a planned process to modify skill through a learning experience. Its purpose, in the work situation, is to develop the abilities of the individual and to satisfy the current and future needs of the organisation. These needs are essential for a company to be competitive in today's marketplace.

"The acquisition of skills through education and training has long been acknowledged as one of the principal forces driving economic growth. Effective training programs can result in increased productivity and quality" [29].

3.2.1 Training Modes

Every organisation should have a training plan in order to succeed. A successful training plan is one in which the person being trained aquires a new skill and knowledge of a specific topic. Two different learning modes exist, incidental and intentional. Incidental learning is a learning mode we encounter every day. This type of learning is generally accidental and is achieved through social activities, such as having a conversation with another crafts-man relating to a different skill.

If a new skill is learned or knowledge gained that was not planned then this is classified as incidental learning. An industrial training plan cannot rely on such a learning modes as specific skills need to be learned at specific time. Seymour [30] believed that training programs, where the tasks are subdivided into small parts, are more beneficial to the learner so that he/she can concentrate on one item at a time. This is still true today as educational psychology is used in schools to create a system of instruction known as mastery learning. Mastery learning is based on the belief that students achieve higher grades if the curriculum is broken down into logically sequenced units.

3.2.2 Safety Training

It is the organisation's responsibility to ensure that the training program adopted has sufficient content in the area of operator safety. The British courts have rule since the mid-1800s that "an employer is no less responsible to its workers for personal injuries cause by a defective system for using the plant machinery than for injuries caused by a defect in the machinery itself" [31]. The most important part of safety training is to pre-program trainee for possible hazardous situation so that they can recognise instantly a potentially hazardous situation.

3.2.3 The Psychology of Training

The psychology implications of a training method adopted by an organisation should be carefully analysed and dealt with correctly, especially when a computer training application is adopted. Often, a fear surrounds the use of computers partly because of a feeling of not being in control. This can be due to badly designed user interface. For some people the prospect of learning a new skill may generate a feeling of being challenged, but others may tend to shy away from new responsibility, often due to a fear of the unknown. Bentley [32] believes that the following are critical psychological barriers, which are true for all forms of change; *Fear of the Unknown; Self-Doubt; Fear of Ridicule; and Fear of Failure.* Once these psychological barriers are removed and the positive side for training is realised, only then can the actual training be beneficial.

3.2.4 Personal Development

In order for an organisation to grow and become successful, it has to plan its future and how the people within the organisation develop as employees. "*Development should enable individuals to grow and be ready to meet changes and challenges they* *will certainly face, and the organisation to become a place where learning is allowed and respected and which continually re-evaluates the way in which it operates*" [33].

The best way to explain the importance of personal development is by viewing the training cycle in figure 3.1. All aspects of this cycle need to give careful consideration before the training can be effective.

The benefits of personal development planning are:

- It is supported from the top and shows that management is interested in staff development and is prepared to put time and resources in to demonstrating that commitment.
- Staff is being valued for what they are doing- they are given the opportunity to talk about what they are achieving and given the opportunity to take things further.
- It is morale boost – it shows everyone that your business knows where it is going and has the professional systems to back this up.
- It can form the basis of structured career development for everyone.
- It develops the team to work more effectively – to be more dynamic, responsive and forward looking.
- It moves the whole organisation forward.

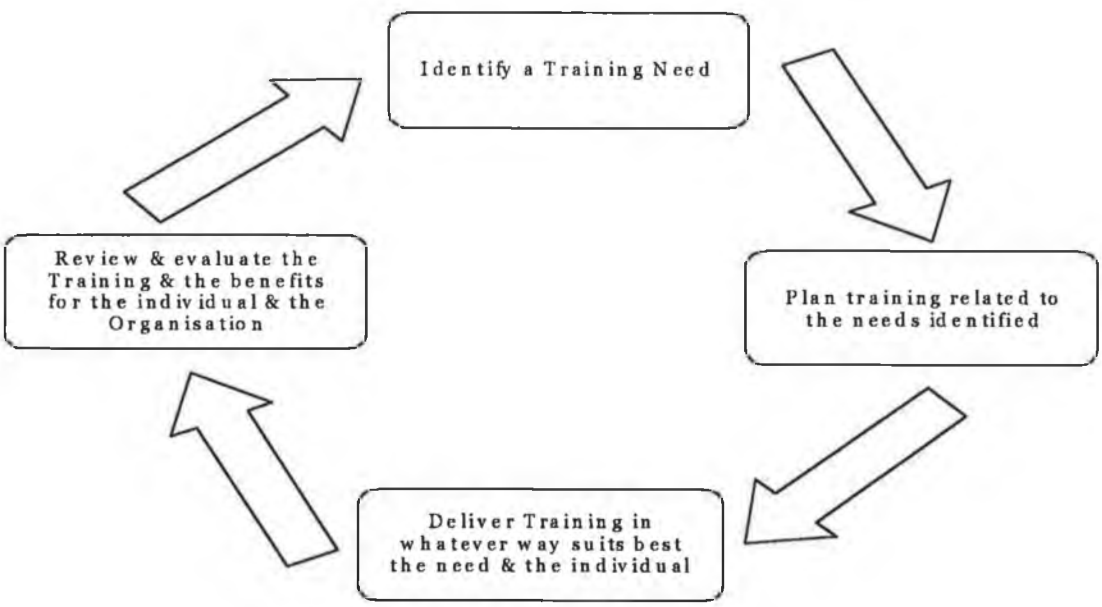


Figure 3.1 The Training Cycle

3.2.5 The Effect of Training

Any organisation that wants to prosper has to recruit and maintain a workforce consisting of people who are versatile in their work and willing to learn and develop their skills continuously. The improved performance of individuals will, in turn, lead to increased profit. It is with this mind that training should be thought of as an investment and proper resources should be allocated. "Spending provides the satisfaction of some immediate need. Investment provides satisfaction of some future return" [32]. This means that investments will usually take place before benefits are received. The resulting is a time lag, which is illustrated in figure3.2.

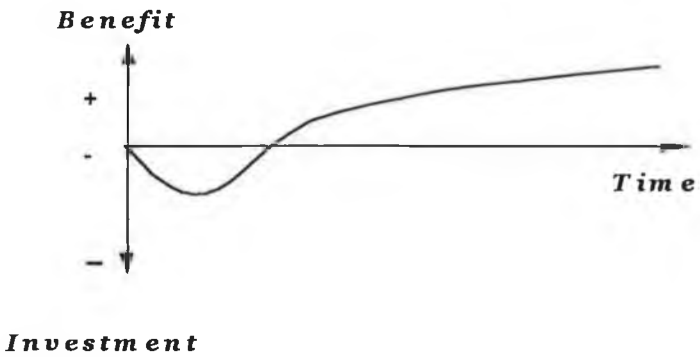


Figure3.2: Training Investment Curve

3.2.6 Evaluating a Training Method

Questionnaires are often used to achieve responses on a specific type of training plan. Written data is gathered and can be hand processed or machine processed, depending on both the quantity and the complexity of the data. During the design stage of the questionnaire, the trainee being evaluated must be taken into consideration.

3.3 Computer Training Techniques

Computers are changing the way we live our lives and the way we train people in the workforce. Various forms of media have been used in the past but computers are now offering a more interactive approach to training. Specific media formats have varied widely, beginning with a focus on workbooks and educational films during the 1940's, changing to video based instruction in the next several decades and currently emphasising the use of computers and multimedia as instructional tool [34]. Each one of these forms of instructional tools were at the cutting edge of their respective technologies at the time of their implementation and now it is the turn of computers using simulation techniques for instructional teaching.

3.3.1 Computer Aided Instruction

Computer Aided Instruction (CAI) is a technology expanding to assist in the teaching and learning process, which a student undergoes. CAI is also known as Computer Assisted Instruction. This form of teaching instruction is growing rapidly and is used in such areas as computer visualisation of complex objects. Information can be represented to the student in text or other multimedia formats such as photographs, simulation and speech. The potential of computer as a teaching tool is immense as applications can be programmed to judge student input and offer a wide range of tutorials.

3.3.2 Multimedia

Multimedia is a combination of text, graphic art, sound, animation and video presented to the user by a computer or other electronic means. By using these features in a software application, "you can electrify the thought and action centres of peoples minds". Multimedia can be used wherever people need access to electronic information. The effect of multimedia is to enhance traditional text-only computer interfaces and it is known to yield benefits by gaining and holding the attention of users. Multimedia also provides a way to reach out to people who are intimidated by computers. Therefore, multimedia is a technology aimed at making computers easier and more interesting to use. VR is an extension of multimedia as it uses the basic multimedia elements of images, sound and animation. VR is, possibly, interactive multimedia at its best.

3.4 Virtual Milling Machine Training

3.4.1 The Need for the Machine Training

Training results in improved quality, which can increase productivity and can contribute to good work practice and a safer working environment. In the area of manufacturing, the productivity of any task can be said that to be directly related to the proficiency of the operator performing the task. This is true in many disciplines of engineering i.e. the time it takes for a design engineer to draw a component using a CAD system is directly related to the designer's CAD system proficiency. Therefore it stands to reason that efficiency and productivity directly related to satisfactory training. Training and Education are more important as we re-engineer companies and increase the use of technology on a day to day basis " [35].

3.4.2 The Benefits of Training

There is a direct correlation between *optimum* machine equipment utilisation and the proficiency of machine's staff in a machine environment. If the machine's staff do not operate efficiently, cycle time will suffer. The benefits of implementing a training program are numerous, such as, (i) *Minimises Mistakes; (ii) Improves quality; (iii) Enhanced Consistency*; and (iv) *Enhances Safety* [36].

3.5 Conclusion

In summary, Virtual Reality is fast becoming a valuable tool in education. The future of training is destined to become a Virtual Reality based interactive process, as the student may participate in the learning experience actively and not passively. Virtual Reality can provide the mechanism to allow the trainee to participate in a learning experience actively and he/she will benefit, since people comprehend images much faster than they can grasp lines of text or columns of numbers. Virtual Reality can help students to simulate and train in environments e.g. operation of milling machines, that otherwise may be too expensive and/or dangerous.

Chapter Four: Virtual Reality Hardware & Software: an Overview

4.1 Introduction

This chapter identifies and describes the software and hardware selected to create the most realistic presentation of Milling Machine for a training application. Superscape VRT 5.6 was selected as the most suitable VR software. It allows the virtual world designer to attach intelligence to objects in a virtual world by adding a programme that will mimic the characteristics of each object. A virtual world can be modelled using VRT or imported from an external CAD application in order to make up a series of objects. To obtain the most realistic model, the Milling Machine was initially modelled using 3D-Studio Max. The Milling Machine CAD file was then exported to the VRT system. The Bridgeport Milling Machine was selected as a suitable machine to model as its primary function is for the use in the educational industry and tool making. Intelligence was then programmed into various parts of this model using the Superscape Control Language (SCL) which is similar to the C programming language.

4.2 The Virtual Reality Software

The potential for using VR technology as a means of simulating various environments is growing and the creation of virtual world is only limited to the imagination of the virtual world designer.

Software companies have realised the potential of VR and are striving to make it more accessible to PC owners. Superscape VRT is a commercially available virtual reality system from Superscape Ltd. and was selected as the most appropriate VR software for this project. Superscape introduces the first PC-based virtual reality software product, which offers the user ability to design, and attaches intelligence to objects in the worlds.

Superscape VRT is a single user facility but recent developments offer optional networking capabilities to provide some degree of multi-user participation. Worlds that are designed using VRT with the intention for desktop VR use can be later modified in order to view the world with VR glasses, thus giving a limited degree of immersion. Superscape offers a variety of powerful features including the ability to edit and create all aspects of the virtual environment.

Also incorporated is a comprehensive selection of object attributes both static and dynamic, which allow objects to behave in various pre-programmed ways. VRT provides its own C-like interpreted language; SCL (Superscape Control Language) which allows the user to attach scripts to objects. SCL permits the user to create complex sophisticated behaviour within virtual world [37].

4.3 VRT Hardware and Software

Superscape VRT 5.6 is one of the fastest software solutions for building intelligent virtual environments on a PC. VRT uses a point and click interface in order to accomplish a user-friendly environment in a 3-D world. It consists of a suit of editors that are used to build a desired world. It also offers two browser platforms, Viscape and Visualiser. Viscape is a real-time 3-D Web browser and is available as a plug-in for Netscape Navigator. Visualiser is available as a stand-alone window application for use by end-users. The world designer will use all the editors on offer in the world development process, but the end-user can view the world by downloading Viscape from the Internet. VRT 5.6 is designed to be installed on Intel based PCs with Pentium processor or later version, running Windows 95, or Windows NT 3.51 (NT 4 recommended). The minimum and recommended requirements for running VRT 5.6 are shown in table 4.1

Table 4.1. Minimum and recommended requirements for running VRT 5.6 [37].

Component	*Minimum*	*Recommended*
Processor	***P5 66 MHz***	***Fastest PC available***
RAM	***16 MB***	***64 MB***
Hard Disk	***500 MB***	***1 GB***
Graphics Card	***SVGA 640x480***	***SVGA 1280x1024***
Sound Card	***None***	***Sound Blaster 16***
Monitor	***SVGA 30 cm***	***SVGA 48 cm***
Input Device	***Mouse***	***Spacemouse***

Details about the main elements and objects in Superscape VRT Software are given in Appendix A.

4.4 The Superscape Control Language (SCL)

Superscape VRT offers its own C-like language called the Superscape Control Language (SCL). SCL programs can be attached to objects within the virtual world to assign different behaviours, thus, in effect, giving the particular item intelligence. All the syntax relating to SCL is based on the C programming language. A simple SCL program consists of list commands that is executed sequentially. For interaction to take place, the object with the SCL program has to be activated. The simplest form of activating an object is to click on the object in the virtual world with the aid of a mouse. For example, a simple SCL program may consist of just two lines.

if (activate(me,0))
togvis(me);

This SCL program is an attribute, which can be attached to an object and will change the visibility of the object if it is activated by a mouse click. SCL programs are made up of argument statements and instructions. If a piece of information is passed to an instruction then this is known as an argument. Arguments are always enclosed in brackets immediately after the instruction itself. A complete line of instructions and arguments is known as a statement, which are always terminated by a semicolon. The first line of the previous short program checks to see if the mouse has activated the object "me". If the object has activated, the 'if' instruction lets the next statement be performed, which will toggle the visibility of the object "me". The "me" in the second line of the program could also be changed to a different object number. Then, the "me" object is activated. A different object will have its visibility changed, thus applying a higher degree of intelligence [2, 38].

4.5 CAD Software

Computer Aided Design (CAD) that relates to using a computer to display and manipulate design made of geometric representations. The computer provides the user, viewing capabilities that allow the design to be viewed in any desired position or perspective view. More recent developments of CAD allow the designer to modify existing models by using parametric design techniques, instead of a complete new design for and small alteration that may be required. CAD is now a common feature in engineering departments, which can out perform traditional methods by up to twenty times the number of hours to produce some design. CAD has also excelled in the manufacturing environment, and currently being used to design parts and entire manufacturing processes [2,39,40].

4.5.1 Parametric Modelling

The latest technology in modelling is utilised in this project by combining AutoCAD software and 3-D Studio Max with Superscape VRT. AutoCAD Designer is a parametric, feature-based solid modelling tool within the AutoCAD Development system (ADS). Designer gives the modeller the ability to sketch in the industry standard 2-D AutoCAD environment and then automatically create a 3-D parametric solid model. Previous 3-D draughting packages, while excellent for their time, lack the capability for the designer to alter previous designs significantly. Parametric capabilities offer the designer the option to change an initial profile of a model, thus updating the previous model and its current drawing. With this type of technology at hand for designers today, the next logical step for a designer would be to visualise a model in a virtual environment. If a model can be easily manipulated using software such as AutoCAD and easily imported into VR software such as VRT 5.6, then the potential cost savings are immense. It is thought that the prototyping industry will be the first to benefit, and that the technology will ultimately reach all major design disciplines.

4.5.2 Parametric

Parametric is the use equations that solve engineering problems using a set of geometric and dimensional values in a design. Applying Parametric to model defines the shape and the size of that model. Editing these parameters causes the design to change. A part can initially be drawn in much the same way as a freehand sketch. A profile can then be applied to this sketch and Designer applies constraints that it considers applicable to the profile. These constraints can vary from all radii to have the same radius, to certain lines to be parallel. The design process usually changes. Designer enables the dimension parameters in a model to be changed and the dimensioning scheme so that drawing and documentation (e.g. Bill of Materials) can take place concurrently with the evolution of the design.

4.5.3 Feature-based design

Features in Design are industrial standard objects such as hole and chamfer. Traditional solid modelling techniques involved in creating a hole in a solid, meant that firstly the solid part was extruded and a cylinder was put in a hole position. The cylinder would then be subtracted from the original solid to give the impression of a hole. An obvious disadvantage of this design technique is that the hole size and position could not be changed later. Designer, with the aid of its parametric capability enables an actual hole to be drilled in a solid and all modifications relating to the size and position can be altered. Even the type of the hole applied can be modified i.e. C'bore, C'sink. A feature-based system allows you to create and edit models faster than you could with traditional methods. Every part in a system could be designed using the above technology and all the parts would be assembled into their respective positions with appropriate constraints. When you constrain components, you indicate how one component relates to another. As you constrain, you eliminate degrees of freedom, or types of movement, for those components. Interference's may occur in any design problem but an assembly display of the final solution will show these faults. Each part can be selected from the assembly and updated as a single drawing. Every assembly or sub-assembly that the component appears in will then update to the new drawing revision number ready for inspection of the updated design.

4.6 Conversion of an AutoCAD Model to a Virtual Model

In order to create a realistic virtual model of the milling machine, an AutoCAD model was imported into the world editor in VRT as a Data Interchange Format (DXF) file. A DXF file is an ASCII (American Standard Code for Information Interchange) coded file, of an AutoCAD drawing for importing and exporting to and from other software packages. The AutoCAD DXF (Drawing Interchange Format) is associated with the CAD application created and maintained by Autodesk. DXF was initially developed to represent data used in CAD programmes but is now being used for many different types of data, most commonly vector-oriented information but also text. Currently almost any type of data can be represented in a DXF. A DXF file consists of up to seven sections: a header, tables, blocks, classes, objects, entities, and an end of file marker.

- ✓ The Header section contains internal AutoCAD variables such as default settings.
- ✓ The Tables section contains information lists such as layer names and linetypes.
- ✓ The Block section contains all internal block creations of the designer.
- ✓ The Classes section holds the description of any application-defined objects that may be initiated in the block section.
- ✓ The Objects section contains non-graphical parts of the drawing e.g. AutoCAD dictionary.
- ✓ The Entities section contains the actual object data of the drawing such as lines and arcs.
- ✓ The end of the DXF data is marked with an EOF directive on the last line of the file.

VRT includes a module that will import DXF files, in which solids that are created as a drawing file are converted in VRT, to a series of facets. Figure 4.1 illustrates the process of converting a realistic AutoCAD model in a Superscape VR model. 3-D Studio files were used to change solids in the CAD application to surfaces as problems occurred when importing solids to VRT i.e. 3-D Studio is a CAD application which may be used to convert solids into surfaces.

The data converter module is mainly for converting data into three-dimensional VR formats. The conversion process involves a substantial amount of calculation and processing time, as the DXF file does not contain much of the specific information required by VRT i.e. DXF file were not created specifically for importing blocks into VRT. Therefore, the missing information is generated by the data converter, which is often obtained by guesswork but can later, be corrected by the user. In most cases, however, a VRT data file bearing a close resemblance to a 3D representation of the original CAD model is produced. The VRT converter is recommended for use on single objects in order to obtain the best result i.e. each individual object should be run through the VRT converter.

