

BCA (DES)306- 3rd YEAR

VIRTUAL REALITY

[Type here]

**Dr. Babasaheb Ambedkar
Open University**



Block 1

BCA(DES)-306

VIRTUAL REALITY

Copyright © 2018 KnowledgeManagement and Research Organization.

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by means of, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

Acknowledgment

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/ amendment in future edition of this book.

The content is developed by taking reference of online and print

Publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken “as is” and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.’



ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a

Successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual

Involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect,

Students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self- instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual- skills objectives may be met by designing instructions that make

use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive

Communication clues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to r e d u c e

This. Divide and to bring this Self Instructional Material as the b e s t

teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self- instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts.

Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline.

These could be in the form of students 'assignments, a research project

or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you
Expect.

All the best for your studies from our team!

All the best for your studies from our team!

UNIT 1- INTRODUCTION TO VIRTUAL REALITY

Notes

"Virtual Reality: A computer system used to create an artificial world in which the user has the impression of being in that world and with the ability to navigate through the world and manipulate objects in the world." - C. Manetta and R. Blade

"Virtual Reality allows you to explore a computer generated world by actually being in it" - B. Sherman and P. Judkins

One of the best definition of Virtual Reality to date comes from the The Silicon Mirage:

"Virtual Reality is a way for humans to visualize, manipulate and interact with computers and extremely complex data".

Virtual reality is an artificial environment that is created with software and presented to the user in such a way that the user suspends belief and accepts it as a real environment. On a computer, virtual reality is primarily experienced through two of the five senses: sight and sound. The simplest form of virtual reality is a 3-D image that can be explored interactively at a personal computer, usually by manipulating keys or the mouse so that the content of the image moves in some direction or zooms in or out. More sophisticated efforts involve such approaches as wrap-around display screens, actual rooms augmented with wearable computers, and haptics devices that let you feel the display images.

The applications being developed for VR run a wide spectrum, from games to architectural and business planning. Many applications are 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. Visualizing the ebb and flow of the world's financial markets. Navigating a large corporate information base, etc.

Virtual reality can be divided into:

- The simulation of a real environment for training and education.
- The development of an imagined environment for a game or interactive story.

Notes

Main types of VR

Although it is difficult to categorise all VR systems, most configurations fall into three main categories and each category can be ranked by the sense of immersion, or degree of presence it provides. Immersion or presence can be regarded as how powerfully the attention of the user is focused on the task in hand. Immersion presence is generally believed to be the product of several parameters including level of interactivity, image complexity, stereoscopic view, field of regard and the update rate of the display. For example, providing a stereoscopic rather than monoscopic view of the virtual environment will increase the sense of immersion experienced by the user. It must be stressed that no one parameter is effective in isolation and the level of immersion achieved is due to the complex interaction of the many factors involved.

As will be shown in this report, the type of VR system being used an important consideration when one investigates the genesis of sickness symptoms and the type of symptoms that may develop.

Adventure games

Textually described virtual worlds where the user perceives the virtual environment through mental images based on the words read (like reading a novel).

Desktop

3D virtual environment graphically displayed on a desktop computer monitor.

Non-immersive systems, as the name suggests, are the least immersive implementation of VR techniques. Using the desktop system, the virtual environment is viewed through a portal or window by utilising a standard high resolution monitor. Interaction with the virtual environment can occur by conventional means such as keyboards, mice and trackballs or may be enhanced by using 3D interaction devices such as a SpaceBallä; or DataGloveä; .

The non-immersive system has advantages in that they do not require the highest level of graphics performance, no special hardware and can be implemented on high specification PC clones. This means that these systems can be regarded as the lowest cost VR solution which can be used for many applications. However, this low cost means that these systems will always be outperformed by more sophisticated

implementations, provide almost no sense of immersion and are limited to a certain extent by current 2D interaction devices. Additionally, these systems are of little use where the perception of scale is an important factor. However, one would expect to see an increase in the popularity of such systems for VR use in the near future. This is due to the fact that Virtual Reality Modelling Reality Language (VRML) is expected to be adopted as a de-facto standard for the transfer of 3D model data and virtual worlds via the internet. The advantage of VRML for the PC desktop user is that this software runs relatively well on a PC, which is not always the case for many proprietary VR authoring tools. Furthermore, many commercial VR software suppliers are now incorporating VRML capability into their software and exploring the commercial possibilities of desktop VR in general.

Projected

3D environment projected onto a screen. Enables a single user to demonstrate concepts to a group of people. A CAVE, where several screens are used to surround the user with images, is the most advanced form of projected VR in use today.

Semi-immersive

Most advanced flight, ship and vehicle simulators are semi-immersive. The cockpit, bridge, or driving seat is a physical model, whereas the view of the world outside is computer-generated (typically projected).

Semi-immersive systems are a relatively new implementation of VR technology and borrow considerably from technologies developed in the flight simulation field.

A semi-immersive system will comprise of a relatively high performance graphics computing system which can be coupled with either:

- A large screen monitor
- A large screen projector system
- Multiple television projection system

In many ways, these projection systems are similar to the IMAX theatres discussed in section 1.1. Using a wide field of view, these systems increase the feeling of immersion or presence experienced by the user. However, the quality of the projected image is an important consideration. It is important to calibrate the geometry of the projected image to the shape of the screen to prevent distortions and the resolution will determine the quality of textures, colours, the ability of define shapes and the ability of the user to read text on-screen. The resolutions of projection systems range from 1000 - 3000 lines but to achieve the highest levels it may be necessary to use multiple projection systems which are more expensive.

Semi-immersive systems therefore provide a greater sense of presence than non-immersive systems and also a greater appreciation of scale. In addition, images can be provided that are of a far greater resolution than HMDs and this implementation provides the ability to share the virtual experience. This may have a considerable benefit in educational applications as it allows simultaneous experience of the VE which is not available with head-mounted immersive systems. Additionally, stereographic imaging can be achieved, using some type of shuttered glasses in synchronisation with the graphics system.

Immersive

3D environment seen through a head-mounted display (HMD). In a completely immersive system the user feels part of the environment (experiences a feeling of 'presence'). The user has no visual contact with the physical world.