Therefore, the amount of times the above cycle would have to be carried out is equal to the amount of parts that make up the AutoCAD model. All block definitions are converted directly into shapes. These shapes are then checked against any restrictions applied by the designer and the converter that attempts to correct them. One major problem with the conversion process is that facets may be created facing the wrong way (inside out). This causes rendering problems and the designer has to 'flip' the facets in the correct direction. All the shapes are then stored as world objects with their associated information. Once all the shapes are imported into the virtual world, the parts that make up the virtual milling machine can then be assembled into their correct positions with the aid of VRT's collision detection and position commands [2].

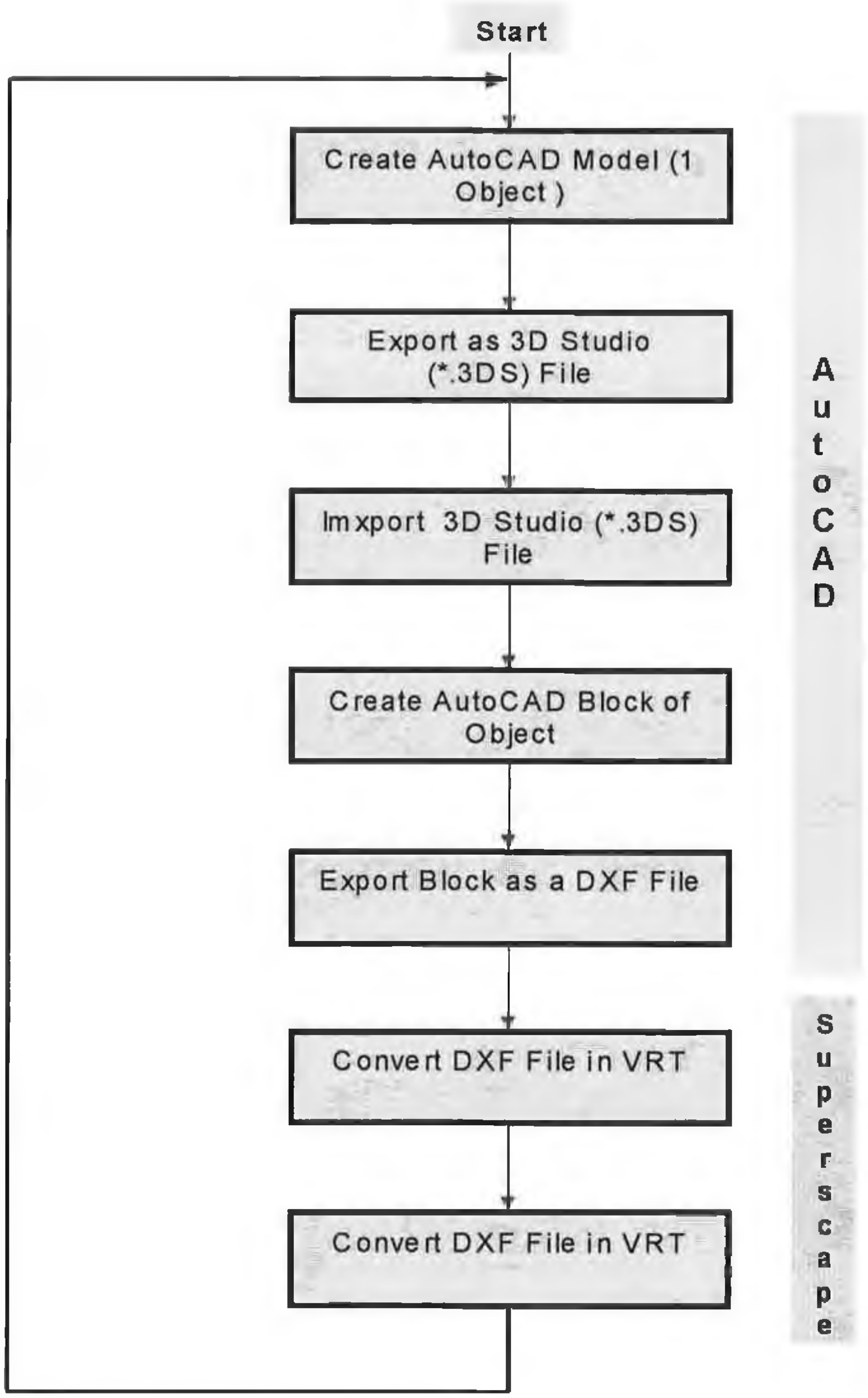


Figure 4.1 *Conversion Process of the AutoCAD Model to a Superscape Virtual Model.*

4.7 3-D Studio Max Hardware and Software

3-D Studio Max is a commercially available advanced 3D modelling and animation based simulation software. It is a PC- based software selected for this project, which offers the users the ability to design, facilities to add, subtract and combine, and can add material to object. Modelling, editing, lighting, rendering and animation tools are always available in the command panels and toolbars.

Designer can draw 2D and convert it to 3D by using tool bars and designer can view in front view, left view, top view, and in perspective view. This software helps the designer to change an initial profile of a model, thus updating the previous model and its current drawing. Animated graphical simulation enhance our understanding of abstract concepts, aid analysis, increase human productivity and reduce design cost. Creation of a virtual Machine using 3-D solid modelling and animation based simulation system can provide a fast, effective method of visualising and 'experiencing' new designs which can be easily modified to consider design alternatives [41,42].

3D Studio Max 1.2 is designed to be installed on Intel based PC's with Pentium processor or better, running windows 95, Windows NT 3.51 (NT 4 recommended. The minimum and recommended requirement for running 3D Studio Max 1.2 is shown in table 4.3.

Table 4.2. Minimum & recommended requirements for running 3-D S Max [41]

Component	*Minimum*	*Recommended*
Computer Processor	*90 MHz*	*Faster PC available*
RAM	*32 MB*	*64-128 MB*
Hard Disk	*100 MB*	*200-300 MB Depending on scene*
Graphics card	*800x600x256 Colour*	*PCI or VLB-based graphics card at 1024x768x256 Colours*
Sound card & Speakers	*None*	*Optional but recommended*

4.8 Conclusion

Animated graphical simulation enhance our understanding of abstract concepts, aid analysis, increase human productivity and reduce design cost and time. Creation of a *Vertical Milling Machine* using 3-D solid modelling and animation based simulation system can provide a fast, effective method of visualising and 'experiencing' new designs which can be easily modified to consider design alternatives. Therefore, Superscape VRT was selected as a suitable software package to model the virtual *Vertical Milling Machine* for training purpose. This software is inexpensive for the end-user in order to view the training application on Milling Machine. The *Virtual Milling Machine* was first developed using AutoCAD R13 and 3D Studio Max. It was later imported into VRT and intelligence was attached to various parts of the machine in order to mimic the working characteristics of the actual machine. The virtual modelling machine was developed using Superscape directly. The intelligence was attached to the various parts of the machine.

Chapter Five:
The Developed System, Results and Discussion

5.1 Introduction

This chapter introduces the concept of a Graphical User Interface (GUI) and shows how the developed system appears to the user. The end-user of the developed VR application is presented with a series of virtual environment in which he/she has the ability to explore and interact with objects in the world with the aid of the mouse.

By the user navigating around the virtual milling machine-training laboratory, he/she can rehearse various operations to attain familiarity with the machine. This developed system will afford the opportunity to repeat a procedure at any time of the cycle and minimise risks of being in contact with a very dangerous world.

In addition, the visual presentation of the virtual training environment is described in this chapter. A selection of views presented in order to give the reader a true perspective of the virtual lab and what a trainee may see on screen.

A questionnaire was completed by a number of participants who used the developed application. The result of which were documented and conclusion were drawn from each answer. These conclusions are presented to the latter part of this chapter

5.2 Graphical User Interface (GUI)

A Graphical User Interface (GUI) is a computer display which enables the user to choose commands from a list of available options by pointing and clicking on texture pictorial representations (icons) on a computer screen. Usually a choice of activation exists between a mouse click and the keyboard. All Microsoft Windows applications have a similar GUI appearance in order to ensure that little or no learning is required when switching between different applications. The user communicates with the computer via the keyboard and mouse and generally, the computer responds via the computer screen with messages. It is with this in mind that a study was undertaken at Stanford University on the Physiological and psychological responses to user [2]. *Consequently, it was recommended that GUI designers should adhere to the following points:*

- Error message should have a consistent tone and style.
- User should be able to customise their software.
- Avoid the use of alarming alerts.
- Try to adopt a context-sensitive help system.
- Toolbars should remain invisible until necessary.
- Attach random praise messages.

GUIs in the future are expected to change radically due to recent advancements in 3D graphics. 3-D graphics and other advanced techniques such as VR will enable more information than ever to be displayed.

A major transition period took place during the change over from DOS PC users to graphical desktops. Not much has changed since the introduction of the GUI system. GUI's are now starting to evolve into tailor-made applications but retain essential similar characteristics of applications. Each one of the developed virtual training environments is designed with a similar GUI in order to create a simple easy to use interface for the use. For example, operators with little experience may use this training application. Each virtual training environment presents the virtual milling machine with an appropriate GUI with similar essential characteristic [2].

5.3 Bridgeport Vertical Milling Machine

5.3.1 General Descriptions

Milling machines differ from lathes in that the work-piece is usually stationary and the cutting tool rotates. The milling machine produces mainly flat surfaces although complicated curved surfaces are also produced. The surface produced may be plane or, by using special cutters, formed surfaces may be produced. There are many types and size of milling machines, but the most versatile in common use in the majority of workshops is the knee and column type, so called because the spindle is fixed in the column or main body and the table arrangement, mounted on a knee, and is capable of movement in the longitudinal, transverse, and vertical directions [43,44,45].

Knee-and-Column machines are subdivided into the following models:

- Plain horizontal, with the spindle located horizontally.
- Universal, this is similar to the plain horizontal but equipped with a swivelling table for use when cutting helical grooves.
- Vertical, with the spindle located vertically.

Typical vertical Knee-and-Column milling machine is shown in figure 5.1

The machine dimensions are 2145 mm (height) x 915 mm (length) x 610mm (width) and basic weight of the machine is approximately 2200 lb. (997 kg) fitted with table size 1065 x 230 mm

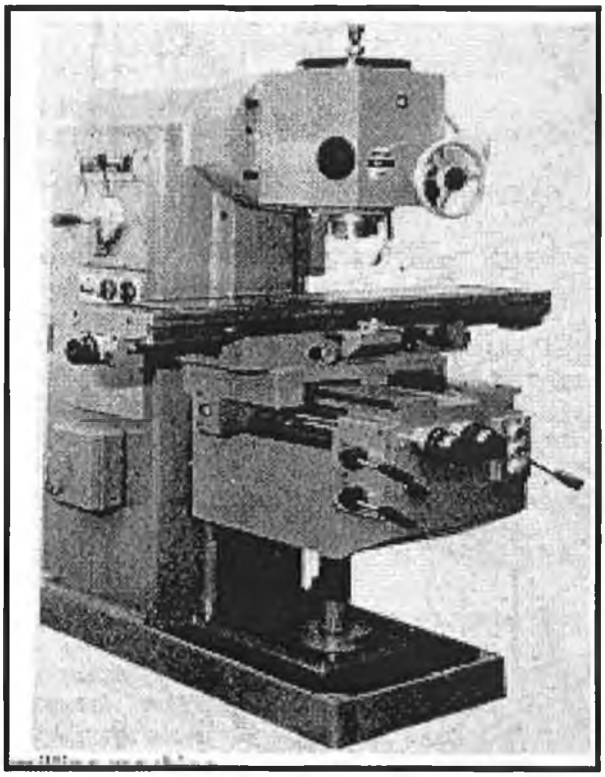


Figure 5.1: Vertical Milling Machine[43]

5.4 Milling Machine Main elements

The main elements of a typical knee-and-column horizontal-milling machine are shown in figure 5.2. The element of a vertical machine are the same except that the spindle head is mounted at the top of the column, as shown in figure 5.3

Column and Base

The column and base form the foundation of the complete machine. Both are made from cast iron, designed with thick sections to ensure complete rigidity and freedom from vibration. The base, upon which the column is mounted, is also the cutting-fluid reservoir and contains the pump to circulate the fluid to the cutting area.

Knee

The knee, mounted on the column guideways, provides the vertical movement of the table. The knee has a guideways on its top surface giving full-width support to the saddle and guiding it in a transverse direction. A lock is provided to clamp the knee in any vertical position on the column.

Saddle

The saddle, mounted on the knee guideways, provides the transverse movement of the table. Two clamps on the side of the saddle achieve clamping of the saddle to the knee. The saddle has dovetail guideways on its upper surface, at right angles to the knee guideways, to provide a guide to the table in longitudinal direction.

Spindle

The spindle, accurately mounted in precision bearings, provides the drive for the milling cutters. Cutters can be mounted straight on the spindle nose or in cutter-holding devices, which in turn are mounted in the spindle, held in position by a drawbolt passing the hollow spindle. On vertical machines, provision is made for axial movement, which is controlled by a handwheel on the spindle head.

Workholding

The simplest method of hold a work-piece for milling is to clamp directly to the worktable. Adequate Tee slots are provided for this purpose. Moreover, there are other methods for holding the work-piece such as a Vice and Rotary table.5.5

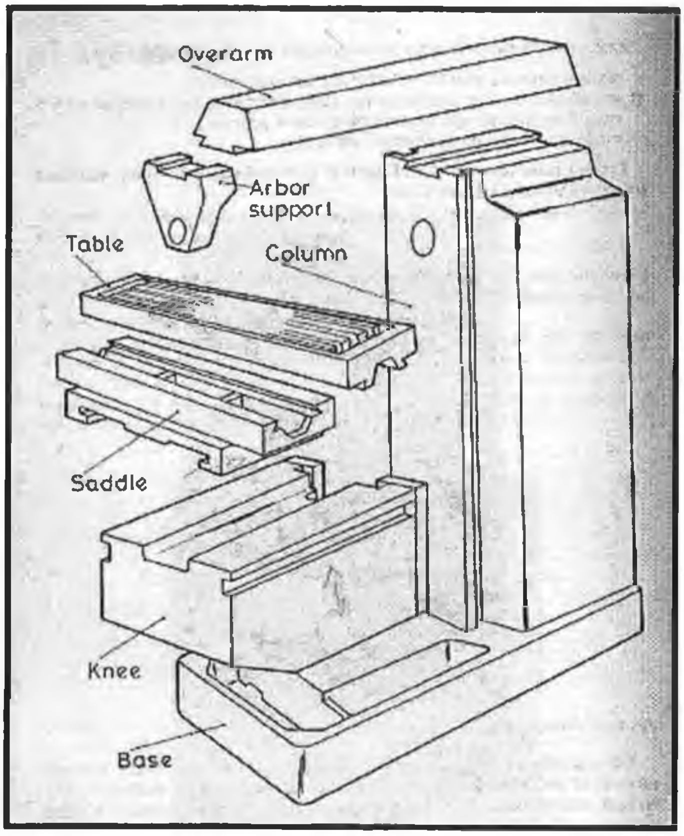


Figure 5.2: Main machine elements [43]

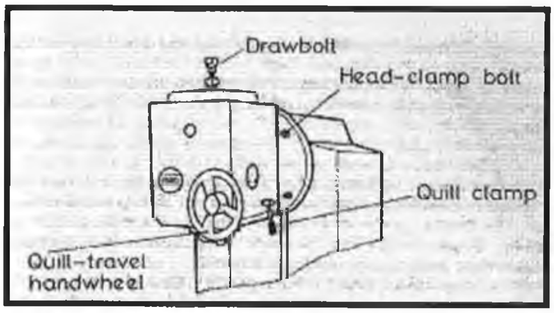


Figure 5.3: Top of Column of Vertical Milling Machine [43]

5.5 The Developed Training Application

5.5.1 Introduction

VR technology in various forms has been used for many years in the development of training systems, particularly for Mechanical and Industrial applications. However, these trainers usually require expensive, specialised, often-customised hardware and software, and so have been limited to uses where the training value justifies their high costs. The advent of commercial VR software development packages, which run on standard commercial hardware platforms, has greatly reduced the cost of using VR technology in training applications. VR is particularly useful for the Equipment Operator Training, Maintenance Training, Flight trainers, and Medical training.

The Virtual Equipment Operator and Maintenance Training Lab (VEOMTL) developed in this project for the Dublin City University (D.C.U.) makes use of these technological advances to add low-cost VR technology to an Operate and Maintenance training system programme.

The VEOMTL is intended to be the starting point in the School of Mechanical & Manufacturing Engineering in D.C.U to train the students how to operate and maintain the workshop machines such as a Virtual Milling Machine. The operation and maintenance of these kind of high cost machines requires familiarity with appearance and location of interior components such as Table, Saddle and Spindle movements.

Since most of the students do not have an open routine access to these Machines or test equipment in the lab, simulating these Labs in a virtual environment provides many of the benefits of having the actual equipment available without the cost. While a virtual environment cannot completely replace the actual equipment, practising maintenance skills in the environment lets students and instructors make better use of limited training opportunities on the actual equipment.

A virtual world as used in this trainer consists of a static visual background with which students cannot interact, and interactive objects which students can manipulate with the mouse.

The static background consists of the workshop which need to be represented visually for context and realism, but which do not need to be interactive.

For the time being students can see the VR world on the monitor using Desktop System. Students interact with the virtual world using a mouse and keyboard. Interactions consist of using mouse to pick and manipulate objects in the virtual world, and navigate within the virtual world.

5.5.2 The Virtual Training Environment

The virtual training environment is an interactive synthetic environment that can be explored, with the aid of a 3-D mouse as shown in Figure (5.4 and 5.5(a/b)) or the typical mouse. The user has the chance to restart the virtual world from scratch to start again if he had made any mistake, this is one of the virtual reality advantages.

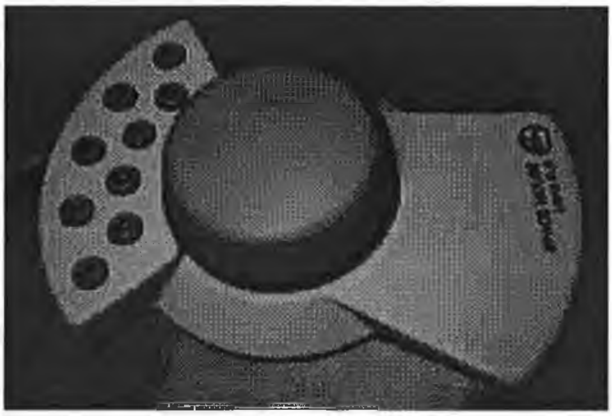


Figure 5.4 Typical 3-D mouse

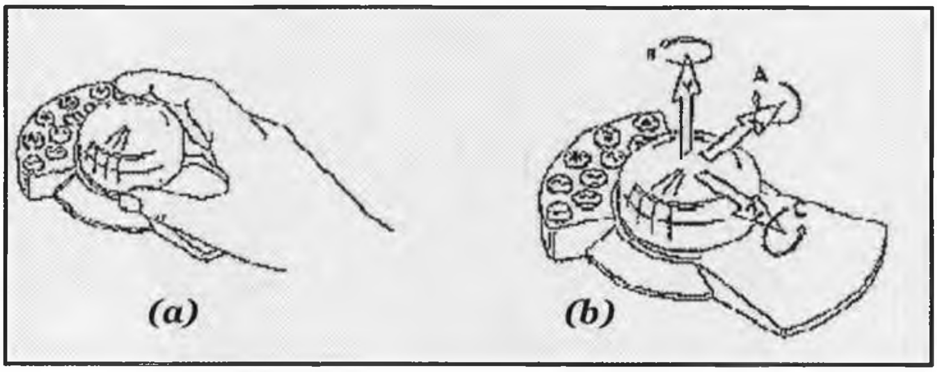


Figure5.5 six degree of freedom input device, (a) the Magellan space mouse & (b) axis labeling for the spacemouse

The Six DoF. (Quasi) isometric input devices such as the Magellan are commonly used to manipulate (translate and rotate) graphical objects in a three dimensional world.

The developed virtual world begins with a big sliding automatic glass doors which allows the end-user to see the lab through them as shown in figure 5.6. This laboratory was designed in such a way that extra training laboratories can be added at any time, by adding extra doors for each new laboratory. If the user clicks on any of the available mouse on the Milling Machine file, he/she is prompted on the screen with a message for example "***Welcome to the Virtual Workshop in Dublin City University***". On a mouse activation of the main door of the laboratory as shown above, the two doors will slide in **x-axis** to open /close.

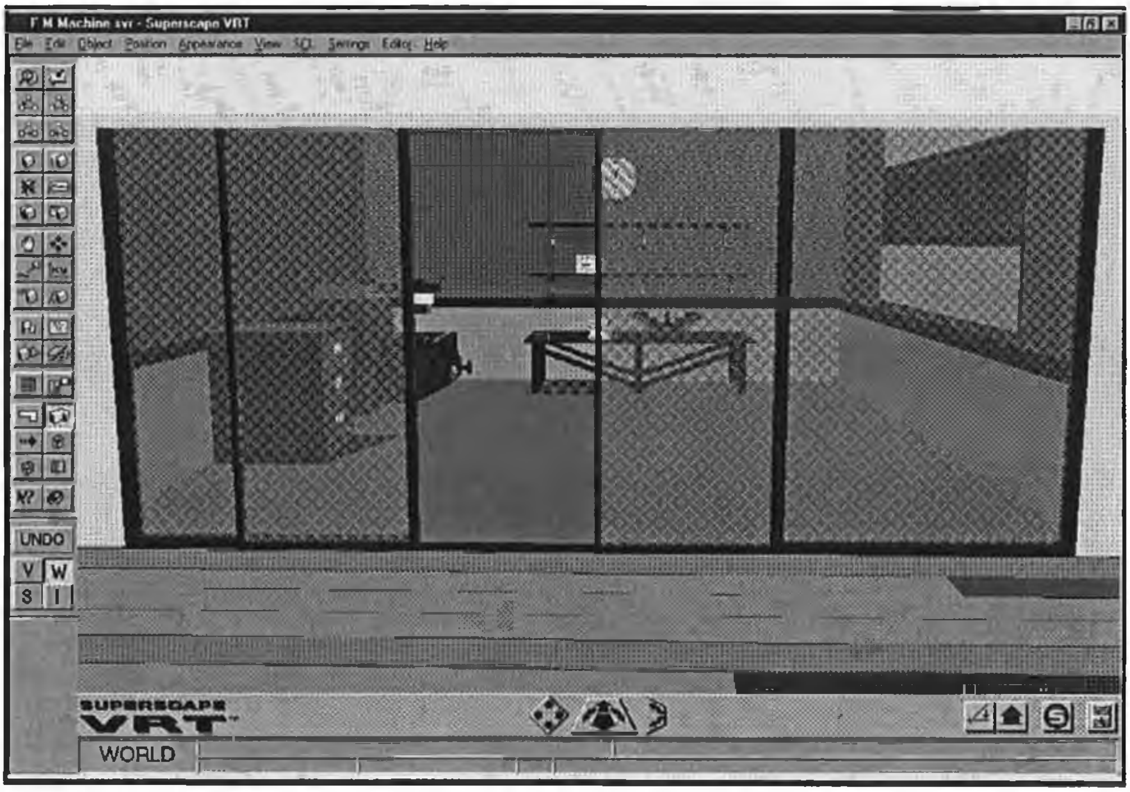


Figure 5.6 the Virtual Training Centre with left door opened

These doors have an SCL attribute that opens the file of the Virtual Milling Machine Laboratory when activated as shown in figure 5.7. When the user is inside the milling machine-training environment laboratory, he/she can view the milling machine from every location in the virtual world. The user can also start the machine and navigate every part in the machine if it is safe to do so.

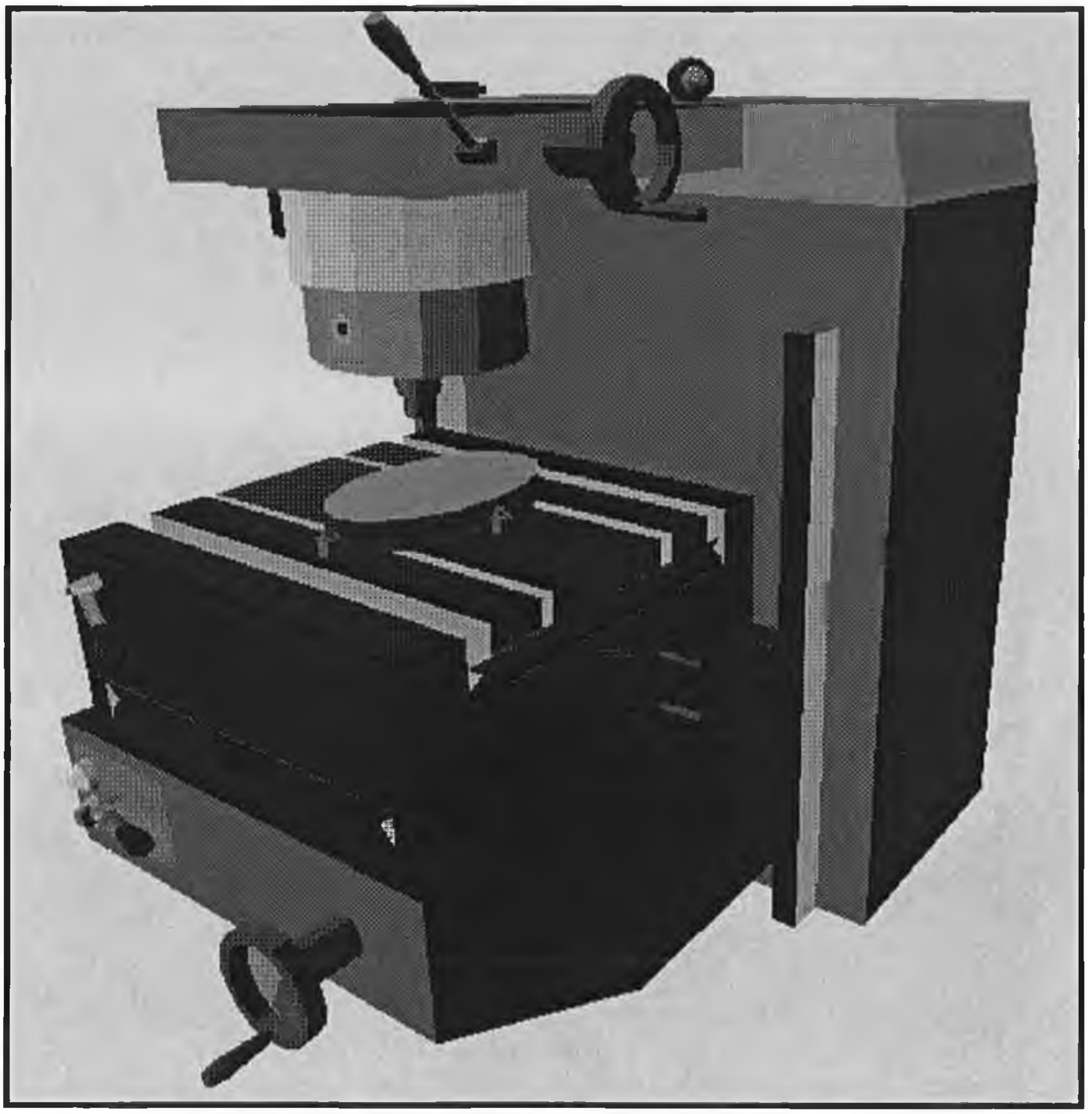


Figure 5.7 The Virtual Milling Machine Training Laboratory.

5.6 Model Qualification (Assumptions, Limitations & Features)

There is an accepted framework for developing models, which simulate real life systems. This framework comprises of three elements, namely *Qualification, Verification and Validation.* In order for a virtual model to represent a real system, a framework comprising of these elements should be taken into consideration during the model development process. The model should firstly undergo a model

Qualification procedure. (*Verification and Validation are the other two elements and are dealt with in more details later in this chapter*).

The qualification procedure is used for “determining the structure, elements and data of the model which are required to provide an acceptable level of agreement with the actual system”[46]. This following section presents the main features of the virtual model, the assumptions made, and the limitations imposed as a result of these assumptions.

1) The virtual machine start-up process mimics the start-up process of the actual machine. The sequence and conditions for safe machine start-up were documented. The virtual machine was then be programmed using SCL in order to comply with these conditions.
2) The emergency stop procedure will stop any operation. The virtual emergency stop button as well as safety lever will interrupt the ‘cutting tool’ sequence and render the machine inoperable.
3) The work-piece cannot be removed from the work-piece holder while the machine is running. The programme attached to the virtual work-piece will act in the same manner as the actual work-piece. Thus, the work-piece may only be removed from its place when it is safe to do so i.e. when the cutter is not rotating. The error message will pop up to tell the user certain message as shown in figure 5.8 (a).
4) The virtual world has two cutting tools; the end-user cannot change from one to another cutting tool, if the virtual machine is running. The programme that is attached to the virtual cutting tool will act in the same manner as the actual cutting tool. Thus, the cutting tool may only be removed from the spindle when

it is safe to do so i.e. when the cutter is not rotating. The error message will pop up to tell the user a certain message as shown in figure 5.8 (b).

5) The virtual machine has some elements which have axial movement such as the Knee, Table and Saddle when it reach the maximum limit of movement it will prompt the end-user that it has reached the end, which is the same as the actual machine

Some of the above features are related in that the model needs to know the state a number of other objects in order to take the appropriate action e.g. If the requirement is to remove the cutting tool or the work-piece, the spindle must not be rotating. If the spindle is rotating, the cutting tool or the work-piece cannot be removed and the model responds with an appropriate error message.

Figure 5.8 (a) The Error Message if clicked on the work-piece while the machine is running.

Figure 5.8 (b): The Error Message if clicked on the cutting-tool while the machine is running.

5.7 The Virtual Milling Machine

The Bridgeport Milling Machine was selected as a suitable machine to model as it is used extensively in the educational/training sector. This machine is a low cost-milling machine used for design work in technology education, particularly as a training model for inexperienced operators. The Virtual Bridgeport Milling Machine was modelled using AutoCAD Release 13. As in Figure 5.9 shows some of the features of the milling machine programmed in SCL language in order to add intelligence to the Virtual Machine Model.

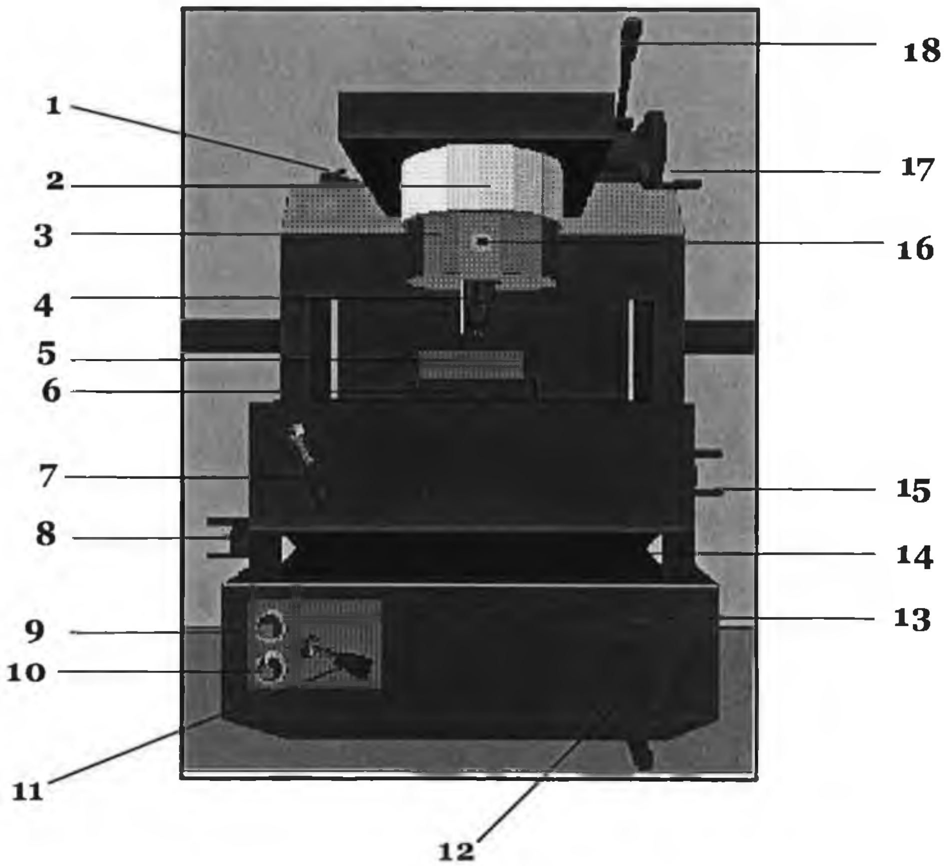


Figure 5.9: The AutoCAD R13 model of The Bridgeport Virtual Milling Machine then transfere to VRT .

This list describe the major elements of the Virtual Milling Machine as following

1. *Key clamping Cutting tool*
2. *Head Top Housing*
3. *Spindle Drive*
4. *Cutting-tool Loaded*
5. *Work-piece Loaded*
6. *Work-Piece Holder*
7. *Red Knob*
8. *Saddle Handwheel Inward/Outward screen*
9. *Power Switch Yellow/Green colour*
10. *Emergency Button Red colour*
11. *Spindle Switch On/Off*
12. *Knee Handwheel Up/Down movement*
13. *Knee*
14. *Saddle Base*
15. *Table Handwheel*
16. *Key Hole*
17. *Spindle Handwheel Up/Down movement*
18. *Spindle Brake and Lock lever*

List 5.1: *The Major Elements of the Virtual Milling Machine*

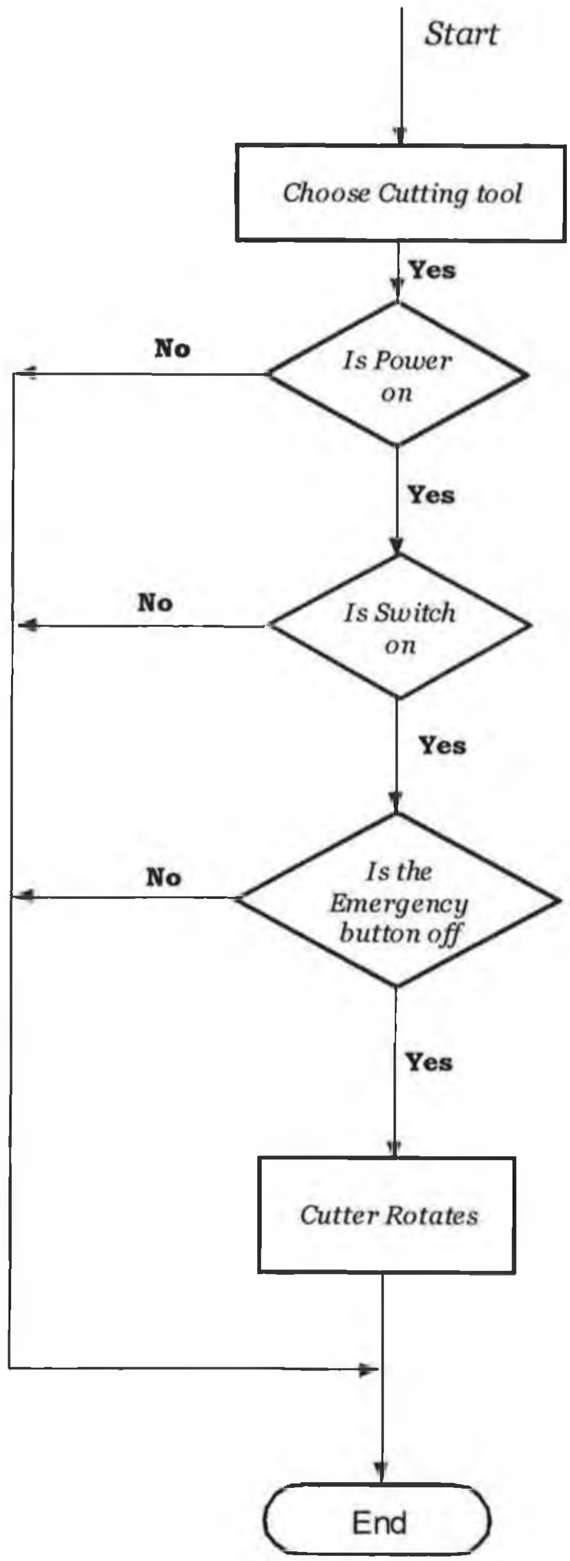


Figure 5.10: *Flowchart of On squence for Cutter Rotation*

5.8 The Virtual Training Application

5.8.1 Introduction

The visual presentation of the virtual training environment is described in this section. A selection of views is presented in order to give the reader a true perspective of the virtual lab and what a trainee may see on screen. A number of field trials were carried out in order to verify that the developed training application was beneficial to trainee operators. A questionnaire was completed by a number of participants who used the developed training application. The result of which were documented and conclusions were drawn from each answer. These conclusions are presented to the latter part of this chapter.

5.8.2 The User Navigation Tools

Navigating around the virtual lab is achieved via a movement bar located at the middle bottom of the screen. The movement bar is presented in figure 5.11 (a).

In order to navigate around the virtual lab, the user should use left mouse to click onto the appropriate icon in the movement bar i.e. if the user intends to move forward in the virtual lab he/she will click on the 'Move Forward' button and drag the mouse away from the icon in the direction he/she desired to travel.

The further the mouse is displaced from the icon, the faster the forward motion will appear to the user. In order to stop the motion, the mouse button will have to be released. The "looks" Up/Down icons give the effect of rotating the user head Up/Down and gives the appropriate visual response.

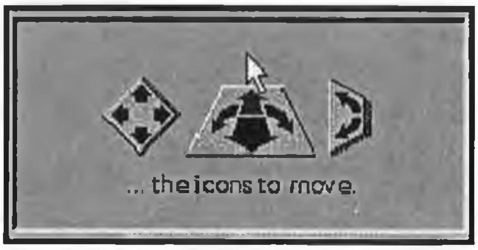


Figure 5.11 (a): The Mouse clicking and drag on the icons to move around the World

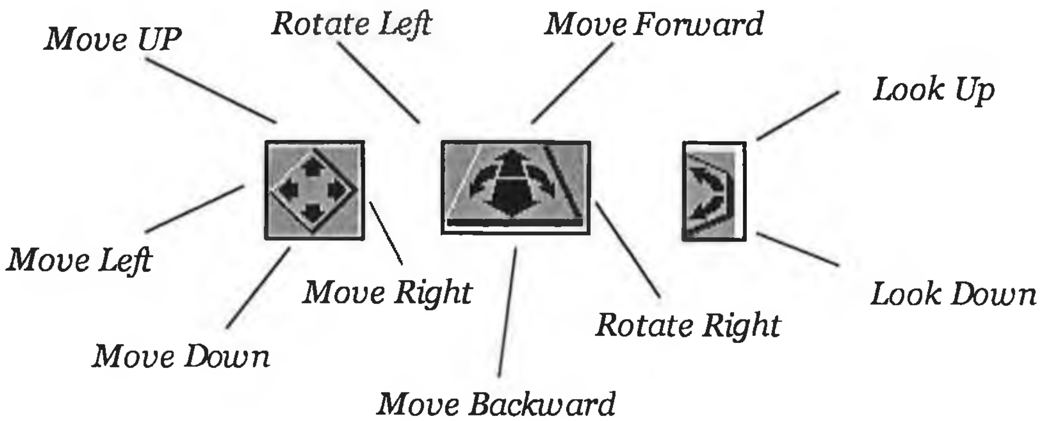


Figure 5.11 (b): The Navigate Arrows to move around the world.