Immersive virtual reality is a hypothetical future technology that exists today as virtual reality art projects, for the most part. It consists of immersion in an artificial environment where the user feels just as immersed as they usually feel in consensus reality.

Types Of Technology:

Output devices

- Head-mounted displays (stereoscopic goggles)
- Caves

Input devices

- Data gloves
- Some kinds of joy sticks, 3D mice, etc.

Distributed Virtual Environments

The idea behind distributed VR is very simple; a simulated world runs not on one computer system, but on several. The computers are connected over a network (possibly the global Internet) and people using those computers are able to interact in real time, sharing the same virtual world. In theory, people can be sitting at home in London, Paris, New York and Edmonton, all interacting in a meaningful way in VR.

It's clear that there are a number of obstacles to be overcome in achieving this goal. The fact that we want people to be able to access it from their homes means that we have to be able to run over relatively limited-bandwidth links, such as

28.8k modems. The fact that we want to run over the Internet means that we have to tolerate a certain amount of latency in the delivery of update information. Finally, the fact that people are running on different computer systems with different hardware and different software means that we must design the system for portability.

Any of the above, where there are multiple users present in the world. Sometimes called 'shared worlds'

Telepresence

Control remote devices by mimicking actual events in the physical world as they happen. Useful for performing operations in hazardous environments or where it is otherwise inconvenient to be present in person

Telepresence refers to a set of technologies which allow a person to feel as if they were present, to give the appearance of being present, or to have an effect, via telepresence, at a place other than their true location.

Telepresence requires that the users' senses be provided with such stimuli as to give the feeling of being in that other location. Additionally, users may be given the ability to affect the remote location. In this case, the user's position, movements, actions, voice, etc. may be sensed, transmitted and duplicated in the remote location to bring about this effect. Therefore information may be traveling in both directions between the user and the remote location.

A popular application is found in telepresence videoconferencing, the highest possible level of videotelephony. Telepresence via video deploys greater technical sophistication and improved fidelity of both sight and sound than in traditional videoconferencing. Technical advancements in mobile collaboration have also extended the capabilities of videoconferencing beyond the boardroom for use with hand-held mobile devices, enabling collaboration independent of location.

Augmented reality

Combining the physical and virtual worlds, typically to annotate the physical world or to provide missing information. For example, to recreate the view that an operator of an aircraft would have seen if it had not been foggy.

Augmented reality (AR) is a live, direct or indirect, view of a physical, real-world environment whose elements are augmented by computer-generated sensory input such as sound, video, graphics or GPS data. It is related to a more general concept called mediated reality, in which a view of reality is modified (possibly even diminished rather than augmented) by a computer. As a result, the technology functions by enhancing one's current perception of reality. By contrast, virtual reality replaces the real world with a simulated one.

Augmentation is conventionally in real-time and in semantic context with environmental elements, such as sports scores on TV during a match. With the help of advanced AR technology (e.g. adding computer vision and object recognition) the information about the surrounding real world of the user becomes interactive and digitally manipulable. Artificial information about the environment and its objects can be overlaid on the real world. The term augmented reality is believed to have been coined in 1990 by Thomas Caudell, working at Boeing.

Research explores the application of computer-generated imagery in live-video streams as a way to enhance the perception of the real world. AR technology includes head-mounted displays and virtual retinal displays for visualization purposes, and construction of controlled environments containing sensors and actuators.

Shutter Glasses

Liquid Crystal Shutter (LCS) glasses are an important technology when considering semi-immersive systems and consist of a lightweight headset with a liquid crystal lens placed over each eye. Stereopsis works on the principle that in order to perceive depth in a scene, the observer must see slightly different images of the scene under regard in each eye. In the real world this occurs because the two eyes are placed slightly apart in the head, and so each eye views the scene from a slightly different position.

The graphics computer used displays slightly different left and right views (known as a stereo pair) of the virtual environment sequentially on the display

system. To achieve the stereoscopic effect, the glasses either pass or block an image that is produced on the VDU or projected display. When the left image is displayed, the left eye lens is switched on, allowing the viewer's left eye to see the screen. The right eye lens, however, remains off, thus blocking the right eyes view. When the right image is displayed, the opposite occurs. This switching between images occurs so rapidly that it is undetectable by the user, who fuses the two images in the brain to see one constant 3D image.

Technology requirements Notes

Following are required for experiencing VR

- **Hardware** capable of rendering real-time 3D graphics and high-quality stereo sound.
- **Input** devices to sense user interaction and motion.
- **Output** devices to replace user's sensory input from the physical world with computer-generated input.
- **Software** that handles real-time input/output processing, rendering, simulation, and access to the world database in which the environment is defined.

What do you think of when you hear the words **virtual reality (VR)**? Do you imagine someone wearing a clunky helmet attached to a computer with a thick cable? Do visions of crudely rendered pterodactyls haunt you? Do you think of Neo and Morpheus traipsing about the Matrix? Or do you wince at the term, wishing it would just go away?

If the last applies to you, you're likely a computer scientist or engineer, many of whom now avoid the words virtual reality even while they work on technologies most of us associate with VR. Today, you're more likely to hear someone use the words **virtual environment (VE)** to refer to what the public knows as virtual reality. We'll use the terms interchangeably in this article.

Naming discrepancies aside, the concept remains the same - using computer technology to create a simulated, three-dimensional world that a user can manipulate and explore while feeling as if he were in that world. Scientists, theorists and engineers have designed dozens of devices and applications to achieve this goal. Opinions differ on what exactly constitutes a true VR experience, but in general it should include:

- Three-dimensional images that appear to be life-sized from the perspective of the user
- The ability to track a user's motions, particularly his head and eye movements, and correspondingly adjust the images on the user's display to reflect the change in perspective

In this article, we'll look at the defining characteristics of VR, some of the technology used in VR systems, a few of its applications, some concerns about virtual reality and a brief history of the discipline. In the next section, we'll look at how experts define virtual environments, starting with immersion.