5.9 Verification and Validation

The qualification procedure described early in section 5.6 is the first step in developing a simulation model. This model qualification procedure was undertaken in order to determine the structure and the elements of the virtual model, required to provide an acceptable level of agreement with the actual system.

Essentially this is a translation of the real life system into a conceptual model. Qualification is followed by a verification and validation procedure. The complete process is illustrated in figure 5.11

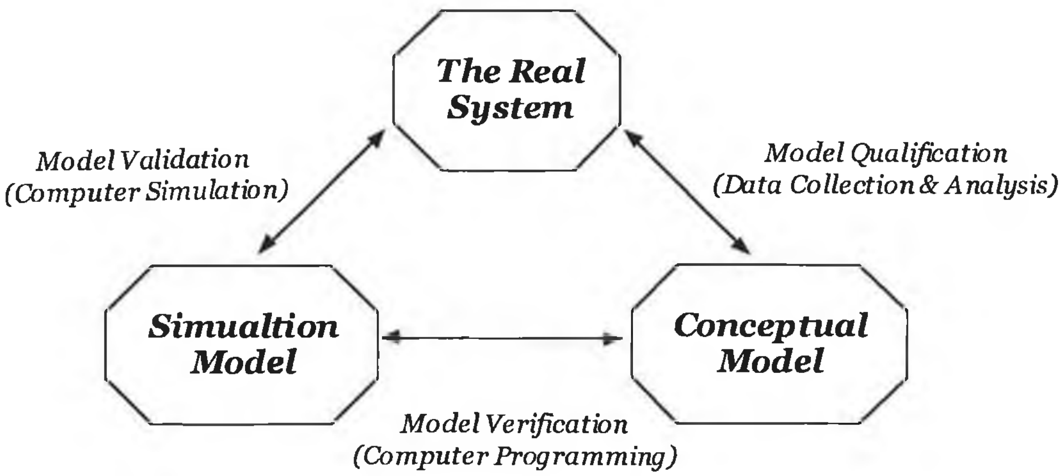


Figure 5.12: *Simulation model development process* [46]

Verification and Validation are carried out in order to ascertain whether a simulation model is a good representation of an actual system. For this model, field trials were undertaken, which went a long way towards verifying and debugging the model and served as a partial validation. The final piece of validation was to test the virtual model to see if the virtual machine responded in the same way as the actual machine.

"Verification refers to the process of the conceptual model has been correctly translated into an operational program, and the calculations made with this program, utilise the correct input data. Validation refers to the process of confirming that the conceptual model is applicable or useful by demonstrating an acceptable correspondence between the computational results of the model and the actual" [47]. Therefore, verification and validation are "both concerned with the problem of trying to determine whether the model is accurate" [46].

5.9.1 Verification

In order to ensure that the training application developed was both easy to use and interesting, a questionnaire was completed by **12** participants who took part in the field trial. The field trails acted as both verification and partial validation of the model. By becoming the first users of the model, the participants were verifying that the model made sense and followed an acceptable or expected logic, thus assisting with its verification. The trials also helped to verify that the model had been properly debugged. Partial validation was reflected in the fact that those who had some experience of using the real life machines were able to evaluate how well the model represented a real system.

The trials were conducted in order to evaluate what the participants thought the benefits of using the developed application, as a training tool would be. The participant consisted of lecturers, engineers and students. Each participant used the virtual training application and navigated around the virtual lab with the aid of a stage manual. Each participant then completed a questionnaire. This questionnaire from the study which was undertaken by reference [2]. This questionnaire related to each user's opinion on how the developed application may benefit novice operators.

Only two questions actually tested the participant on a working characteristic of the machine, the remaining questions being solely concerned with the participant's subjective opinion of the application. One type of computer was used during the evaluation process "***Pentium 200 with 64 MB RAM, 17" Monitor***".

It should be noted that the developed training application was initially designed using a Pentium 133 with 32 MB RAM. A significant difference was noticed using the latest computer hardware available at the time of the evaluation.

5.9.2 Validation

Testing was conducted into two parts. The creation of Bridgeport Milling Machine in virtual world and the ability of the machine to operate after SCL programme were attached to each object of the virtual machine. To test the first part, the world was switched to plan and elevation views in turn. By moving the world up and down left and right with the Spacemouse to check for correct rendering and sorting. If any overlapping of objects occurred, rectification was then carried out. Trial and error would be necessary until no further overlapping of objects were found. Secondly, SCL programmes were attached to the objects concerned. Testing was completed if the selected objects functioned correctly. Otherwise, some changing and manipulation were performed to modify the SCL until the satisfied results were achieved. The same procedure was repeated for other objects.

5.10 Field Trail Results

Once the questionnaires were completed, collected, each answer was carefully documented, and an appropriate comment was attached to the degree of satisfaction attained by the participant using the virtual training application. A complete listing of the comments associated with the questions in presented in Appendix D. As can be seen from the questionnaire (appendix D) a range of answer/comments may exist for some of the questions. It is purely up to the participant if he/she agrees with a statement and space was allocated for additional comments if required. The participants were firstly asked to fill their name, sex and age group in order to obtain a true description of the types of participants that undertook this exercise.

e.g.
Name: Mr. Virtual Computer
Sex: Male/Female
Age: 100

a) 15-20
b) 21-25
c) 26-30
d) Over 30

Student Response:

Answer	**a**	**b**	**c**	**d**
No. of Participants	0	1	7	4
Percentage	0 %	8 %	58 %	33 %

Comment

The majority of the participants that evaluated this training application consisted of Post-graduate students in Mechanical Engineering School. Some of the other participants consisted of totally unrelated professions.

Part 1

Questionnaire Statement:

Previous operator experience with a Vertical Milling Machine (**V.M.M**):

a) None.

b) Basics.

c) Intermediate.

d) Expert.

Student Response:

Answer	a	b	c	d
No. of Participants	3	7	2	0
Percentage	25 %	58 %	17 %	0 %

Part 2

Questionnaire Statement:

The Z-axis on a milling machine moves

a) Right-Left

b) Left-Right

c) Up-Down

d) Backwards-Forwards

Student Response:

Answer	a	b	c	d
No. of Participants	1	0	11	0
Percentage	8 %	0 %	92 %	0 %

Part 3

Questionnaire Statement:

Basic milling machines move in three directions, which are called:

a) Up, down, back.

b) P axis, Q axis, R axis.

c) A axis, B axis, C axis.

e) X-axis, Y-axis, Z-axis.

Students response

Answer	a	b	c	D
No. of Participants	1	0	0	11
Percentage	8 %	0 %	0 %	92 %

Part 4

Questionnaire statement:

Was the virtual milling machine difficult to operate?

a) It was difficult to follow because …

b) Some part were hard to follow and understand

c) All parts were easy to follow and understand.

Student response:

Answer	a	b	c
No. of Participants	1	3	8
Percentage	8 %	25 %	67 %

Part 5

Questionnaire Statement:

Did you find this training method interesting?

a) Not interesting because.

b) Some parts interesting.

c) All parts interesting.

Students response:

Answer	a	b	c
No. of Participants	1	2	9
Percentage	8 %	17 %	75 %

Part 6

Questionnaire Statement:

Do you think you would find what you learned easier using a book?

a) Yes because

b) No because

Students response:

Answer	a	b
No. of Participants	3	9
Percentage	25 %	75 %

Part 7

Questionnaire Statement

Do you think this form of training application is beneficial to first time learners?

a) Yes because

b) No because

Students response:

Answer	a	b
No. of Participants	11	1
Percentage	92 %	8 %

Part8

Questionnaire Statement:

If this training application was used at the early stage of training an operator. Do you think this would reduce the risk of a dangerous accident?

a) Yes because

b) No because

Students response

Answer	a	b
No. of Participants	10	2
Percentage	83 %	17 %

Part 9

Questionnaire Statement:

Were the navigate instructions clear and easy to understand?

a) Yes because

b) No because

Students response

Answer	Yes	No
No. of Participants	11	1
Percentage	92 %	8 %

Part 10

Questionnaire Statement:

Was it easy to navigate throughout the virtual worlds?

a) Yes because

b) No because

If no, please describe the difficulty(s) you experienced, and where in the program the difficulty was experienced.

Students response:

Answer	**Yes**	**No**
No. of Participants	8	4
Percentage	67 %	33 %

Part 11

Questionnaire Statement:

Was it easy to identify the various components of the milling machine?

a) Yes because

b) No because

If no, please describe the difficulty(s) you experienced.

Students response:

Answer	**Yes**	**No**
No. of Participants	10	2
Percentage	83 %	17 %

Part 12

Questionnaire Statement:

In your option, will Virtual Reality be an effective training tool for learning how to operate a milling machine?

a) Yes because

b) No because

If no, please explain why.

If yes, what do you feel makes it an effective learning simulation?

Students response:

Answer	Yes	No
No. of Participants	11	1
Percentage	92 %	8 %

Part 13

Questionnaire Statement:

Did you enjoy the exercise?

a) Yes because

c) No because

Students response:

Answer	Yes	No
No. of Participants	12	0
Percentage	100 %	0 %

5.11 Results and Discussion

The Bridgeport Milling Machine has been created successfully in the virtual world. With no sorting and rendering problems found, the world was shown perfectly from all viewing angles. Therefore, a realistic of the machine has been achieved.

By activating the buttons on the objects directly, objects with SCL attached to them were able to function as for the real machine. For example, if the Knee handwheel was activated, the knee will move up and down corresponding to the direction of the handwheel rotation. Similarly, if the Saddle handwheel was activated a saddle movement will be achieved. Pressing a control button could preview operation. The machine would begin to operate if another button was activated. Other capabilities such as tool change work-piece loaded, and safety lever active is available.

Shapes were created by defining points within the bounding cube, and facets were created by joining the points together. Facets created must be defined in the correct orders; otherwise, it could cause rendering and sorting problems in the World Editor.

Modification of shapes must be carried out in the Shape Editor if required changing the co-ordinate points within the bounding cube did this. Therefore, it was worth to create a new shape if it involved too much change.

If half section of a shape was created, in order to make the other symmetrical shape, this object must be duplicated and given a new shape name, the size of its bounding cube must be doubled. The new shape was transformed or rotated, then was moved to its new position within its bounding cube. This shape was finally merged with the previous shape to form the complete shape. Notice that the Shape could only be transformed or rotates through 90° degree. Therefore, only a quarter or half shapes were allowed if shapes merging was required.

If an object is needed to perform animation in the world, its animated cells must be predefined in the Shape Editor.

Hole facet could be created on another facet. In order to look through inside, an inside facet must be created. Although it was a hole, no insertion of other object was allowed or rendering problems would result.

Only one shape could be created at a time. A new shape name must be given for each duplicated. When a shape was completed, it was stored in the Shape Files. Shape Files consisted of the basic shapes Cube and Group, plus the other shapes that were created.

The first object created was simply positioned in the world. Without setting a reference point, it was realised later that a great deal of calculations was required to determine the new position for each object that was created.

Shapes that were created in S.E. might not be the same size as the objects created in W.E. Sometime it was unavoidable to re-size the objects so that they could be assembled correctly in the world. However, this has led to the change to the actual dimensions. To compromise this, the size of objects was acceptable if their proportionality was maintained.

When the completed Bridgeport Milling Machine was viewed for the first time. Overlapping of objects were found as the world was moved around. The reason was unknown, trail and error were carried out extensively but the problems remained unsolved. Sometimes it was just able to fix these objects, while other objects would appear to overlap.

Considerable amount of time was spent in finding the actual causes. Manuals were studied repeatedly. Examples supplied in this software were analysed in depth. It was recognised at last that these problems were due to incorrect grouping of objects, resulted in sorting and rendering difficulties. Nevertheless, this was not enough to solve the problem, mainly due to the compactness of this machine. Which objects should be grouped under which group was still a question mark? Trail and error were carried out until the satisfactory results were obtained.

For example, moving object of the area should be grouped together. Smaller objects should be grouped together. If an object needed to be attached to other object, these objects must be grouped together and became siblings between each other's etc. Still there were no fixed rules to group the objects together and experience was gained through practice.

As a rule, each group should contain only 10-15 siblings, more sorting time was required if over this limit. However, this rule did not apply to the 'group' that held all the 'Sub-groups' together.

It was also found that when the world was checked for correct sorting and rendering, an object which was not obscured in any view point should be selected or else these problems would never be settled.

All works must be saved for each update because the computer could crash, as the world became more complex.

It was simple enough to write a programme such as "if activate me, visible me or else invisible me". However, it was rather difficult to write a program, to instruct objects by giving them condition.

A list of operating sequences was listed before any attempt was made to write any programme. The same programmes were written in a different way. If the first programme failed to work, the next one was tried and so on, until the desired results were achieved. The Manual or other examples were always referred to if in doubt. Most of the times, when SCL was attached to an object, undesired result were often obtained. When a programme was thought to be absolutely, perfect but it did not work at all. With a lot of patience and efforts the right SCL was written and machine at last was able to perform accordingly, although not perfectly.

5.12 Conclusion

This chapter introduced the concepts behind the development of the virtual training application. The developed system has been described from the user's point of view. It describes how a user may navigate around and interact with various objects in the virtual laboratory. Also described some basic general description about Bridgeport Milling Machine. A number of 12 participants completed a virtual training tutorial using the developed virtual milling machine. Each participant completed a questionnaire relating to the ease of use and practicality of using such an application to train milling machine operators. The results shown in section 5.10 show a high satisfaction rate with the training application developed i.e. 100% of participants enjoyed the exercise and 92 % believed that VR will be an effective tool for training milling machine operators...

Chapter Six: Conclusion and Suggestion for Further Work

6.1 Conclusion

This thesis has presented work undertaken in the area of industrial training and the creation of a Virtual Reality (VR) training application for milling machine operations. Creating a Bridgeport Vertical Milling Machine world has been carried out successfully. The machine was able to operate in a sequence replicating the real machine. Therefore, training of operators to use the Bridgeport Milling Machine could be accomplished. A windows based application was developed using the latest in non-immersive VR software.

Using the developed training application, a trainee Bridgeport Milling Machine operator may do the following interaction with the virtual world:

- √ Fully user defined interactive viewing from any direction.
- √ Navigate around a virtual milling machine with the aid of a simple computer mouse.
- √ Ability to move and 'look-around' the machine.
- √ Interact with the virtual milling machine, practice various operations, and learn the basic functionality of the machine.

√ Gain a familiarity with a commonly used machine in industry

√ Ability to move backward/forward, left/right, and up/down of the moving elements, preview programme, change the tool orientation loads the work-piece.

Although a lot of improvement needs to be done as detailed in recommendation for the Bridgeport Milling Machine, the basic requirements for an interactive training demonstration has been achieved. As well as the benefit of the application of VR for training and familiarisation have been proven.

Other than creating a Bridgeport Machine, Virtual Reality could be used to create other machines such as Lathes or CNC milling machine. Training of operators would no longer be restricted in the workshop. Training could be carried out in the classroom, office and even at home. It is easier and more economical to train operators by using computers. Overall, efficiency can be increased since there is no restraint in materials, time and training venue.

Grouping of objects was very important for effective sorting and rendering. In the world Editor, objects size was modified from their actual size in order to be assembled correctly. There were some limitations when creating shapes. For example, only quarter or half shapes could be created if the objects was to be merged together.

Converting DXF files to VRT involved a great deal of calculation and processing, as the DXF file did not contain much of the specific information required by VRT. The missing information must be generated by the converter and only could be obtained by calculated guesswork. Therefore, it was recommended for use on single objects, which then are assembled using the VRT editors.

The developed system has a number of benefits for first time Vertical Milling Machine operators:

- It offers a unique learning experience, which helps retain the student's attention longer than traditional learning methods such as reading a manual.
- This training application is very interactive.
- It increases the level of safety for first time users.
- Various cost savings can be achieved. In particular, the machine does not have to be taken out of production for initial training of new operators. Operators may learn at their own time and once the training application is purchased it can be used repetitively at any time.

6.2 suggestion for Further Work

Further work could be carried out to link this present study to a real more comprehensive application of VR into educational and industrial training programs. Some suggestions for further work are listed as follows:

(i) Incorporate the features of material removal and chips formation into the virtual machine. In addition, to look at the system ability to measure the products with a measuring device that is available in the virtual world.

(ii) Adding sound to the world could further enhanced its realism, the users could generate the deep sense of feeling of immersion in the virtual environment, hence in believing that they were actually in the real world. Also, create an online audio and video training files for improving the learning capability of the student and trainee.

(iii) To develop and maintain an interactive dynamic homepage for machining process using **HTML** and **Java** scripts Languages.

(iv) In virtual environment, using simplified physical constraints could provide simple and realistic behaviour of objects. These included constraints such as gravity, friction, pushing and anchoring of objects.

(v) Develop more Virtual machines towards complete virtual manufacturing using the concept of concurrent engineering.

(vi) Development of Internet based virtual machine for online distance learning.

References

References:

[1] *John Vince, "Virtual Reality Systems", Addison-Wesley Publishing Company, ISBN 0-201-87687-6, (1995).*

[2] *Damin O'Sullivan, "Virtual Reality Modelling of a CNC Machine" Department of Mechanical & Manufacturing Engineering, Cork Institute of Technology, (1998).*

[3] *Chong, P. L., Abd. Magid Hamouda "Development of Virtual Reality Labless for Engineering Mechanics Part I" Department of Mechanical & Manufacturing Engineering, University Purta Malaysia*

[4] *Vance J.M., "Virtual Reality: What Potential Does It Hold for Engineering ?", Current Advances in Mechanical Design & Production, Sixth Cairo Universtiy International MDP Conference, Cairo, Jan 2-4, (1996)*

[5] *Rheingold Howard, "Virtual Reality", Simon & Schuster, (1991)*

[6] *Mazzola John, "Had CAD/CAM/CAE Met its Expectations? What Does The Future Hold?" Keynote Speech, MSC World User's Conference, University City, CA, May 8-12, (1995).*

[7] *Deitz Dan, "Impact Codes for The Virtual Laboratory", Mechanical Engineering, Vol.117, No.5, pp 66-70, May (1995).*

[8] *Dvorak Paul, "Planning for Peak Production", Machine Design, pp 124-127, March 9, (1995).*

[9] *Mahoney Diana Philips, "Driving VR", Computer Graphics World, pp. 22-33, May (1995).*

[10] *Louka, M., "An Introduction to Virtual Reality", (1996) http://home.sol.no/~mlouka/vr*

[11] *K. I.Bakkar, M. Iqbal, M.S.J. Hashmi "An Overview of Virtual Reality in Mechanical Design Engineering", U.P.M Advances in Materials and Processing Technologies, SIRIM Press, Vol. 2, pp 1147-1155, 24th –28th August (1998).*

[12] *Adam John A, "Virtual Reality is for Real", IEEE Spectrum, pp. 22-29, October (1993).*

[13] *Chris Hand, and Author Jerry Isdale Published on World Wide Web http://www.cms.dmu.ac.uk/~cph/VR/whatisvr.html*

[14] *Wilson, J., (1996), "Virtual Environments and ergonomics: needs and opportunities", VR applications research team, University of Nottingham*

[15] *Wilson, J.R & Cobb, S.V.G & D'Cruz, M.D.& Eatgate, R.M., (1996), "Virtual Reality for Industrial Applications: opportunities and limitations", Nottingham University press, Nottingham.*

[16] *Caird, J.K., (1996), "Persistant Issues in the application of Virtual environment Systems to Training", IEEE Computer Society: Interaction with Complex Systems. Vol. 3, 124-132*

[17] *Zeltzer, D., (1991), Task-Level Graphical simulation: Abstraction Representation, and Control. In Badler, N. (ed.) Making them move: mechanics, control, and animation of articulated firgures. Pp3-33*

[18] *Steve Bryson, "Approaches to the successful design and implementation of VR applications", R. A. Earshaw, J. A. Vince, H. Jones, "Virtual Reality Applications", PP 3-15, ISBN 0-12-227755-4, ACADEMIC Press (1995).*

[19] *Rahman, T., "An examination of virtual reality and virtual worlds technology applictions" Unpublished paper, MSC Information Systems and Technology, (1993).*

[20] *Masson Robert, "Parallel and Almost Personal", Machine Design, pp. 70-76, April 20, (1995).*

[21] *Roy Train M., Cruz-Neira Carolina and Defanti Thomas, "Cosmic Worm in the CAVE: Steering a high-performance Computing Application from A Virtual Environment", Presence, Vol. 4, No. 2, pp. 121-129, spring, (1995).*

[22] *Virtual Prototype Figure.*

[23] *Hancock Dennis, "Prototyping the Hubble Fix", IEEE Spectrum, pp 34-39, October (1993).*

[24] *Macedonia Michael R., Zyda Machael J., Pratt David R., Barham Paul T., and Zeswitz Steven, "NPSNET: A Network Software A architecture for large-scale Virtual Environments", Presence, Vol. 3, No. 4, pp. 265-287 ,Fall (1994).*

[25] *Pausch, R., Crea, T., "A literature Survey for Virtual Environments: Military Flight Simulator Visual Systems and Sickness" Computer Science Report No. TR-92-25, Virginia Centre for Innovative Technology, August (1992).*

[26] *Rebecca Louise Belyea "An Amateur's Analysis of Virtual Reality" published on World Wide Web http://www.people.virginia.edu/~rlb9h/vrpaper.html*

[27] *Dysart, Joe VR: A $6 Billion Industry By 1999 http://www.univ-paris8.fr/~loic/News/sci/virtual-world/0000000050 Apr. 8,1995(Accessed 4/19/96)*

[28] *Mikko, K., Marko, S., Mako, H., "Virtual Reality Prototype- a Framework for the Development of Electronics and Telecommunication Products", www.ele.vtt.fit/projects/vrp/vrp.html.*

[29] *Roche, F. & Tansey, P., "Industrial Training in Ireland; Study Prepared for 'The Industrial Policy Review Group', The Stationary office (1992).*

[30] *Seymour, E., "Skills analysis Training", Pitman London (1968).*

[31] *Heath, E.D, "Training in the Work Place: Strategies for improved Safety and performance", Aloray Inc. Professional and Academic Publisher (1990).*

[32] *Bentley, T.J., "The business of Training", P.33, McGraw-Hull Book Company (1991).*

[33] *Barnet, J., Graham, L., "Training for the small Business": How to make the most of all the opportunities, Kogan Page, (1992).*

[34] *Millhiem, W.D., "Virtual Reality and its Potential applications in education and theory", Machine Mediated Learning, Vol. 5, pp 43-65.*

[35] *Conkal, G.K, "The Role of CAD/CAM in CIM: The User Perspective", SME, (1994).*

[36] *Lynch, M, "Managing Computer Numerical Control Operations": How to get the most out of your CNC machine tools, Society of Manufacturing Engineers. (1995)*

[37] *The Manual of VRT for Windows User's Guide; Superscape, (1996).*

[38] *John Vince, "Virtual Reality Systems", Addison Wesley, and ISBN: - 0-201-87687-6,(1995).*

[39] *A. Yarwood, "An Introduction to AutoCAD Release 13", ISBN 0-582-27450-8, Longman Group Limited 1995*

[40] *AutoCAD Release 13 User's Guide Manual, and ISBN 00105-000000-5020, February 1, 1995 AutoDesk Inc*

[41] *3 D Studio Max User's Guide Manual, and ISBN 12801-000000-5020, March 1996, AutoDesk Inc*

[42] *Jim Lammers, Michael Todd Peterson, "3D Studio For Beginners", and ISBN 1-56205-419-8, New Riders Publishing 1995.*

[43] *Bruce J. Black, "Workshop Processes Practices, and materials", Engineering drawing for technicians Vol. 1, Edward Arnold, ISBN: 0-7131-3409-7, (1979).*

[44] *W. A. J. Chapman, "Workshop Technology: Part 2", Edward Arnold, ISBN: 0-7131-3272-8, (1972).*

[45] *Bridgeport Machines Limited, SERIES ONE, "Vertical Milling Machine: Operation and Maintenance Manual" (1996)*

[46] *Wu, B, "Manufacturing Systems Design and Analysis", Chapman and Hall, (1992)*

[47] *Schlesinger, P., "Developing Standard Validation and Verification Procedures", Winter Simulation Conference (1984).*

[48] *David M. McLin & James C. Chung "Combining Virtual Reality and Multimedia techniques for effective maintenance training", Center for Digital System Engineering Research Training Institute.*

[49] *M. Iqbal and M. S. J. Hashmi "Production Facilities Planing in Virtual Environment", U.P.M Advances in Materials and Processing Technologies, SIRIM Press*

Appendix A

Basic Construction of VR

Basic Construction of VR

The Editors

VRT contained seven editors, which could be used to create virtual worlds, and control how the end-user interacted with them. The Shape Editors (*SE*) and the World Editor (*WE*) were the most important editors, in which the shape and objects were created to make up the world. The Layout Editors and Resource Editors were used to customise the computer screen. The Image Editor and Sound Editor were used to import and create pictures, textures and sounds to enhance the appearance of the world. The keyboard Editor was to provide additional information and interaction for the user.

Visualiser

Visualiser was an application that enables the users to display and interact with the virtual worlds that are create using the VRT editors. All basic movement and interaction with worlds could be made from the mouse, Spacemouse or Keyboard.

Superscape Control Language (SCL)

A simple but powerful C-based programming language, that allowed the user to attach a vast range of conditional attributes to an object. These include a high level of ‘Intelligence’ so that they react and inter-react with the end-user and other objects in the world, which will be discussed in paragraph 4.4

Data Converter

Data converter includes in VRT was for importing and exporting data files from many other applications. VRT includes a module for importing the popular DXF format, as used in many CAD packages.

Objects in Superscape's

VRT can be given a range of dynamic features that mimic the dynamics of objects in the real world. They include gravity, fuel, climbing, falling, friction, and restitution, driving velocity, maximum velocity, angular velocity and whether an object can be pushed or not. Time can be defined in terms of absolute time, or based upon the scene update rate [37,38].

Gravity

Gravity is entered as units which alter an object's Y-velocity at a rate of g per frame^2 downwards (where g is the acceleration due to gravity), thus accelerating the object towards the ground. In practice, g is given a default value of 98 *mm per frame^2 for a frame update rate of 10 frames per second.*

Fuel

Fuel is an attribute that can control an object's behaviour. For example, using Superscape's control language SCL, an object's fuel attribute can be used to determine the intensity of a light source, or the velocity of an object. Moreover, if the attribute is reduced over time, this in turn will attenuate the light source or slow the object down.

Climbing

The climbing attribute specifies how high an obstacle is allowed to be before a collision occurs and restitution is applied. An object, for example, may be moving in a straight line on a horizontal surface. If it has a climbing value of 20 and meets an obstacle on the same surface which is only 10 units high, then the moving object will rise 10 units over the obstacle and continue with none of its velocity values changed. If the object hits any obstacle that rises more than 20 units above surface, then this is interpreted as a collision.

Falling

The falling attribute specifies the maximum distance that an object can fall without its being 'damaged'. If an object falls by a distance greater than that specified by the falling value, then it is stopped and flagged as having fallen '*tool far*'. This can be accessed with the control language SCL.

Friction

In VRT, friction is an attribute of a moving object rather than of the virtual surfaces, it moves over, and it is expressed as the percentage reduction in the object's external velocity per frame on the horizontal plane.

Restitution

Restitution is associated with collisions between objects. Basically, when objects collide, their velocity is changed. When hitting a wall, for example, a car will come to a halt, but a ball will bounce off. The amount of 'bounce' in a collision is measured by the restitution attribute. Superscape's VRT permits an object to be given horizontal and vertical restitution attributes. When an object hits a surface, the object's current velocity is multiplied by the restitution, and the direction of the velocity is reversed.

Driving Velocity

The driving velocity is a constant velocity assigned to objects as they move about the VE. It is unaltered by friction, and is associated with one or a combination of the *X*-, *Y*-, or *Z*-directions.

External Velocity

An object's external velocity is acquired through external events such as collisions with other objects.

Maximum Velocity

The maximum velocity parameter restricts the velocity of an object to some upper limit.

Angular velocity

The angular velocity updates an object's rotation attribute and turns the object through a given angle every frame.

Pushable

The pushable attribute allows an object to respond to collisions with other objects. Thus, walls, floors and ceilings would have this attribute switched off. Apart from the above physical attributes, Superscape's VRT includes facilities for animations, bending, viewpoints and paths.

Animation

Preparing a sequence of states that show the object in different states animates an object. At run time, these are activated at a specified rate, resulting in a smooth animation. Sequences can consist of a linear pass, a cyclic repeat, a bounce (the animation runs forwards and then backwards), include pauses and combinations of all the modes.

Bending

Bending, like animation, alters the position of vertices in an object, which in turn the position of facets. The bend is made about an existing point, and is effected by selecting a collection of points in the object and specifying a bend angle.

This is shown figure A.1 part (***a***) shows the initial position of a shape; part (***b***) shows how a group of vertices are rotated; and part(***c***) shows how a second bend is introduced.

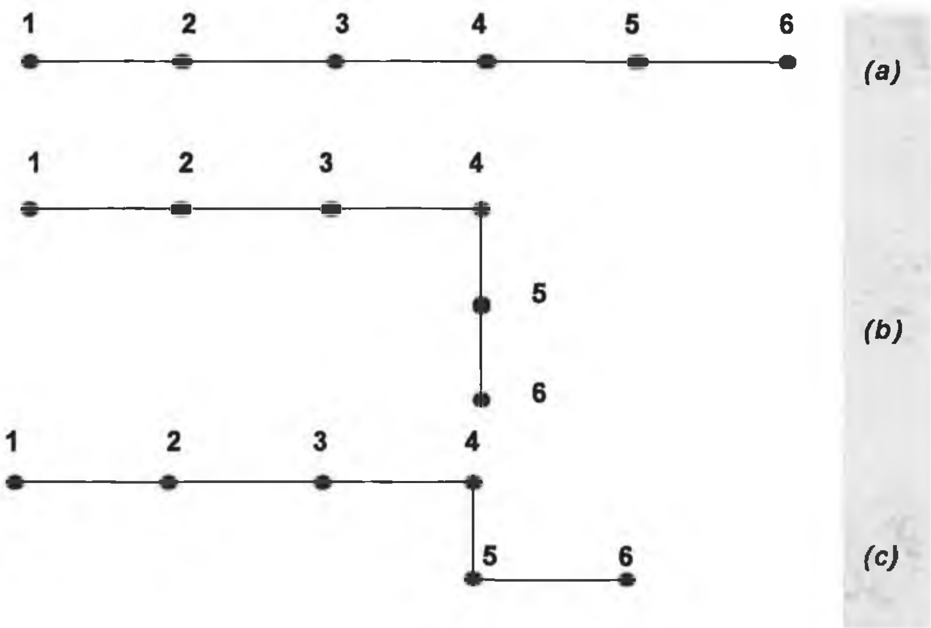


Figure A.1 An object can be bent about one of its vertics to create new shapes. (a) The original object. (b) Bending part of the object about vertex 4. (c) Bending part of the object about vertex 5.

Viewpoints

As the Virtual Observer (VO) is treated as an object in the VE, and controls the viewpoint, it is a simple process to attach the viewpoint to any other object. For example, a VE might have one viewpoint attached to, and controlling, a moving car. This can be steered using a Spaceball, while the viewpoint provides a view as seen from the driver's seat. Other viewpoints can be established external to the car that are either moving with the car or stationary.

Viewpoint paths

A viewpoint can be moved through a VE along predefined path. The path is a closed loop, defined by a few set positions of key frames. The viewpoint moves between the key frames at a speed set by the number of frames between them. At the end of the sequence, control returns to the beginning. Each key frame can define the position of the viewpoint, its rotation, or both. There are several different options for each of these. Positions may move along a straight line between key frames or they may follow a smooth curve through them. Rotations may be set relative to the rotation of the object, to which the viewpoint is attached, targeted to look at another object, or looking forward along a defined path.

Shape Editor

The Shape Editor (*SE*) permits the designer to create three-dimensional shapes using points and facets, which are used to give a profile of the objects to be used in the virtual world.

An object is called 'Shape' in the Shape Editor. A shape consists of combined facet and point information contained inside a bounding cube. Defining points in three-dimensional space within bounding cubes are first created. These points are connected together to form facets, in which they are grouped together to create shapes.

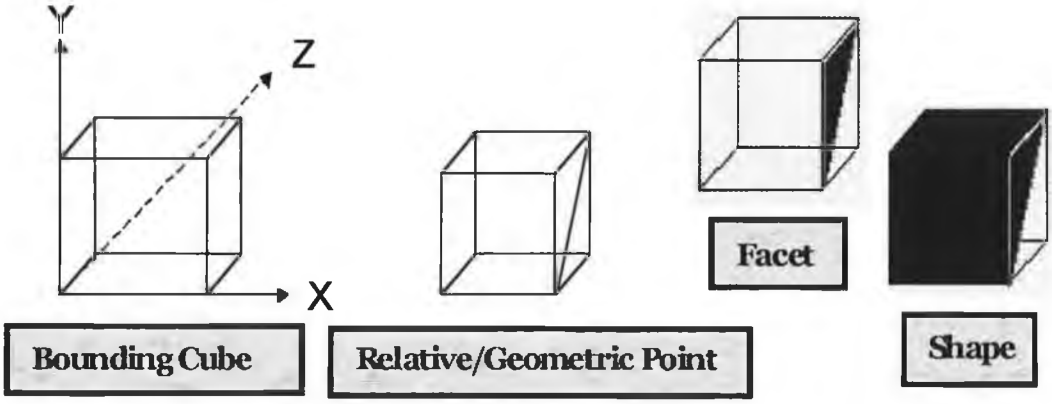


Figure A.2 Points and Facets Grouped together

The bounding cube is an invisible cube, outlined with *x*, *y*, and *z*-axes in the Shape Editor, used by VRT to sort objects within the world. In the virtual world, VRT orders objects, created from shapes, by position of their bounding cubes. Therefore, to create convincing contacts between objects it is important that a shape's bounding cube *x, y, and z* proportions should match the outer edges of a shape whenever possible and that *all points and facets are within this cube.*

Generally, the Shape Editor is used mainly to create complex shapes. All points and facets created are within the bounding cube regardless of the complexity of the shapes. Simple shapes, which have an appearance of cubes, are not normally created with this method. Instead, they are created in World Editor by selecting the default shape-cube, then are re-set to the required attributes.

Similar object could be duplicated, re-sized and repositioned conveniently as shown in figure A.3

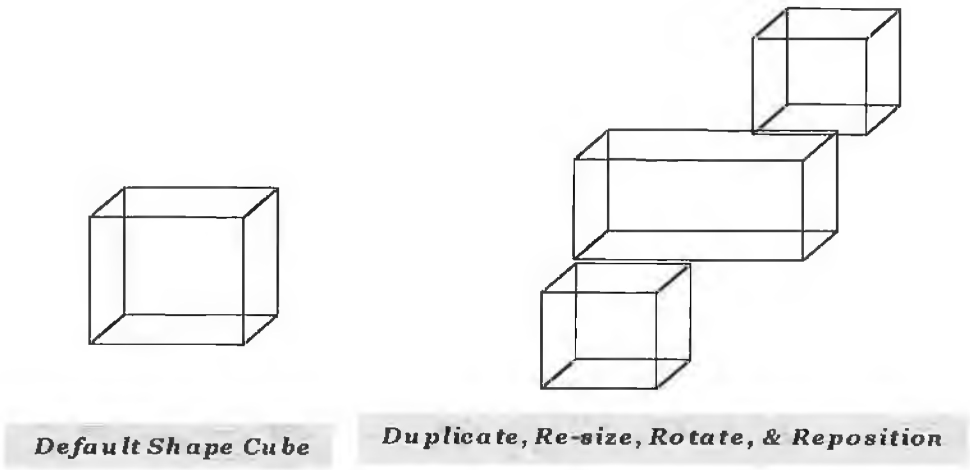


Figure A.3 Duplicate, Re-size, Rotate, and Reposition Shapes

Facets are surfaces created by connecting points together. However, facets containing indents or concave areas are not legal facets and should be avoided as they create rendering difficulties.

A T-shape facet say, with concavity should be built up from several facets, i.e. to make T from a single vertical facet and a single horizontal facet.

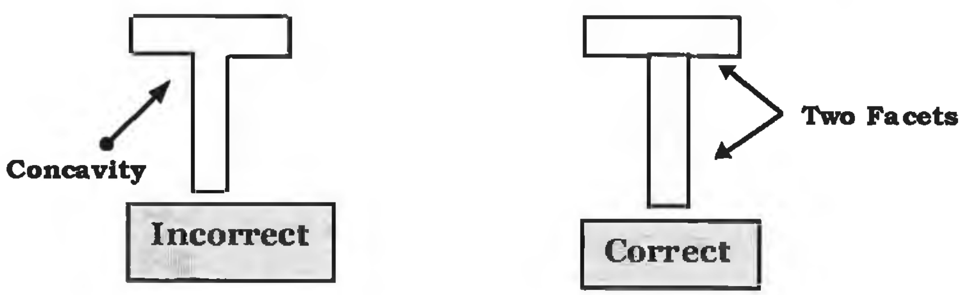


Figure A.4 Correct & Incorrect Facets

The order of facets within a shape is important in producing a convincing shape. Facets at the back of shape must be rendered first so that they do not obscure facets at the front of the shape. Sometimes it is unavoidable to produce shapes that could not be rendered correctly from all angles.

Therefore, re-ordering these facets were necessary which could be done through the facet editor. For instant in figure A.5, if facet (*A*) obscures facet (***B***) it is moved on the top of facet (***B***) and so on. Sometimes trial and error is the only possible way.

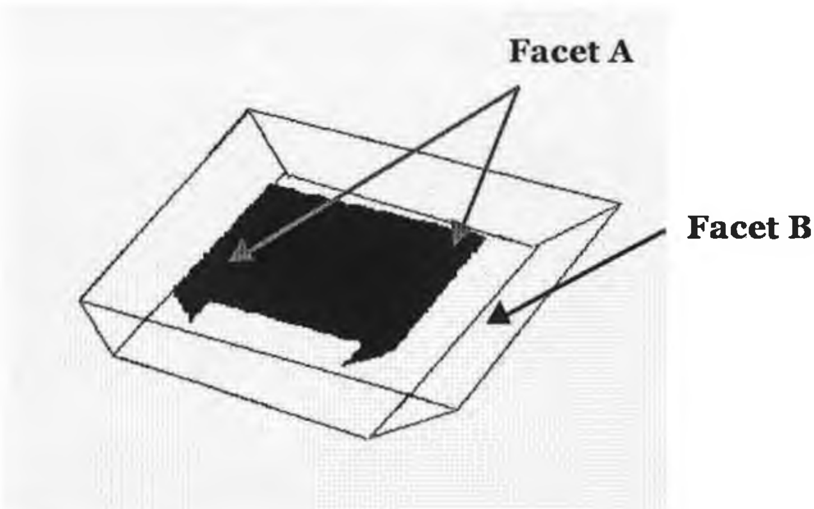


Figure A.5 Two Facets obscure each other

World Editor

Virtual World are built using the World Editor (***WE***) from simple objects (cubes) whose profile is defined by the shapes that are created in the shape editor or imported using Virtual Clip Art (VCA) objects.