Virtual Reality Immersion

In a virtual reality environment, a user experiences **immersion**, or the feeling of being inside and a part of that world. He is also able to **interact** with his environment in meaningful ways. The combination of a sense of immersion and interactivity is called **telepresence**. Computer scientist Jonathan Steuer defined it as “the extent to which one feels present in the mediated environment, rather than in the immediate physical environment.” In other words, an effective VR experience causes you to become unaware of your real surroundings and focus on your existence inside the virtual environment.

Jonathan Steuer proposed two main components of immersion: **depth of information** and **breadth of information**. Depth of information refers to the amount and quality of data in the signals a user receives when interacting in a virtual environment. For the user, this could refer to a display's resolution, the complexity of the environment's graphics, the sophistication of the system's audio output, et cetera. Steuer defines breadth of information as the "number of sensory dimensions simultaneously presented." A virtual environment experience has a wide breadth of information if it stimulates all your senses. Most virtual environment experiences prioritize visual and audio components over other sensory-stimulating factors, but a growing number of scientists and engineers are looking into ways to incorporate a users' sense of touch. Systems that give a user force feedback and touch interaction are called **haptic systems**.

For immersion to be effective, a user must be able to explore what appears to be a life-sized virtual environment and be able to change perspectives seamlessly. If the virtual environment consists of a single pedestal in the middle of a room, a user should be able to view the pedestal from any angle and the point of view should shift according to where the user is looking. Dr. Frederick Brooks, a pioneer in VR technology and theory, says that displays must project a frame rate of at least 20 - 30 frames per second in order to create a convincing user experience.

The Virtual Reality Environment

Other sensory output from the VE system should adjust in real time as a user explores the environment. If the environment incorporates 3-D sound, the user must be convinced that the sound's orientation shifts in a natural way as he maneuvers through the environment. Sensory stimulation must be consistent if a user is to feel immersed within a VE. If the VE shows a perfectly still scene, you wouldn't expect to feel gale-force winds. Likewise, if the VE puts you in the middle of a hurricane, you wouldn't expect to feel a gentle breeze or detect the scent of roses.

Lag time between when a user acts and when the virtual environment reflects that action is called **latency**. Latency usually refers to the delay between the time a user turns his head or moves his eyes and the change in the point of view, though the term can also be used for a lag in other sensory outputs. Studies with flight simulators show that humans can detect a

latency of more than 50 milliseconds. When a user detects latency, it causes him to become aware of being in an artificial environment and destroys the sense of immersion.

An immersive experience suffers if a user becomes aware of the real world around him. Truly immersive experiences make the user forget his real surroundings, effectively causing the computer to become a non entity. In order to reach the goal of true immersion, developers have to come up with input methods that are more natural for users. As long as a user is aware of the interaction device, he is not truly immersed.

In the next section, we'll look at the other facet of telepresence: interactivity.

Virtual Reality Interactivity

Immersion within a virtual environment is one thing, but for a user to feel truly involved there must also be an element of **interaction**. Early applications using the technology common in VE systems today allowed the user to have a relatively passive experience. Users could watch a pre-recorded film while wearing a **head-mounted display (HMD)**. They would sit in a motion chair and watch the film as the system subjected them to various stimuli, such as blowing air on them to simulate wind. While users felt a sense of immersion, interactivity was limited to shifting their point of view by looking around. Their path was pre-determined and unalterable.

Today, you can find virtual roller coasters that use the same sort of technology. DisneyQuest in Orlando, Florida features CyberSpace Mountain, where patrons can design their own roller coaster, then enter a simulator to ride

their virtual creation. The system is very immersive, but apart from the initial design phase there isn't any interaction, so it's not an example of a true virtual environment.

Interactivity depends on many factors. Steuer suggests that three of these factors are **speed**, **range** and **mapping**. Steuer defines speed as the rate that a user's actions are incorporated into the computer model and reflected in a way the user can perceive. Range refers to how many possible outcomes could result from any particular user action. Mapping is the system's ability to produce natural results in response to a user's actions.

Navigation within a virtual environment is one kind of interactivity. If a user can direct his own movement within the environment, it can be called an interactive experience. Most virtual environments include other forms of interaction, since users can easily become bored after just a few minutes of exploration. Computer Scientist Mary Whitton points out that poorly designed interaction can drastically reduce the sense of immersion, while finding ways to engage users can increase it. When a virtual environment is interesting and engaging, users are more willing to suspend disbelief and become immersed.

True interactivity also includes being able to modify the environment. A good virtual environment will respond to the user's actions in a way that makes sense, even if it only makes sense within the realm of the virtual environment. If a virtual environment changes in outlandish and unpredictable ways, it risks disrupting the user's sense of telepresence.

The Virtual Reality Headset

Today, most VE systems are powered by normal personal computers. PCs are sophisticated enough to develop and run the software necessary to create virtual environments. Graphics are usually handled by powerful graphics cards originally designed with the video gaming community in mind. The same video card that lets you play World of Warcraft is probably powering the graphics for an advanced virtual environment.

VE systems need a way to display images to a user. Many systems use HMDs, which are headsets that contain two monitors, one for each eye. The images create a stereoscopic effect, giving the illusion of depth. Early HMDs used cathode ray tube (CRT) monitors, which were bulky but provided good resolution and quality, or liquid crystal display (LCD)

monitors, which were much cheaper but were unable to compete with the quality of CRT displays. Today, LCD displays are much more advanced, with improved resolution and color saturation, and have become more common than CRT monitors.

Other VE systems project images on the walls, floor and ceiling of a room and are called Cave Automatic Virtual Environments (CAVE). The University of Illinois-Chicago designed the first CAVE display, using a rear projection technique to display images on the walls, floor and ceiling of a small room. Users can move around in a CAVE display, wearing special glasses to complete the illusion of moving through a virtual environment. CAVE displays give users a much wider field of view, which helps in immersion. They also allow a group of people to share the experience at the same time (though the display would track only one user's point of view, meaning others in the room would be passive observers). CAVE displays are very expensive and require more space than other systems.