A library of VCA objects is available in VRT. Many of these objects have SCL attributes. Objects created could be placed, grouped together, re-sized and coloured within the virtual world.

All the objects in a world were related to each other in a structure similar to a family tree. At the top of the tree is the Root Object, followed by Parents; and Children at the bottom.

Two children with the same parent were known as siblings. As shown in figure A.6.

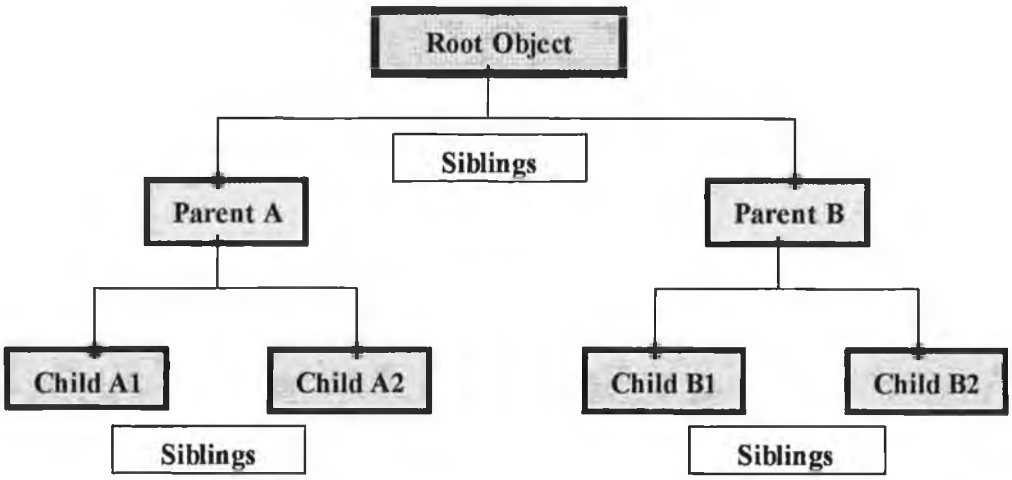


Figure A.6 Tree of all Objects on the World

VRT uses the tree-like structure defined by the objects in the world, as an aid to efficient sorting. Apart from that, it would greatly improve the speed of the virtual world. An object and its siblings are sorted against each other to determine which should be drawn at the back and which at the front. Then VRT assumes that each set of objects is within its parent, and the process is repeated with the next level up the tree. Therefore, objects should be grouped as siblings rather than as parents and children. It is recommended to have maximum of fifteen siblings for each parent.

To do this, a special shape called 'Group', which could not be seen or rendered, is defined. It acts as 'Parent-like' and all the objects under it become its children; siblings among the children. Children however should not ***extend outside*** their parent object, as it would cause sorting and rendering difficulties. Objects should be built from the base i.e. y = zero and worked upward. This is to avoid a collision that causes sorting and rendering problems. During assembly in world, objects could be positioned precisely to each other. Nevertheless, the new position of the object and its sizes must be worked out manually. If the objects were siblings, their positions were relative to the root object. Conversely, if one object is a child to the other, its position is relative to its parent as illustrated in figure A.7. An object's bounding cube should not overlap with other object-bounding cube. However, there were times when it was unavoidable, if this happened, facets and objects need to be ordered manually by attaching an object to ***facets*** of another objects. Both objects must be siblings of each other.

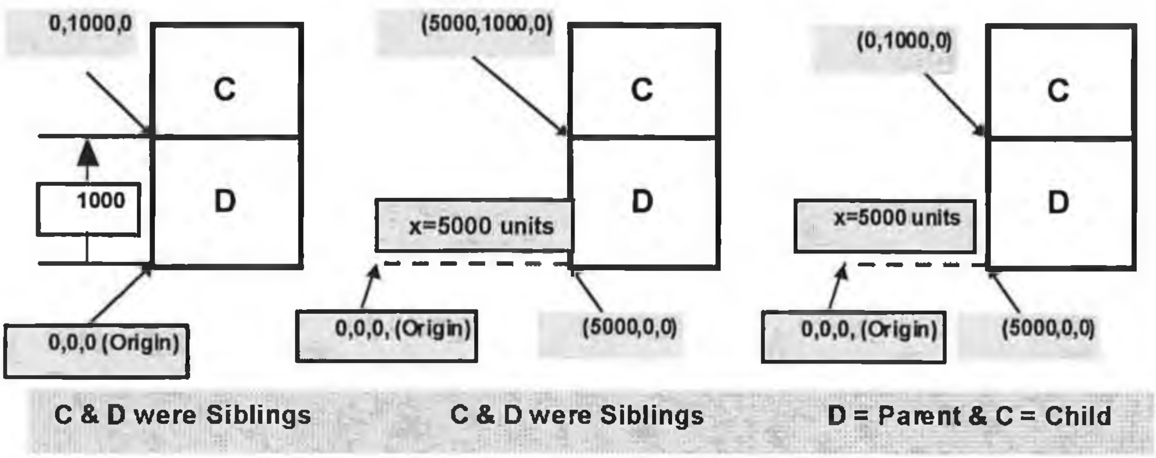


Figure A.7 Position the Objects in/out their Group

Image Editor

The Image Editor permits the world creator to design and edit images, which may be applied as textures to the objects. With the image editor, one can draw, re-scale, crop and edit each image pixel by pixel.

Layout Editor

The Layout Editor permits the designers to customise the Visualiser screen. Multiple windows can be added to let the user have an array of viewpoints of the world.

Sound Editor

The Sound Editor, in conjunction with the end users Sound Card (optional), allows the designer to record, play, import and modify sounds. The sounds are represented as sound waves.

Keyboard Editor

All the keys on the keyboard have been assigned default settings, which may be changed to suit the end user.

Resource Editor

This Editor is used for creating dialog boxes and menus. This lets the designer tell the end user specific details, which may be difficult to translate using the standard interface. VRT stores all information relating to objects in the world in a mathematical description rather than as a picture, thus enabling the complete world to be stored, as a small amount of computer data. Invisible bounding cubes are used in VRT in much the same way as blocks in AutoCAD. These enable objects to be ordered correctly in the world, and henceforth simply moving the bounding cubes can move a series of objects. Facets are used which give the effect of a surface between points. Facets are one-dimensional or two-dimensional surface created by connecting two or more points together. These points are first created in a virtual space and the facets are then created using the points. The invisible bounding cube will then be created around the complete set of facets.

Comparison between *AutoCAD* & *VRT*

Some differences between AutoCAD & VRT are listed in the following Table A.1

AutoCAD	*VRT*
Shape consists of Lines – No Facets, No bounding Cube.	Shape consists of Facet, which was created from Points. Shape created was within its bounding Cube.
Object created could be Positioned directly within the Viewing Area.	Shape created must be stored in shape file and used in W.E. to create objects.
Only one colour is applicable to the boundary of one object.	Different colour could be applied to the facets of the same shape.
Only Static parts could be created.	Static and Dynamic parts could be created.
Object of any shapes could be mirrored or rotated to any angle required and directly positioned within the Viewing Area.	Shape must be duplicated, transformed or rotated, then merged with previous shape to form a complete shape. Only quarter and half shape could be done in this way.
An object could be positioned on the other object accurately by using the assist command such as Intersect, or mid Point etc.	Positions must be calculated before an object was placed to another object.
Overlapping objects have no effect on performance.	Object Overlapped could cause Sorting and Rendering problems.
Arrays command available.	No Array command.
Object dimension could be very accurate.	All fractional unit values were rounded up.
No Grouping of objects.	Objects must be Grouped for effective Sorting.

Appendix B

The Software Manual

This appendix shows the steps and commands used to create shapes in Shape Editor for this project.

Only objects with irregular shapes would be created. Shapes, which have the appearance of –cube would be created in World Editor. Dimensions used in Shape and World Editor are all in units.

All objects were created from relative point – a point that was given an X, Y, and Z position. This position was referred to as the point of origin (0, 0, 0) where the axes met.

Three types of axes were used to build the co-ordinate points.

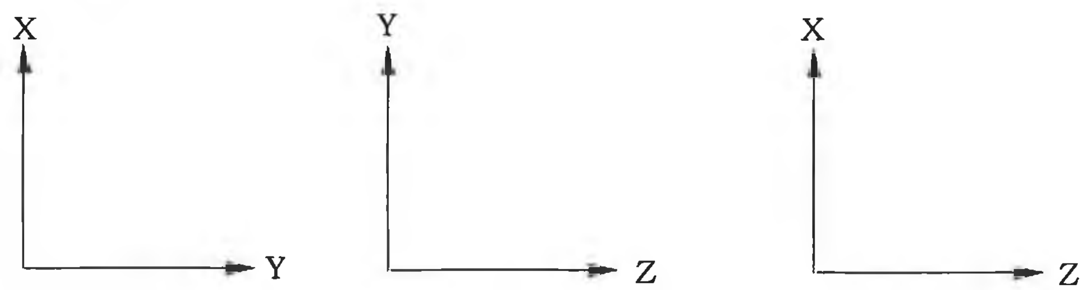
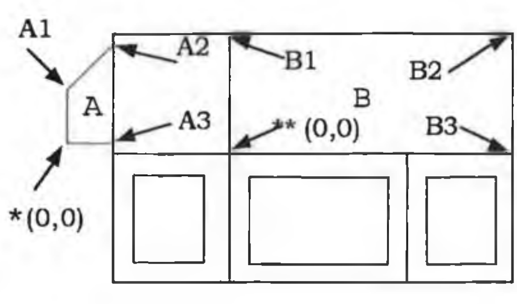


Figure B.1

Relative points of objects were obtained with the aid of AutoCAD. Distance of objects in co-ordinate form was obtained by defining the new origin point for each selected new object. An example is illustrated in figure B 2.



use **UCS** command to define * as point of origin. then to find co-ordinates at point A1, A2, & A3 wrt the point for shape A.

Repeat the command to make point ** as point of origin, then to obtain co-ordinates at B1, B2, & B3 for shape B and so on.

Figure B.2

Objects in the world were created in three manner **(i) AutoCAD, (ii)** World Editor, and **(iii)** Shape Editor. The next section shows the main icon available within the Shape Editor as shown in figure B.3, and example of using **SE** as in figure B.4.

Shape Editor (SE)

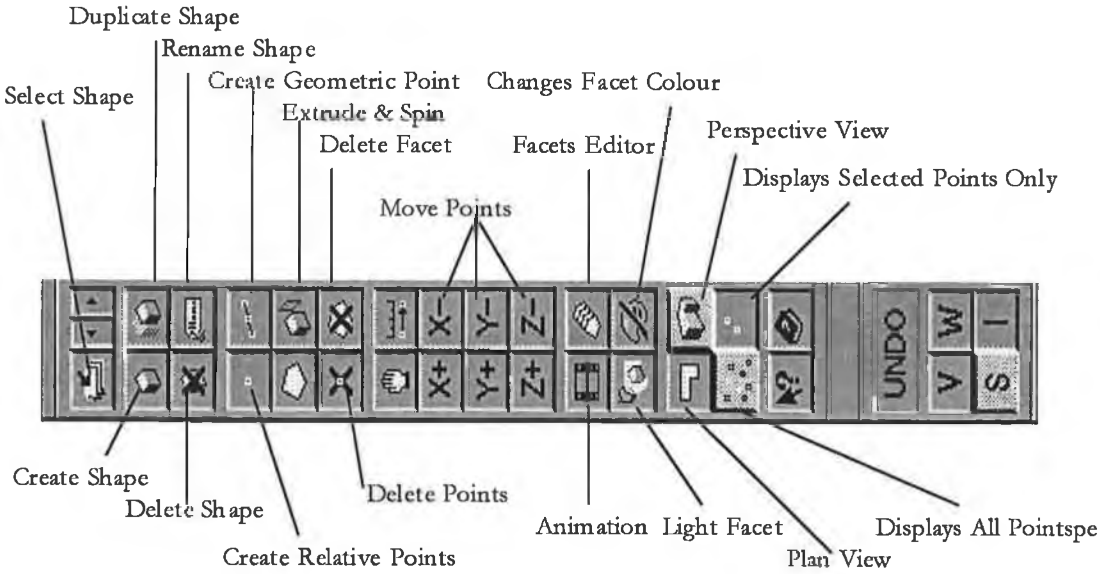


Figure B.3

Shape Name: **Chuck**

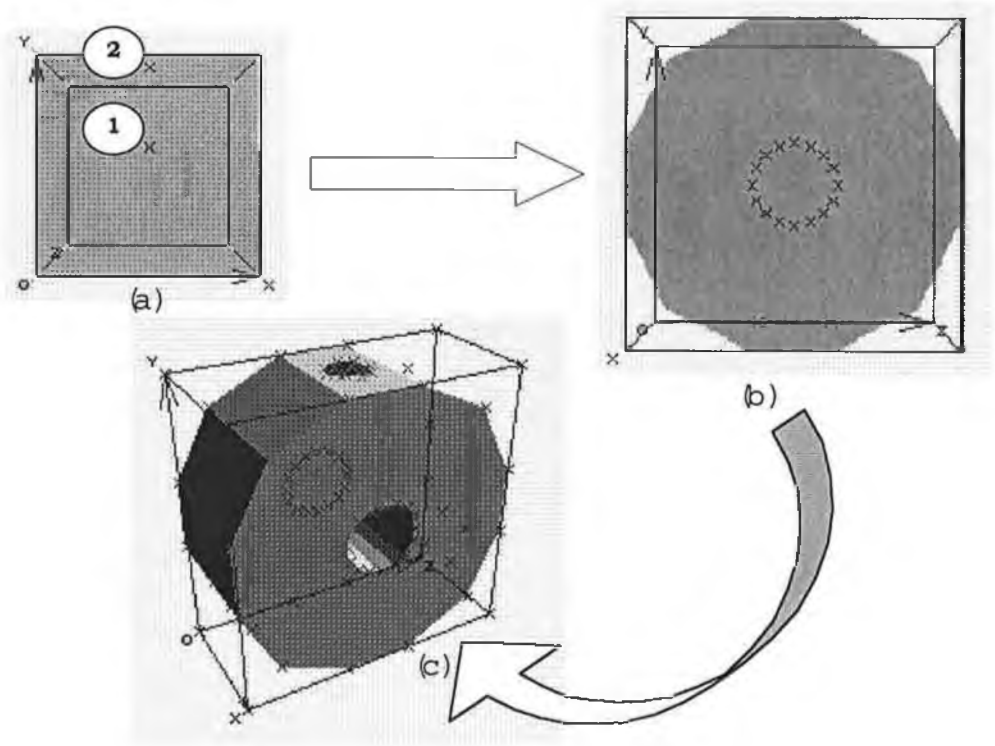


Figure B.4

Procedure

Create two points (*Point 1 & point 2*) in *yz* plane as shown in figure B.4 (a) using *create relative points command* as shown in figure B.4. After that, rotate them 360° around ***x-axis*** for *16 and 12 steps* respectively. All points were picked in Anti Clockwise "acw" direction to create facet. If created in opposite direction, *undo command*. Points were picked again in acw direction. These points will became facet as shown in figure B 4 (b). Selected all the previous points, and then extrude them at 900 units along ***x-axis*** using *Extrude & Spin command*, squared facets were chosen from extrude and spin command.

Pick all the points on other side in acw direction to create facet. 'Shrink-wrap' the bounding cube around the shape using *Wrap Bounding Cube command*. This shape was stored as Chuck in Shape Editor. Create object in object in World Editor name it "Main-Chuck" and then import the Chuck from the Shape Editor.

All facets were in Shape Editor defined in the correct order. Note that if higher step values are chosen, the rounder the shape would be. Higher numbers of facets in the Shape Editor were generated.

The same process could be used to create the following objects:

- Stop-Button
- Indicator-Green
- Indicator-B
- Table-Lever-Handbrake
- Overarm-Chuck-Shaft
- Invis-Centre-Tip
- Invis-Centre-Body
- Spare-Centre-Tip
- Spare-Centre-Body
- Wheel-Handle-1a
- Wheel-Handle-1b
- Wheel-Handle-1c
- Etc...

Word Editor

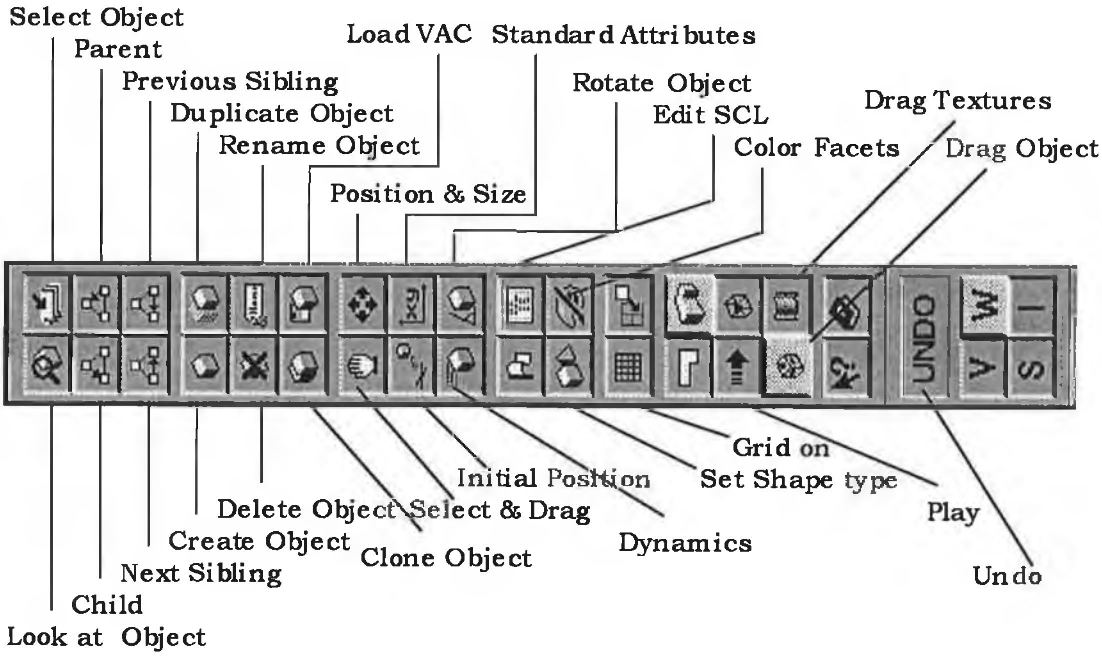


Figure B.5

Object Name: - **Table-Holder**

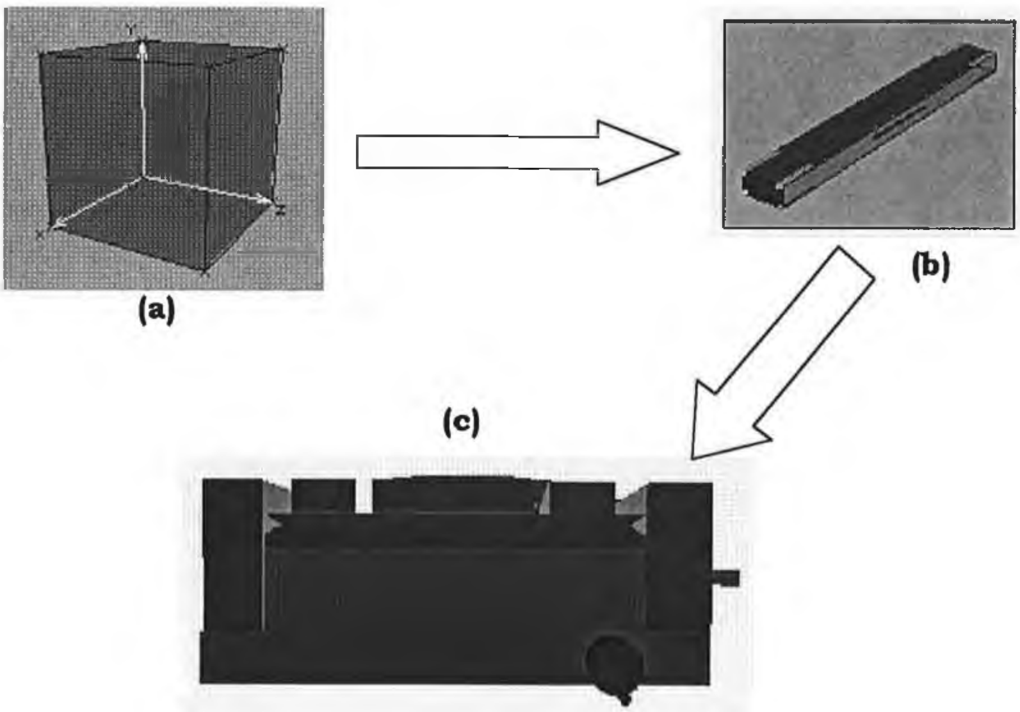


Figure B.6

Procedure

Create a cube using *create object command* as shown in figure B 6 (a), then resize to the desired scale as shown in figure B 6 (b) using *position and size command*. Three copies were created of this object using duplicate object. Put all these three objects into one group and wrap them in group bounding using *warp bounding cube command* and position them in the wanted place using the *position and size command*.

Finally, give this object an initial position. It will be positioned there when you reset the world using the *initial position command*.

Note: use the position and size command to avoid the overlapping problems.

The same way could be carried to create the following objects with a little different:

- Knee-Holder
- Overarm
- Etc...

AutoCAD R13

The third way is using Traditional AutoCAD R13 or whatever CAD package is available and then transfer data in the **DXF** file to Superscape **VRT** Software as mentioned in section 4.6 chapter 4 page 67 already.

Appendix C

SCL Code for Virtual Milling Machine

The programme that attached to Sample-A

```
resume (0, 0);
if (activate (me, 0) || activate (me, 13))
{
invis (me);
vis ('Sample-B');
waitf;
}
```

This Programme that attached to Sample-B

```
resume (1, 1);
if (actchild (me, 0) || actchild (me, 13))
{
  while (mouseb)
  {
    if (yangv ('Main-Chuck')!=iyangv ('Main-Chuck'))
    alert (" Please stop the machine before you take out
the Sample",0);
    else
    {
      invis (me);
      vis ('Sample-A');
      waitf;
    }
  }
}
```

Programme that attached to Spare-Drill-Holder at initial Position

```
if (actchild (me, 0) || actchild (me, 13))
{
  while (mouseb)
  {
  }
  if (!vis? ('Spare-Drill-Holder') || !vis?
      ('Spare-Centre-Holder'))
    alert ("Stop the spindle before positioning tools", 0);
  else
{
if (!vis? ('Spare-Centre-Holder'))
alert (" Disengage and remove the attached tool before
positioning this Tool", 0);
    else
    {
      invis (me);
      vis ('Invis-Drill-Tip');
      vis ('Invis-Drill-Body');
      vis ('Invis-Drill-Holder');
    }
  }
}
```

Programme attached to Spare-Centre-Holder at initial Position

```
if (actchild (me, 0) || actchild (me, 13))
{
  while (mouseb)
  {
  }
  if (!vis? ('Spare-Drill-Holder') || !vis?
      ('Spare-Centre-Holder'))
```

```
    alert ("Stop the spindle before positioning tools", 0);
  else
  {
    if (!vis? ('Spare-Drill-Holder'))
      alert (" Disengage and remove the attached tool
before positioning this Tool", 0);
    else
    {
      invis (me);
      vis ('Invis-Centre-Tip');
      vis ('Invis-Centre-Body');
      vis ('Invis-Centre-Holder');
    }
  }
}
```

The Programme attached Switch-On/Off Button

```
resume (0, 0);
if (activate (me, 0))
{
  while (mouseb)
  {
    if (vis? ('Spare-Drill-Holder') & vis?
         ('Spare-Centre-Holder'))
alert ("Please, Insert or remove tool before start
machining, which is at the Top of the machine", 0);
    else
    {
      if (vis? ('Invis-Drill-Holder') & !vis?
          ('Invis-Centre-Holder'))
      {
        zrot (me)=-200;
        yangv ('Main-Chuck')=20;
        yangv ('Invis-Drill-Holder')=20;
```

```
        waitf;
        clrtrig (me, 0);
      }
      if (!vis? ('Invis-Drill-Holder') & vis?
          ('Invis-Centre-Holder'))
      {
        zrot (me)=-200;
        yangv ('Main-Chuck')=20;
        yangv ('Invis-Centre-Holder')=20;
        waitf;
        clrtrig (me, 0);
      }
    }
  }
}
if (activate (me, 13))
{
  while (mouseb)
  {
    if (vis? ('Invis-Drill-Holder') || vis?
        ('Invis-Centre-Holder'))
    {
      xrot ('OverArm-Handle-Holder')=ixrot
          ('OverArm-Handle-Holder');
      zrot (me)=izrot (me);
      ypos    ('Invis-Drill-Holder')=iypos    ('Invis-Drill-
Holder');
      ypos ('Invis-Centre-Holder')=iypos
          ('Invis-Centre-Holder');
      yangv ('Invis-Drill-Holder')=iyangv
          ('Invis-Drill-Holder');
      yangv ('Invis-Centre-Holder')=iyangv
          ('Invis-Centre-Holder');
      yangv ('Main-Chuck')=iyangv ('Main-Chuck');
      ypos ('Main-Chuck')=iypos ('Main-Chuck');
      ysize ('OverArm-Chuck-Shaft')=iysize
```

```
            ('OverArm-Chuck-Shaft');
        ypos ('OverArm-Chuck-Shaft')=iypos
            ('OverArm-Chuck-Shaft');
      }
    }
}
```

Programme that attached to Stop-Button at SCL

```
resume (0, 1);
if (activate (me, 0) || activate (me, 13))
{
  while (mouseb)
    restart ('Stop-Button');
  waitf;
  clrtrig (me, 0);
}
```

Programme that attached to Stop-Button at Locally Triggered

```
resume (0, 0);
if (activate (me, 0) || activate (me, 13))
{
  zpos ('Stop-Button')+=50;
  if (vis? ('Indicator-B'))
  {
    while (mouseb)
      waitf;
  }
  zpos ('Stop-Button')=izpos ('Stop-Button');
}
```

Programme attached to Indicator-B at SCL

```
resume (0, 0);
if (activate (me, 0))
{
  if (yangv ('Main-Chuck')!=iyangv ('Main-Chuck'))
   alert (" Please Turn the Switch lever before you Stop
the Ma"
        "chine", 0);
  else
  {
    invis (me);
    vis ('Indicator-Green');
    if (vis? ('Stop-Button'))
      waitf;
    clrtrig (me, 0);
  }
}
```

Programme attached to Indicator-Green Button at SCL Program

```
if (first)
  invis (me);
```

Programme attached to Indicator-Green Button at Locally Triggered

```
resume (1, 1);
if (actchild (me, 13))
{
  if (yangv ('Main-Chuck')!=iyangv ('Main-Chuck'))
    alert (" Please Turn the Switch lever before you Stop
the Machine", 0);
  else
```

```
  {
    invis (me);
    vis ('Indicator-B');
    waitf;
  }
}
```

This Programme attached to Knee-Handle-Holder at SCL

```
resume (0, 0);
if (actchild (me, 0))
{
  while (mouseb)
  {
    if (ypos ('ToolVice-Holder')>=ypos ('Main-Chuck')-950)
    {
      stop ('Knee-Handle-Holder');
  alert ("Hey, \r\r The cutting tool will be damaged!", 0);
    }
    else
    {
      ypos ('Knee-Holder')+=25;
      ypos ('Knee-Handle-Holder')+=25;
      zrot ('Knee-Handle-Holder')+=20;
      ypos ('Side-Slider-1')+=25;
      ypos ('Side-Slider-2')+=25;
      ypos ('Cross-Slide-Block-Main-2')+=25;
      ypos ('Table-HandWheel-Holder')+=25;
      ypos ('Saddle-HandWheel-Holder')+=25;
      ypos ('Table-Holder')+=25;
      ypos ('ToolVice-Holder')+=25;
      ypos ('Sample-B')+=25;
      ypos ('Table-Lever-HandBrake')+=25;
      ypos ('Grey')+=25;
      ypos ('Indicator-Button-Group')+=25;
```

```
      ypos ('Stop-Button-Group')+=25;
      ypos ('Switch-On/Off ')+=25;
      waitf;
    }
  }
}

if (actchild (me, 13))
{
  while (mouseb)
  {
    if (ypos ('Knee-Holder')>=iypos ('Knee-Holder'))
    {
      ypos ('Knee-Holder')-=25;
      ypos ('Knee-Handle-Holder')-=25;
      zrot ('Knee-Handle-Holder')-=20;
      ypos ('Side-Slider-1')-=25;
      ypos ('Side-Slider-2')-=25;
      ypos ('Cross-Slide-Block-Main-2')-=25;
      ypos ('Table-HandWheel-Holder')-=25;
      ypos ('Saddle-HandWheel-Holder')-=25;
      ypos ('Table-Holder')-=25;
      ypos ('ToolVice-Holder')-=25;
      ypos ('Sample-B')-=25;
      ypos ('Table-Lever-HandBrake')-=25;
      ypos ('Grey')-=25;
      ypos ('Indicator-Button-Group')-=25;
      ypos ('Stop-Button-Group')-=25;
      ypos ('Switch-On/Off ')-=25;
      waitf;
    }
  }
}
```

The SCL Attached To Saddle Wheel To Control The z-axis Movements

```
resume (0, 0);
if (actchild (me, 0))
{
  while (mouseb)
  {
    'Cross-Slide-Block-Main-2'.flag=1;
    'Cross-Slide-Block-Main-2'.value=100;
    waitf;
  }
}
if (actchild (me, 13))
{
  while (mouseb)
  {
    'Cross-Slide-Block-Main-2'.flag=2;
    'Cross-Slide-Block-Main-2'.value=100;
    waitf;
  }
}
```

Programme attached for the Movement Saddle Along z-axis

```
short  flag=0, value=0;

if (flag==1)
{
  if  (zpos  ('ToolVice-Holder')<=izpos  ('ToolVice-Holder')-
2500)
  {
    stop (me);
```

```
    alert ("Hey you, \r\r I have reached my Max outer
limit!",
        0);
  }
  else
  {
    xrot ('Saddle-HandWheel-Holder')-=value*1.4400;
    zpos ('Cross-Slide-Block-Main-2')-=value;
    zpos ('ToolVice-Holder')-=value;
    zpos ('Sample-B')-=value;
    zpos ('Table-Holder')-=value;
    zpos ('Table-HandWheel-Holder')-=value;
    zpos ('Table-Lever-HandBrake')-=value;
  }
  flag=0;
}
if (flag==2)
{
  if (zpos ('ToolVice-Holder')>=izpos ('ToolVice-Holder'))
  {
    stop (me);
     alert ("Hey you, \r\r I have reached my Max inner
     limit!",0);
  }
  else
  {
    xrot ('Saddle-HandWheel-Holder')+=value*1.4400;
    zpos ('Cross-Slide-Block-Main-2')+=value;
    zpos ('ToolVice-Holder')+=value;
    zpos ('Sample-B')+=value;
    zpos ('Table-Holder')+=value;
    zpos ('Table-HandWheel-Holder')+=value;
    zpos ('Table-Lever-HandBrake')+=value;
  }
  flag=0;
}
```

SCL Attached To The Table Wheel To Control The x-axis Movement

```
resume (0, 0);
if (actchild (me, 0))
{
  while (mouseb)
  {
    'Cross-Slide-Block-Main-1'.flag=1;
    'Cross-Slide-Block-Main-1'.value=100;
    waitf;
  }
}
if (actchild (me, 13))
{
  while (mouseb)
  {
    'Cross-Slide-Block-Main-1'.flag=2;
    'Cross-Slide-Block-Main-1'.value=100;
    waitf;
  }
}
```

This Programme was attached to Knee at SCL

```
resume (0, 0);
if (first)
{
  alert ("\r\r Welcome to My Virtual world", 0);
  invis ('Invis-Drill-Holder');
  invis ('Invis-Centre-Holder');
  invis ('Sample-B');
  invis ('Invis-PaintPot');
  invis ('Invis-PaintPot-Shelf');
  invis ('Wall-BelowPart-1[254]');
```

```
  waitf;
}
```

The attached Program to the OverArm Handle Wheel.

```
resume (0, 0);
if (actchild (me, 0))
{
  while (mouseb)
  {
    if (vis? ('Spare-Drill-Holder') & vis?
        ('Spare-Centre-Holder'))
      alert ("Please insert Tool!", 0);
    if (vis? ('Invis-Drill-Holder') || vis?
    ('Invis-Centre-Holder'))
    {
      if (ypos ('ToolVice-Holder')>=ypos
          ('Invis-Drill-Holder')+ysize
          ('Invis-Drill-Holder')-980)
      {
        stop ('OverArm-Handle-Holder');
        alert ("Hey Man,\r\r Yor are going to break your
Tools!"
            , 0);
      }
      else
      {
        xrot (me)+=25;
        ypos ('Main-Chuck')-=20;
        ypos ('OverArm-Chuck-Shaft')-=20;
        ypos ('Invis-Drill-Holder')-=20;
        ypos ('Invis-Centre-Holder')-=20;
        ysize ('OverArm-Chuck-Shaft')+=20;
        waitf;
```

```
        }
      }
    }
  }
  if (actchild (me, 13))
  {
    while (mouseb)
    {
      if (vis? ('Invis-Drill-Holder') || vis?
          ('Invis-Centre-Holder'))
      {
        if (ypos ('Main-Chuck')>=iypos ('Main-Chuck'))
        {
          stop ('OverArm-Handle-Holder');
          alert ("Hey Man,\r\r The end!", 0);
        }
        else
        {
          xrot (me)-=25;
          ypos ('Main-Chuck')+=20;
          ypos ('OverArm-Chuck-Shaft')+=20;
          ypos ('Invis-Drill-Holder')+=20;
          ypos ('Invis-Centre-Holder')+=20;
          ysize ('OverArm-Chuck-Shaft')-=20;
          waitf;
        }
      }
    }
  }
```

Programme attached to Cross-Slide-Block-Main-1 for Left and Right Movements

```
short  flag=0, value=0;

if (flag==1)
{
  if (xpos ('ToolVice-Holder')<=ixpos ('ToolVice-Holder')-
2500)
  {
    stop (me);
    alert ("Hey you, \r\rI have reached my Max limit!", 0);
  }
  else
  {
    xrot ('Table-HandWheel-Holder')-=value*1.4400;
    xpos ('ToolVice-Holder')-=value;
    xpos ('Table-San-1')-=value;
    xpos ('Table-San-2')-=value;
    xpos ('Table-San-3')-=value;
    xpos ('Sample-B')-=value;
    xpos ('Cross-Slide-Block-Main-1')-=value;
  }
  flag=0;
}
if (flag==2)
{
  if (xpos ('ToolVice-Holder')>=ixpos ('ToolVice-
Holder')+2500)
  {
    stop (me);
    alert ("Hey you, \r\rI have reached my Max limit!", 0);
  }
  else
  {
    xrot ('Table-HandWheel-Holder')+=value*1.4400;
```

```
    xpos ('ToolVice-Holder')+=value;
    xpos ('Table-San-1')+=value;
    xpos ('Table-San-2')+=value;
    xpos ('Table-San-3')+=value;
    xpos ('Sample-B')+=value;
    xpos ('Cross-Slide-Block-Main-1')+=value;
  }
  flag=0;
}
```

Programme attached to OverArm-Level-Handbrake with Spare-Centre-Holder

```
resume (0, 0);
if (activate (me, 0))
{
  if (vis? ('Invis-Centre-Holder'))
  {
    alert (" Are you sure for this operration", 2);
    zrot (me)+=30;
    invis ('Invis-Drill-Holder');
    vis ('Spare-Drill-Holder');
    vis ('Spare-Centre-Holder');
    ypos ('Main-Chuck')=iypos ('Main-Chuck');
    zsize ('OverArm-Handle-Holder')=izsize
        ('OverArm-Handle-Holder');
    while (mouseb)
    {
    }
    waitf;
    zrot (me)=izrot (me);
    clrtrig (me, 0);
  }
}
```

Appendix D

Virtual Milling Machine Questionnaire

Virtual Milling Machine Questionnaire

Name: ____________________ Class: ____________________

Sex: Male ☐ Female ☐

Age:

a) 15-20 ☐

b) 21-25 ☐

c) 26-30 ☐

d) 31+ ☐

1. Previous operator experience with a CNC milling machine:

 a) None. ☐

 b) Basics. ☐

 c) Intermediate. ☐

 d) Expert. ☐

2. The Z axis on a milling machine moves:

 a) right - left. ☐

 b) left - right. ☐

 c) up - down. ☐

 d) backward - forward. ☐

3. Basic milling machines move in three directions, which are called:

 a) up, down, back. ☐

 b) P, Q, & R axis. ☐

 c) A, B, & C axis. ☐

 d) X, Y, & Z axis. ☐

4. Was the Virtual milling machine difficult to operate ?

   a) ☐ It was difficult to follow because ______________________

   b) ☐ Some parts were hard to follow and understand.

   c) ☐ All parts were easy to follow and understand.

5. Did you find this training method interesting ?

   a) ☐ Not interesting because ______________________

   b) ☐ Some parts interesting.(what parts) ______________________

   c) ☐ All parts interesting.

6. Do you think that a book would be more beneficial in order to learn the working characteristics of the machine ?

   ☐ Yes because ______________________

   ☐ No because ______________________

7. Do you think this form of training application is beneficial to first time learners?

   ☐ Yes because ______________________

   ☐ No because ______________________

8. If this training application was used at the early stage of training an operator. Do you think this would reduce the risk of a dangerous accident ?