Closely related to display technology are tracking systems. Tracking systems analyze the orientation of a user's point of view so that the computer system sends the right images to the visual display. Most systems require a user to be tethered with cables to a processing unit, limiting the range of motions available to him. Tracker technology developments tend to lag behind other VR technologies because the market for such technology is mainly VR-focused. Without the demands of other disciplines or applications, there isn't as much interest in developing new ways to track user movements and point of view.

Input devices are also important in VR systems. Currently, input devices range from controllers with two or three buttons to electronic gloves and voice recognition software. There is no standard control system across the discipline. VR scientists and engineers are continuously exploring ways to make user input as natural as possible to increase the sense of telepresence. Some of the more common forms of input devices are:

- Joysticks
- Force balls/tracking balls
- Controller wands
- Datagloves
- Voice recognition
- Motion trackers/bodysuits
- Treadmills

Virtual Reality Games

Scientists are also exploring the possibility of developing biosensors for VR use. A biosensor can detect and interpret nerve and muscle activity. With a properly calibrated biosensor, a computer can interpret how a user is moving in physical space and translate that into the corresponding motions in virtual space. Biosensors may be attached directly to the skin of a user, or may be incorporated into gloves or bodysuits. One limitation to biosensor suits is that they must be custom made for each user or the sensors will not line up properly on the user's body.

Mary Whitton, of UNC-Chapel Hill, believes that the entertainment industry will drive the development of most VR technology going forward. The video game industry in particular has contributed advancements in graphics and sound capabilities that engineers can incorporate into virtual reality systems' designs. One advance that Whitton finds particularly interesting is the Nintendo Wii's wand controller. The controller is not only a commercially available device with some tracking capabilities; it's also affordable and appeals to people who don't normally play video games. Since tracking and input devices are two areas that traditionally have fallen behind other VR technologies, this controller could be the first of a new wave of technological advances useful to VR systems.

Some programmers envision the Internet developing into a three-dimensional virtual space, where you navigate through virtual landscapes to access information and entertainment. Web sites could take form as a three-dimensional location, allowing users to explore in a much more literal way than before. Programmers have developed several different computer languages and Web browsers to achieve this vision. Some of these include:

- **Virtual Reality Modeling Language (VRML)**- the earliest three-dimensional modeling language for the Web.
- **3DML** - a three-dimensional modeling language where a user can visit a **spot** (or Web site) through most Internet browsers after installing a plug-in.
- **X3D** - the language that replaced VRML as the standard for creating virtual environments in the Internet.
- **Collaborative Design Activity (COLLADA)** - a format used to allow file interchanges within three-dimensional programs.

Of course, many VE experts would argue that without an HMD, Internet-based systems are not true virtual environments. They lack critical elements of immersion, particularly tracking and displaying images as life-sized.

Virtual Reality History

The concept of virtual reality has been around for decades, even though the public really only became aware of it in the early 1990s. In the mid 1950s, a cinematographer named Morton Heilig envisioned a theatre experience that would stimulate all his audiences' senses, drawing them in to the stories more effectively. He built a single user console in 1960 called the **Sensorama** that included a stereoscopic display, fans, odor emitters, stereo speakers and a moving chair. He also invented a head mounted television display designed to let a user watch television in 3-D. Users were passive audiences for the films, but many of Heilig's concepts would find their way into the VR field.

Philco Corporation engineers developed the first HMD in 1961, called the **Headsight**. The helmet included a video screen and tracking system, which the engineers linked to a closed circuit camera system. They intended the HMD for use in dangerous situations -- a user could observe a real environment remotely, adjusting the camera angle by turning his head. Bell Laboratories used a similar HMD for helicopter pilots. They linked HMDs to infrared cameras attached to the bottom of helicopters, which allowed pilots to have a clear field of view while flying in the dark.

In 1965, a computer scientist named Ivan Sutherland envisioned what he called the "**Ultimate Display**." Using this display, a person could look into a virtual world that would appear as real as the physical world the user lived in. This vision guided almost all the developments within the field of virtual reality. Sutherland's concept included:

- A virtual world that appears real to any observer, seen through an HMD and augmented through three-dimensional sound and tactile stimuli
- A computer that maintains the world model in real time
- The ability for users to manipulate virtual objects in a realistic, intuitive way

In 1966, Sutherland built an HMD that was tethered to a computer system. The computer provided all the graphics for the display (up to this point, HMDs had only been linked to cameras). He used a suspension system to hold the HMD, as it was far too heavy for a user to support comfortably. The HMD could display images in stereo, giving the illusion of

depth, and it could also track the user's head movements so that the field of view would change appropriately as the user looked around.

Virtual Reality Development

NASA, the Department of Defense and the National Science Foundation funded much of the research and development for virtual reality projects. The CIA contributed \$80,000 in research money to Sutherland. Early applications mainly fell into the vehicle simulator category and were used in training exercises. Because the flight experiences in simulators were similar but not identical to real flights, the military, NASA, and airlines instituted policies that required pilots to have a significant lag time (at least one day) between a simulated flight and a real flight in case their real performance suffered.

For years, VR technology remained out of the public eye. Almost all development focused on vehicle simulations until the 1980s. Then, in 1984, a computer scientist named Michael McGreevy began to experiment with VR technology as a way to advance **human-computer interface (HCI)** designs. HCI still plays a big role in VR research, and moreover it led to the media picking up on the idea of VR a few years later.

Jaron Lanier coined the term Virtual Reality in 1987. In the 1990s, the media latched on to the concept of virtual reality and ran with it. The resulting hype gave many people an unrealistic expectation of what virtual reality technologies could do. As the public realized that virtual reality was not yet as sophisticated as they had been led to

believe, interest waned. The term virtual reality began to fade away with the public's expectations. Today, VE developers try not to exaggerate the capabilities or applications of VE systems, and they also tend to avoid the term virtual reality.

For nearly 30 years, ever since Pong was introduced to arcades in the early 1970s. Computer graphics have become much more sophisticated since then, and game graphics are pushing the barriers of photorealism. Now, researchers and engineers are pulling graphics out of your television screen or computer display and integrating them into real-world environments. This new technology, called **augmented reality**, blurs the line between what's real and what's computer-generated by enhancing what we see, hear, feel and smell.

On the spectrum between virtual reality, which creates immersive, computer-generated environments, and the real world, augmented reality is closer to the real world. Augmented reality adds graphics, sounds, haptic feedback and smell to the natural world as it exists. Both video games and cell phones are driving the development of augmented reality. Everyone from tourists, to soldiers, to someone looking for the closest subway stop can now benefit from the ability to place computer-generated graphics in their field of vision.

Augmented reality is changing the way we view the world -- or at least the way its users see the world. Picture yourself walking or driving down the street. With augmented-reality displays, which will eventually look much like a normal pair of glasses, informative graphics will appear in your field of view, and audio will coincide with whatever you see. These enhancements will be refreshed continually to reflect the movements of your head. Similar devices and applications already exist, particularly on smartphones like the iPhone.

Augmenting Our World

The basic idea of augmented reality is to superimpose graphics, audio and other sensory enhancements over a real-world environment in real time. Sounds pretty simple. Besides, haven't television networks been doing that with graphics for decades? However, augmented reality is more advanced than any technology you've seen in television broadcasts, although some new TV effects come close, such as RACEf/x and the super-

imposed first down line on televised U.S. football games, both created by Sportvision. But these systems display graphics for only one point of view. Next-generation augmented-reality systems will display graphics for each viewer's perspective.

Some of the most exciting augmented-reality work is taking place in research labs at universities around the world. In February 2009, at the TED conference, Pattie Maes and Pranav Mistry presented their augmented-reality system, which they developed as part of MIT Media Lab's Fluid Interfaces Group. They call it SixthSense, and it relies on some basic components that are found in many augmented reality systems:

- Camera
- Small projector
- Smartphone
- Mirror

These components are strung together in a lanyardlike apparatus that the user wears around his neck. The user also wears four colored caps on the fingers, and these caps are used to manipulate the images that the projector emits.

SixthSense is remarkable because it uses these simple, off-the-shelf components that cost around \$350. It is also notable because the projector essentially turns any surface into an interactive screen. Essentially, the device works by using the camera and mirror to examine the surrounding world, feeding that image to the phone (which

processes the image, gathers GPS coordinates and pulls data from the Internet), and then projecting information from the projector onto the surface in front of the user, whether it's a wrist, a wall, or even a person. Because the user is wearing the camera on his chest, SixthSense will augment whatever he looks at; for example, if he picks up a can of soup in a grocery store, SixthSense can find and project onto the soup information about its ingredients, price, nutritional value -- even customer reviews.

By using his capped fingers -- Pattie Maes says even fingers with different colors of nail polish would work -- a user can perform actions on the projected information, which are then picked up by the camera and processed by the phone. If he wants to know more about that can of soup than is projected on it, he can use his fingers to interact with the projected image and learn about, say, competing brands. SixthSense can also recognize complex gestures -- draw a circle on your wrist and SixthSense projects a watch with the current time.

Augmented Reality on Cell Phones

While it may be some time before you buy a device like SixthSense, more primitive versions of augmented reality are already here on some cell phones, particularly in applications for the iPhone and phones with the Android operating system. In the Netherlands, cell phone owners can download an application called Layar that uses the phone's camera and GPS capabilities to gather information about the surrounding area. Layar then shows information about restaurants or other sites in the area, overlaying this information on the phone's screen. You can even point the phone at a building, and Layar will tell you if any companies in that building are hiring, or it might be able to find photos of the building on Flickr or to locate its history on Wikipedia.

Layar isn't the only application of its type. In August 2009, some iPhone users were surprised to find an augmented-reality "easter egg" hidden within the Yelp application. Yelp is known for its user reviews of restaurants and other businesses, but its hidden augmented-reality component, called Monocle, takes things one step further. Just start up the Yelp app, shake your iPhone 3GS three times and Monocle activates. Using your phone's GPS and compass, Monocle will display information about local restaurants, including ratings and reviews, on

your cell phone screen. You can touch one of the listings to find out more about a particular restaurant.

There are other augmented reality apps out there for the iPhone and other similar phones -- and many more in development. Urbanspoon has much of the same functionality as Yelp's Monocle. Then there's Wikitude, which finds information from Wikipedia about sites in the area. Underlying most of these applications are a phone's GPS and compass; by knowing where you are, these applications can make sure to offer information relevant to you. We're still not quite at the stage of full-on image recognition, but trust us, people are working on it.

We've looked at some of the existing forms of augmented reality. On the next page, we'll examine some of the other applications of the technology, such as in video games and military hardware.

Augmented Reality in Video Games and the Military

Video game companies are quickly hopping aboard the augmented-reality locomotive. A company called Total Immersion makes software that applies augmented reality to baseball cards. Simply go online, download the Total Immersion software and then hold up your baseball card to a webcam. The software recognizes the card (and the player on it) and then displays related video on your computer screen. Move the card in your hands -- make sure to keep it in view of the camera -- and the 3-D figure on your screen will perform actions, such as throwing a ball at a target.

Total Immersion's efforts are just the beginning. In the next couple of years, we'll see games that take augmented reality out into the streets. Consider a scavenger-hunt game that uses virtual objects. You could use your phone to "place" tokens around town, and participants would then use their phones (or augmented-reality enabled goggles) to find these invisible objects.

Demos of many games of this order already exist. There's a "human Pac-Man" game that allows users to chase after each other in real life while wearing goggles that make them look like characters in Pac-Man.

Arcane Technologies, a Canadian company, has sold augmented-reality devices to the U.S. military. The company produces a head-mounted display -- the sort of device that was supposed to bring us virtual reality -- that superimposes information on your world. Consider a squad of soldiers in Afghanistan, performing reconnaissance on an opposition hideout. An AR-enabled head-mounted display could overlay blueprints or a view from a satellite or overheard drone directly onto the soldiers' field of vision.

Now that we've established some of the many current and burgeoning uses of augmented reality, let's take a look at the technology's limitations and what the future holds.