   ☐ Yes because ______________________

   ☐ No because ______________________

10. Were the navigation instructions clear and easy to understand?

   ☐ Yes because ______________________

   ☐ No (How could they be improved) ______________________

11. Was it easy to navigate throughout the virtual worlds?

- [ ] Yes because ______________________________
- [ ] No because ______________________________

12. Was it easy to identify the various components of the milling machine?

- [ ] Yes because ______________________________
- [ ] No (describe the difficulty(s) you experienced) ______________

13. In your opinion, will Virtual Reality be an effective training tool for learning how to operate a milling machine?

- [ ] Yes because ______________________________
- [ ] No because ______________________________

14. Did you enjoy the exercise?

- [ ] Yes
- [ ] No

15. What is the greatest benefit you can foresee by using this technology to train milling machine operators?

____
____
____
____

Additional Comments:

____
____
____
____
____
____
____

Appendix E

World Editor Family Tree

Knee-Holder

Knee
Cross-Slide-Block-Main-3

Left-Vertical-Slider

Right-Vertical-Slider

Grey

Switch-On/Off
Stop-Button-Group
  Stop-Button-Root
  Stop-Button

Indicator-Button-Group
  Indicator-A
  Indicator-B

Knee-Handle-Holder

Wheel-Handle-1A
Wheel-Handle-1B
Wheel-Handle-1C

Cross-Slide-Block-Main-2

Side-Slide-1
Side-Slide-2

Saddle-HandWheel-Holder

Vernier1-A
Vernier1-B
Vernier1-C

Table-HandWheel-Holder

Vernier2-A
Vernier2-B
Vernier2-C

Table-Holder

Table-Front
Table-San-1
Table-San-2
Table-San-3
Table-Back
Table-Base

Table-Lever-HandBrake

Tool-Vice-Holder

Nut-Holder-1
- *Top-Piece-1*
- *Hex-Nut-1*

Nut-Holder-2
- *Top-Piece-2*
- *Hex-Nut-2*

Nut-Holder-3
- *Top-Piece-3*
- *Hex-Nut-3*

Saddle-Turn-Table

Main-Chuck

OverArm-Hold

OverArm-Chuck-Shaft
OverArm

Base-Body

Base-Block-Top

Spare-Drill-Holder

Spare-Drill-Tip
Spare-Drill-Body
Spare-Drill-Taper

Spare-Centre-Holder

Spare-Centre-Tip
Spare-Centre-Body
Spare-Centre-Taper

Spare-Chuck-Key

Chuck-Key-Arm-A-Rep
Chuck-Key-Arm-B-Rep
Chuck-Key-Main-Rep
Chuck-Key-Lag-Rep

OverArm-Handle-Holder

Wheel-Handle-2B
Wheel-Handle-2C
Wheel-Handle-2D

OverArm-Lever-HandBrake

Appendix F

Virtual Reality Terms

Virtual Reality Terms

-A-

A-Buffering -A variation of z buffering that keeps track of pixels in z-planes for blending and anti-aliasing.

Accelerator -specialised hardware to increase speed of graphics manipulation.

Ambient light -General non-directional illumination.

Artificial Intelligence -The attempt to mimic and automate human cognitive skills through rules and knowledge representation techniques (e.g., understanding visual images, recognising speech and written text, solving problems, making medical diagnoses, heuristic knowledge, etc.).

Augmented Reality -This is the use of transparent display worn as glasses on which data can be projected. This allows someone to repair radar, for example, and have the needed data displayed on the glasses while walking around the radar.

-B-

Binocolour –Displaying a different image to each eye for the purpose of stereographic viewing

Bodysuit -A complete human covering supporting VR sensors and effectors (promised but never rally produced by VPL. (See data suit)

BOOM -Binocular omni-orientational monitor. A 3-D display device suspended from a weighted boom that can swivel freely about so the viewer does not have to wear an HMD; instead, it steps up to the viewer like a pair of binoculars. The boom's position communicates the user's point of view to the computer.

Browser -Overviews such as indexes lists or animated maps to provide a means of navigating through the physical, temporal, and conceptual elements of a VR.

-C-

Co-ordinates –A set of data values that determine the location of a point in space. The number of co-ordinates to the dimensionality of the space.

CAVE -A VR using projection devices on the walls and ceiling togive the illusion of immersion.

Concept Map -A browser or terms, definitions, or icons arranged in semantic proximity.

Convergence -The angle between the two eyes at a fixation point. This changes for objects at varying depths in the real world and on 3-D displays.

Culling -Removing invisible pieces of geometry and only sending potentially visible geometry to the graphics subsystem. Simple culling involves rejecting objects not in the view frustum. More systems that are complex take into account occlusion of some objects by others, e.g. a building hiding trees behind it.

Cybernetic Simulation -Dynamic model of a world filled with objects that exhibit lesser or greater degrees of intelligence.

Cyberspace -A place filled with virtual "stuff" populated by people with virtual bodies. A special virtual space, which promotes experiences involving the whole body

Cyberspace Playhouse -Social centre or place where people go to play roles in simulations.

-D-

DataGlove -A glove wired with sensors and connected to a computer system for gesture recognition. It is used for tactile feedback and it often enables navigation through a virtual environment and interaction with *3-D* objects within it.

DataSpace -A visualised representation of complex information.

DataSuit -Same as a DataGlove, but designed for the entire body. Only one DataSuit has yet been built, with limited capabilities.

Direct Manipulation -A term coined by Shneiderman to reflect the use of computer icons or text as if they were real objects.

DIS -Distributed Interactive Simulations. Based on SIMNET.

Disorientation -Confusion about distances and directions for navigation.

DSI -Defense SImulation Internet: A component of the Internet that supports DIS and SIMNET and permits scheduled guaranteed bandwidth.

Dynamics -The way that objects interacts and moves. The rules that govern all actions and behaviours within the environment.

Dynamic Lighting -Changes in lighting effects on objects as they and the observer move.

-E-

E-mail -electronic mail, often sent over the Internet or a commercial carrier.

Effectors -The output techniques that communicate a user's movements or commands to the computer and to the VR.

-F-

Facets -Facets is one to two dimensional surfaces created by joining two or more points.

Force Feedback -Representations of the inertia or resistance objects have when they are moved or touched.

Fish Tank VR -With stereographic display systems attached to a monitor and the scene's virtual image behind the screen, the egocentric projection is called a fish tank.

-H-

Haptic Interfaces -Interfaces that use all the physical sensors that provide us with a sense of touch at the skin level and force feedback information from our muscles and joints.

Head-coupled -Displays or robotic actions that are activated by head motion through a head-tracking device.

Head Mounted Display –A set of goggles or helmet with tiny monitors in front of each eye to generate images seen by the wearer as 3-D. Often the HMD is combined with a head tracker so that the images displayed in the HMD changes as the head moves.

Head Tracking -monitoring the position of the head through various devices.

Heads Up Display -(HUD) A display device that lets users sees graphics superimposed on their view of the world. (Created for aviators to see symbols and dials while looking out the window.)

-I-

Immersion -The observers emotional reaction to the virtual world as being part of it

Interface (VR) -A set of devices, software, and techniques that connect computers with people to perform tasks

Internet -A worldwide digital network capable of supporting shared virtual worlds

-M-

Model (VR) -A computer generated simulation of something real.

-N-

Navigate (VR) -Purposeful motion through virtual space.

-P-

Perspective -The rules that determine the relative size of objects on a flat viewing surface to give the perception of depth.

Presence -A feeling of being immersed in an environment, able to interact with object there. A defining characteristic of a true immersive VR system

-R-

Real Time (VR) -Action taking places with no perceptible or significant delay after the input that initiates the action.

Real Time Imaging -Graphics or images synchronised with real world time events

Resolution VR -Usually the number of pixels in a VR display

-S-

SCL -Superscape Control Language. Similar in many respects to that of the programming language C that controls objects within the virtual world. The language enables the world designer to give objects limited intelligence.

Simulation -A displays of graphic information which may represent a physical situation.

Simulator Sickness -Various Disturbances, ranging in degree from a feeling of unpleasantness, disorientation and headaches to extreme nausea, caused by various aspects of a simulator. Possible factors include sensory distortions such as abnormal movement of arms and heads because of the weight of equipment, long delays or lags in feedback and missing visual cues from convergence and accommodation.

-T-

Texture mapping -A bitmap added to an object to give added realism.

Tracker -A device that emits numeric co-ordinates for it's the changing position in space. (Enactive tracking – voluntarily creating the kinaesthetic cues that correlate with scene motion, without a tracker).

Transparency -How invisible and unobtrusive a VR system is.

-U-

Universe -This is the "container" of all entities in a VR. Entities can be temporarily added or removed from consideration by the simulation manager. The sequence of events in the simulation loop can be user-defined.

-V-

Viewpoints -Points from which raytracing and geometry creation occurs. The geometric eye point of the simulation. You can have multiple viewpoints. They can be attached to multiple sensors.

Virtual Environment -Realistic simulations of interactive scenes

Virtual Prototyping –Simulation of an intended designer product to illustrate the characteristics before actual construction. Usually used as an exploratory tool for developers as a communications prop for persons reviewing proposed designs.

Virtual Environments -Realistic simulations of interactive scenes.

Visualisation -Use of computer graphics to make visible numeric or other quantifiable relationships.

Voxel -A cubic volume pixel for quantizing 3D space.

-X-

X-axis -In relation to Milling Machine it is axis of motions always perpendicular to the Z-axis, and horizontal and parallel to the Workholding surface. In the Superscape world, the X-axis runs from east to west.

-Y-

Y-axis -In relation to Milling Machine it is axis of motions perpendicular to both the X and Z-axis. In the Superscape world, the Y-axis runs Up and Down.

-Z-

Z-axis -In relation to Milling Machine it is axis of motions parallel to the principal spindle of the Machine. In the Superscape world, the Z-axis runs from north to south and the positive Z-axis is defined as "forward".

Appendix G

Virtual Milling Machine File Setup

Virtual Milling Machine File Setup

What do you need to run this disk?

Software Requirements: Superscape VRT 5.1 or Later.

Hardware Requirements: You need to have the same specifications which are described previously in this thesis in page 63, table 4.1 in chapter 4 "Minimum and recommended requirements for running VRT 5.1"and the Hardware Key "*Dongle*".

1. Create a directory on C:\ drive called "VR_DCU"
2. Copy the contents of the floppy disk into C:\ VR_DCU.

You should now have a directory structure as follows:

C:\ VR_DCU

VMM.vrt

3- Start Superscape VRT by clicking on Superscape icon on the screen.

4- Figure G.1 will be appeared then go to File/Open to open "**VMM.vrt**".

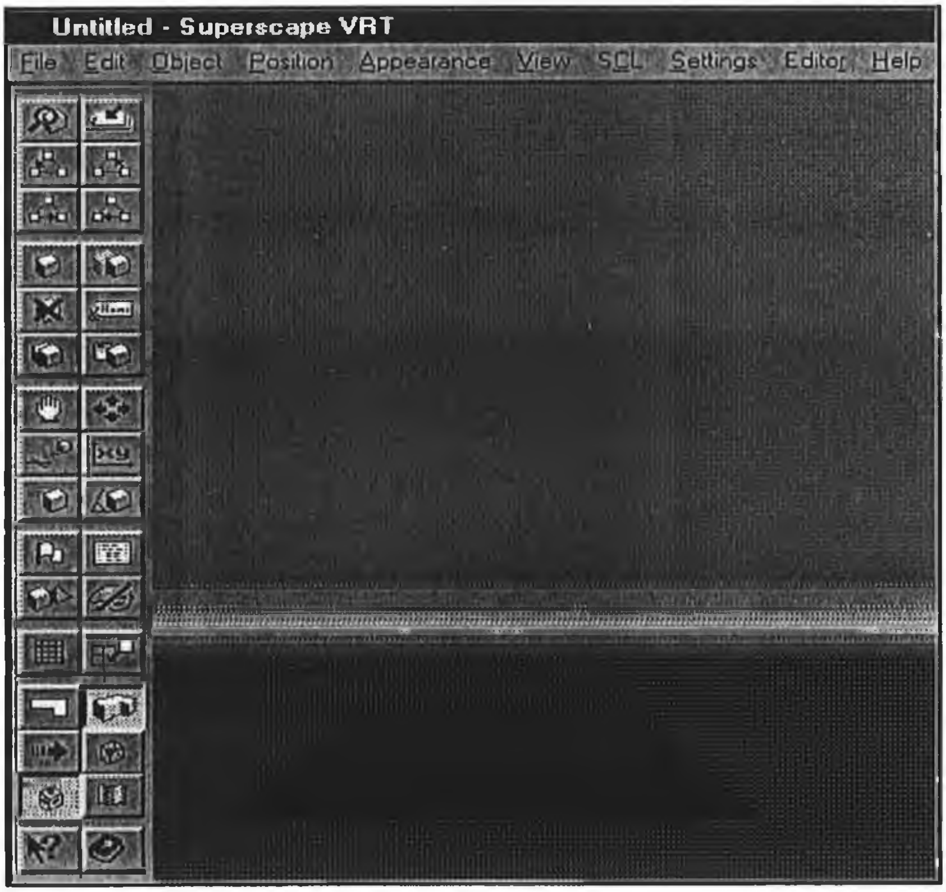


Figure G.1

5- This screen will appear as in figure G.2 and if you want move it, you should click on close button as shown.

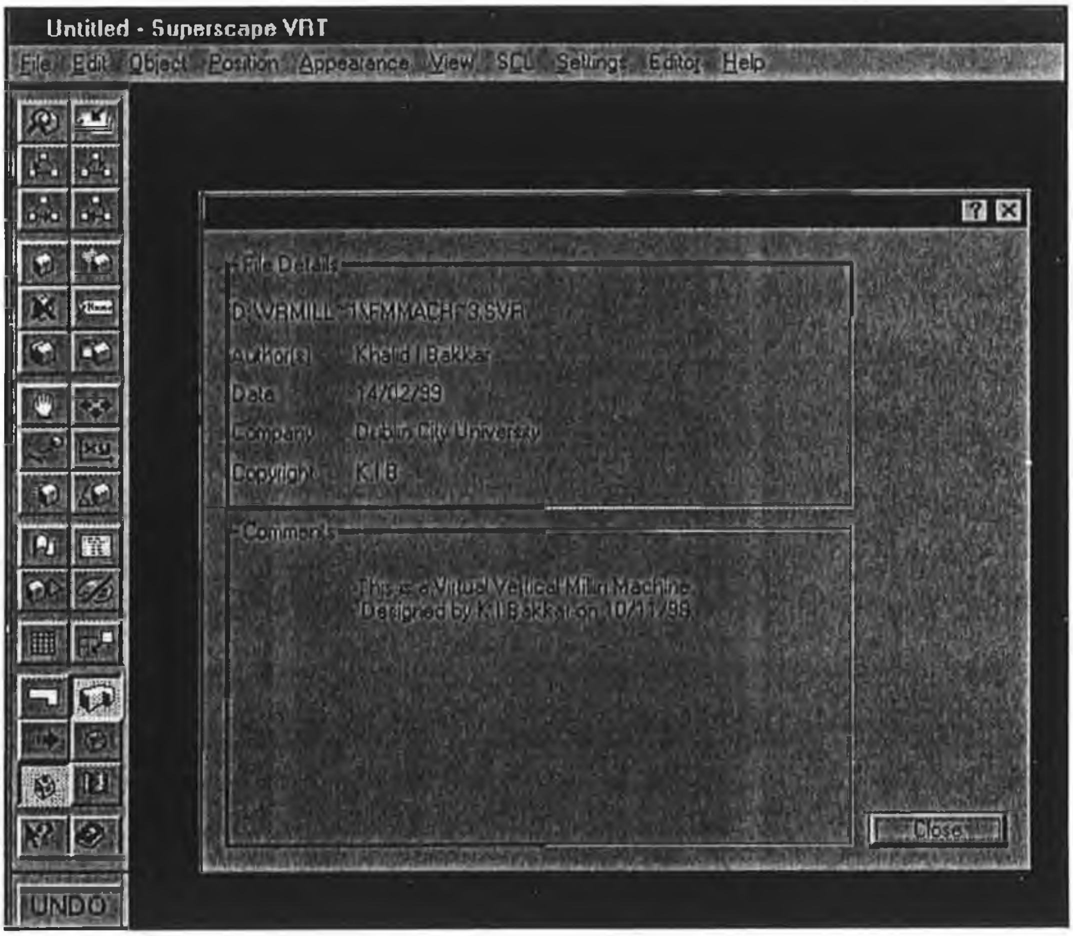


Figure G.2

6- This greeting message will appear as shown in figure G.3.

Figure G.3.

7- After you press Ok icon in figure G.3, if you Press F2 'Function Key' on the keyboard to switch the view to Visualiser Editor as shown in figure G.4. if you,

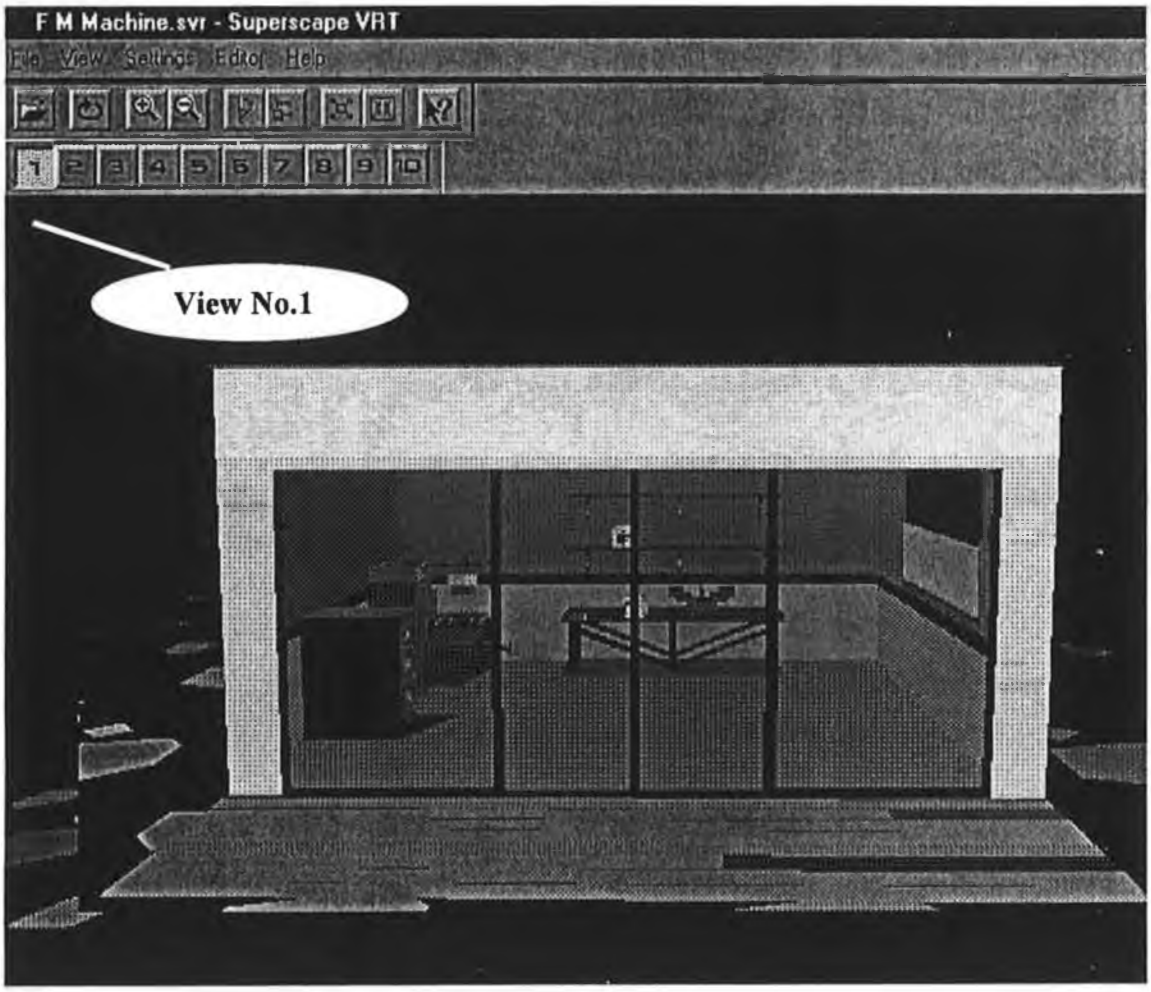


Figure G.4

8- Press F4 to switch you to the World Editor.

9- Press F5 to switch you to the Shape Editor.

10- Use the numbering icons at the 'Top-Left' as shown in figure G.4 to switch to different views.

11- To move the virtual machine parts e.g. "Table, Knee, or Saddle" of the virtual machine, take the mouse indicator to the part you want to move then click on the mouse right button to do so, if you want to reverse the action use the other mouse button.

12- If you want to reset the world Press F12.