Limitations and the Future of Augmented Reality

Augmented reality still has some challenges to overcome. For example, GPS is only accurate to within 30 feet (9 meters) and doesn't work as well indoors, although improved image recognition technology may be able to help

People may not want to rely on their cell phones, which have small screens on which to superimpose information. For that reason, wearable devices like SixthSense or augmented-reality capable contact lenses and glasses will provide users with more convenient, expansive views of the world around them. Screen real estate will no longer be an issue. In the near future, you may be able to play a real-time strategy game on your computer, or you can invite a friend over, put on your AR glasses, and play on the tabletop in front of you.

There is such a thing as too much information. Just as the "CrackBerry" phenomenon and Internet addiction are concerns, an overreliance on augmented reality could mean that people are missing out on what's right in front of them. Some people may prefer to use their AR iPhone applications rather than an experienced tour guide, even though a tour guide may be able to offer a level of interaction, an experience and a personal touch unavailable in a computer program. And there are times when a real plaque on a building is preferable to a virtual one, which would be accessible only by people with certain technologies.

There are also privacy concerns. Image-recognition software coupled with AR will, quite soon, allow us to point our phones at people, even strangers. With most of these services people willingly put information about themselves online, but it may be an unwelcome shock to meet someone, only to have him instantly know so much about your life and background.

Despite these concerns, imagine the possibilities: you may learn things about the city you've lived in for years just by pointing your AR-enabled phone at a nearby park or building. If you work in construction, you can save on materials by using virtual markers to designate where a beam should go or which structural support to inspect. Paleontologists working in shifts to assemble a dinosaur skeleton could leave virtual "notes" to team members on the bones themselves, artists could produce virtual graffiti and doctors could overlay a digital image of a patient's X-rays onto a mannequin for added realism.

The future of augmented reality is clearly bright, even as it already has found its way into our cell phones and video game systems.

The Mirage Effect: Carbon Nanotubes

First, let's try this carbon nanotube invisibility cloak on for size and experience the wonders of the mirage effect.

You're probably most familiar with mirages from tales of desert wanderers who glimpse a distant oasis, only to discover it was only a mirage -- no miraculous lake of drinking water, only more hot sand.

The hot sand is key to the **mirage effect** (or **photothermal deflection**), as the stiff temperature difference between sand and air bends, or refracts, light rays. The refraction swings the light rays up toward the viewer's eyes instead of bouncing them off the surface. In the classic example of the desert mirage, this effect causes a "puddle" of sky to appear on the ground, which the logical (and thirsty) brain interprets as a pool of water. You've probably seen similar effects on hot roadway surfaces, with distant stretches of the road appearing to gleam with pooled water.

In 2011, researchers at the University of Texas at Dallas NanoTech Institute managed to capitalize on this effect. They used sheets of **carbon nanotubes**, sheets of carbon wrapped up into cylindrical tubes. Each page is barely as thick as a single molecule, yet is as strong as steel because the carbon atoms in each tube are bonded incredibly tightly. These sheets are also excellent conductors of heat, making them ideal mirage-makers.

In the experiment, the researchers heated the sheets electrically, which transferred the heat to the surrounding area (a petri dish of water). As you can see from the photographs, this caused light to bend away from the carbon nanotube sheet, effectively cloaking anything behind it with invisibility.

Needless to say, there aren't many places you'd want to wear a tiny, super-heated outfit that has to stay immersed in water, but the experiment demonstrates the potential for such materials. In time, the research may enable not only invisibility cloaks but also other light-bending devices -- all of them with a handy on/off switch.

Metamaterials: Bending Light Waves

Next, let's slip into an invisibility cloak made from metamaterials.

Metamaterials offer a more compelling vision of invisibility technology, without the need for multiple projectors and cameras. First conceptualized by Russian physicist Victor Veselago in 1967, these tiny, artificial structures are smaller than the wavelength of light (they have to be to divert them) and exhibit negative electromagnetic properties that affect how an object interacts with electromagnetic fields.

Natural materials all have a **positive refractive index**, and this dictates how light waves interact with them. Refractivity stems in part from chemical composition, but internal structure plays an even more important role. If we alter the structure of a material on a small enough scale, we can change the way they refract incoming waves -- even forcing a switch from positive to negative refraction.

Remember, images reach us via light waves. Sounds reaches us via sound waves. If you can channel these waves around an object, you can effectively hide it from view or sound. Imagine a small stream. If you stick a teabag full of red dye into the flowing water, its presence would be apparent downstream, thanks to the way it altered the water's hue, taste and smell. But what if you could divert the water around the teabag?

In 2006, Duke University's David Smith took an earlier theory posed by English theoretical physicist John Pendry and used it to create a metamaterial capable of distorting the flow of microwaves. Smith's metamaterial fabric consisted of concentric rings containing electronic microwave distorters. When activated, they steer frequency-specific microwaves around the central portion of the material.

Obviously humans don't see in the microwave spectrum, but the technology demonstrated that energy waves could be routed around an object. Imagine a cloak that can divert a third grader's straw-fired spitball, move it around the wearer and allow it to continue on the other side as if its trajectory had taken it, unopposed, straight through the person in the cloak. Now how much more of a stretch would it be to divert a rock? A bullet?

Smith's metamaterials proved the method. The recipe to invisibility lay in adapting it to different waves.

Metamaterials: Invisible Tanks

In 2007, the University of Maryland's Igor Smolyaninov led his team even farther down the road to invisibility. Incorporating earlier theories proposed by Purdue University's Vladimir Shaleev, Smolyaninov constructed a metamaterial capable of bending visible light around an object.

A mere 10 micrometers wide, the Purdue cloak uses concentric gold rings injected with polarized cyan light. These rings steer incoming light waves away from the hidden object, effectively making it invisible. Chinese physicists at Wuhan University have taken this concept into the audible range, proposing the creation of an acoustic invisibility cloak capable of diverting sound waves around an object.

For the time being, metamaterial invisibility cloaks are somewhat limited. They're not only small; they're limited to two dimensions -- hardly what you'd need to vanish into the scenery of a 3-D war zone. Plus, the resulting cloak would weigh more than even a full-grown wizard could hope to lug around. As a result, the technology might be better suited to applications such as hiding stationary buildings or vehicles, such as a tank.

Optical Camouflage: Altered Reality

Ready to slip into some old-school optical camouflage fashions?

This technology takes advantage of something called augmented-reality technology -- a type of technology first pioneered in the 1960s by Ivan Sutherland and his students at Harvard University and the University of Utah.

Optical camouflage delivers a similar experience to Harry Potter's invisibility cloak, but using it requires a slightly complicated arrangement. First, the person who wants to be invisible (let's call him Harry) dons a garment that resembles a hooded raincoat. The garment is made of a special material that we'll examine more closely in a moment.

Next, an observer (let's call him Professor Snape) stands before Harry at a specific location. At that location, instead of seeing Harry wearing a hooded raincoat, Snape sees right through the cloak, making Harry appear to be invisible. The above photograph shows you what Snape would see. And if Snape stepped to the side and viewed Harry from a slightly different location? Why, he'd simply see the boy wizard wearing a silver garment. Scowls and detentions would likely follow. Lucky for Harry, his fictional cloak affords 360-degree protection.

Optical camouflage doesn't work by way of magic. It works by taking advantage of something called **augmented-reality technology** -- a type of technology first pioneered in the 1960s by Ivan Sutherland and his students at Harvard University and the University of Utah. You can read more about augmented reality in [How Augmented Reality Works](#), but a quick recap will be helpful here.

Augmented-reality systems add computer-generated information to a user's sensory perceptions. Imagine, for example, that you're walking down a city street. As you gaze at sites along the way, additional information appears to enhance and enrich your normal view. Perhaps it's the day's specials at a restaurant or the showtimes at a theater

or the bus schedule at the station. What's critical to understand is that augmented reality is not the same as virtual reality. While virtual reality aims to replace the world, augmented reality merely tries to supplement it with additional, helpful content. Think of it as a heads-up display (HUD) for everyday life.

Most augmented-reality systems require a user to look through a special viewing apparatus to see a real-world scene enhanced with synthesized graphics. They also call for a powerful computer. Optical camouflage requires these things as well, but it also necessitates several other components. Here's everything needed to make a person appear invisible:

-
-
-
-

a garment made from highly reflective material

a digital video camera

a computer

a

projector

- a special, half-silvered mirror called a combiner

Optical Camouflage: Invisibility Cloak Components

All right, so you have your video camera, computer, projector, combiner and wondrous reflective raincoat. Just how does augmented-reality technology turn this odd shopping list into a recipe for invisibility?

First, let's take a closer look at the raincoat: It's made from retro-reflective material. This high-tech fabric is covered with thousands and thousands of small beads. When light strikes one of these beads, the light rays bounce back exactly in the same direction from which they came.

To understand why this is unique, look at how light reflects off other types of surfaces. A rough surface creates a diffused reflection because the incident (incoming) light rays scatters in many different directions. A perfectly smooth surface, like that of a mirror, creates what is known as a **specular reflection** -- a reflection in which incident light rays and reflected light rays form the exact same angle with the mirror surface.

In retro-reflection, the glass beads act like prisms, bending the light rays by refraction. This causes the reflected light rays to travel back along the same path as the incident light rays. The result: An observer situated at the light source receives more of the reflected light and therefore sees a brighter reflection.

Retro-reflective materials are actually quite common. Traffic signs, road markers and bicycle reflectors all take advantage of retro-reflection to be more visible to people driving at night. The movie screens found in most modern commercial theaters also take advantage of this material because it allows for high brilliance under dark conditions. In optical camouflage, the use of retro-reflective material is critical because it can be seen from far away and outside in bright sunlight -- two requirements for the illusion of invisibility.

Optical Camouflage: Real-world Invisibility Applications

The words "invisibility cloak" tends to summon images of fantastic adventure, magical espionage and otherworldly deception. The actual applications for optical camouflage, however, are far less out there. You can forget hiding your Romulan starship or hanging out in the lady wizards' dormitory, but that doesn't mean there aren't a number of viable uses for the technology.

For instance, pilots landing a plane could use this technology to make cockpit floors transparent. This would enable them to see the runway and the landing gear simply by glancing down at the floor (which would display the view from the outside of the fuselage) Similarly, drivers wouldn't have to deal with mirrors and blind spots. Instead, they

could just "look through" the entire rear of the vehicle. The technology even boasts potential applications in the medical field, as surgeons could use optical camouflage to see through their hands and instruments for an unobstructed view of the underlying tissue.

Interestingly enough, one possible application of this technology actually revolves around making objects more visible. The concept is called **mutual telexistence** and essentially involves projecting a remote user's appearance onto a robot coated in retro-reflective material. Say a surgeon were operating on a patient via remote control robotic surgery. Mutual telexistence would provide the human doctors assisting the procedure with the perception that they're working with another human instead of a machine.

Right now, mutual telexistence is science fiction, but scientists continue to push the boundaries of the technology. For example, pervasive gaming is already becoming a reality. **Pervasive gaming** extends gaming experiences out into the real world, whether on city streets or in remote wilderness. Players with mobile displays move through the world while sensors capture information about their environment, including their location. This information delivers a gaming experience that changes according to where users are and what they are doing.

Some Applications of Virtual Reality

Imagine the following academic fiction:

Eighteen professors from five departments decide to work together and submit a request for a virtual reality system. Suppose further that the administration actually believes that this is a wonderful idea and approves the proposal, provided that the virtual reality system is put to use in the classroom. The faculty eagerly agree to this condition, and to their amazement they acquire the funds to purchase an SGI Onyx 2 Reality Engine and 10 SGI Indigos.

The above scenario is not some introduction to a John Grisham suspense novel, but a real story at Clemson University. Recently Steve (D.E.) Stevenson from the Department of Computer Science at Clemson University came to the Geometry Center and talked about applications of Geometry with computers. Steve mentioned briefly how various

departments had been using the virtual reality system they acquired, and showed specific examples of what they had done with them.

The departments using the system range from those which traditionally might use virtual reality, such as the Computer Science department, the Mechanical Engineering department and the Architecture department, to fields not generally associated with the technology such as the Biomedical Engineering department and the Performing Arts department. All these disciplines' projects use the technology in ways that create images and objects that otherwise would take a long time to construct, or not be feasible to construct at all.

In particular, software is currently under development for Mechanical Engineering students that extends CAD/CAE software to virtual reality. Instead of clicking keystrokes to try to alter perspective views, a user is able to wear a helmet and by moving their head around are able to view an object as if it were before them. Moreover one is able to look through different layers of an object to view how the device is operating internally. Although these are all things that CAD/CAE software allows, the virtual reality system gives a user a more natural way to view an object, which accordingly allows one to easier ask the question, "what if?"

Some of the other projects involving engineering are simulation-based design, multipurpose design optimization and visualization in High Performance Computing-Computer Formulated Design structures. Lastly one professor dreams of creating a simulation of the famous Tacoma Narrows bridge collapsing so that Civil and Mechanical Engineers can fully appreciate the consequences of their errors.

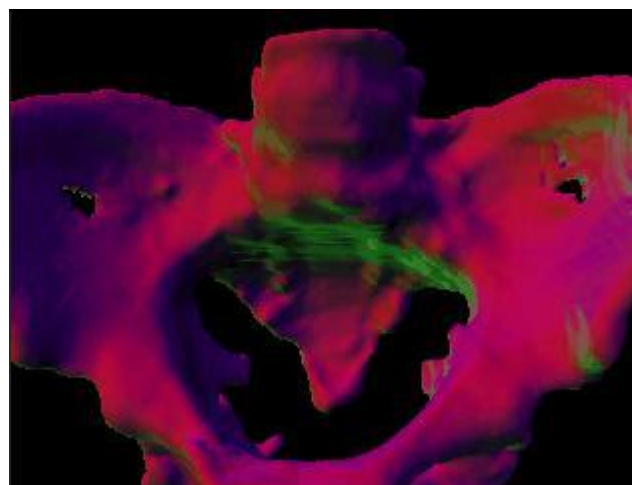
In the Biomedical Engineering department some of the projects mentioned are use of virtual reality for viewing of X-RAY's and MRI's, using stereolithography to make prototypes of joints, and even having students perform test surgery.

In the Computer Science department some of the projects range from creating a toolkit for non-computer science designers, rendering and 3-D lighting, viewing non-euclidean geometries, and modeling for resource management.

Projects in the Architecture department include creating a virtual reality model of campus, and a laboratory on building design.

People in the Performing Arts department use virtual reality for Stage Lighting and Stage Design Courses.

Of the above projects, two of the more interesting applications common to both Mechanical Engineering and Biomedical Engineering, involve stereolithography or 3D printing. One is able to design or input given data about an object and actually create a prototype made out of polymers of the object viewed in the virtual reality. One interesting example is that of an image of a Pelvis taken from an MRI, piped into the virtual reality software so that one is able to view it, and then a model of the bone is manufactured using the polymer machine. The following figure is a virtual reality image of this pelvis.



Similarly, a model of a "ship in a bottle" was created using CAD/CAE software viewed through the virtual reality software, and then made.



The virtual reality machines nicely compliment the polymer machine. One is able to thoroughly view an object before making a prototype, thus saving on the production costs of making a prototype.

The Computer Science department has also created some interesting programs. Two software programs are titled Steve's Room and Oliver's Room. Steve's Room is a program which allows the user via the helmet to look around a room, turn on lights, and place objects by voice or mouse commands. Oliver's Room also is a high resolution room. In this room, one can see in high resolution, an Impressionist painting on the wall, a tiled floor, and a window with a view of mountains. The following picture is a view of Oliver's Room.



As with Steve's Room, the user is able via voice commands to move about the room. The next picture is an image of what one might see through the helmet after a request to move has been made.



The visual results from these projects are amazing, both in a practical sense and in a pure aesthetic sense. The images created are useful in understanding the structure of an object, as well as being suitable for framing. However, what is equally impressive is that various departments were able to get together and pool their resources so that this system could be acquired. By doing this, they have provided themselves, and more importantly, their students, an opportunity to use computer systems today that will no doubt be commonplace in the future.

Military applications of virtual reality

Author: Jim Baumann

One of the first areas where virtual reality found practical application is in military training and operations. In this article, we will explore three views of military applications of virtual

reality; as a simulation of reality, as an extension of human senses through telepresence, and as an information enhancer through augmented reality.

Virtual Reality in military simulations

One of the earliest uses of simulators in a military environment was the flight trainers built by the Link Company in the late 1920's and 1930's. These trainers looked like sawed-off coffins mounted on a pedestal, and were used to teach instrument flying. The darkness inside the trainer cockpit, the realistic readings on the instrument panel, and the motion of the trainer on the pedestal combined to produce a sensation similar to actually flying on instruments at night. The Link trainers were very effective tools for their intended purpose, teaching thousands of pilots the night flying skills they needed before and during World War II.

To move beyond the instrument flying domain, simulator designers needed a way to produce a view of the outside world. The first example of a simulator with an outside view appeared in the 1950's, when television and video cameras became available. With this equipment, a video camera could be 'flown' over a scale model of the terrain around an airport, and the resulting image was sent to a television monitor placed in front of the pilot in the simulator. His movement of the control stick and throttle produced corresponding movement of the camera over the terrain board. Now the pilot could receive visual feedback both inside and outside the cockpit.

The logical extension of the video camera/television monitor approach was to use multiple monitors to simulate the entire field of view from the airplane cockpit. This method is still in use for transport aircraft simulators, where the field of view needs to be only about 180 degrees horizontally and 60 degrees vertically. For fighter aircraft simulators, the field of view must be at least 180 degrees horizontally and vertically. For these applications, the simulator consists of a cockpit placed at the center of a domed room, and the virtual images are projected onto the inside surface of the dome. These types of simulators have proven to be very effective training aids by themselves, and the newest innovation is a project called SIMNET to electronically connect two or more simulators to produce a distributed simulation environment. [McCarty, 1993] Distributed simulations can be used not only for training, but to develop and test new combat strategy and tactics. A significant development in this area is an IEEE data protocol standard for distributed interactive simulations. [IEEE, 1993] This standard allows the distributed simulation to include not only aircraft, but also land-based vehicles and ships. Another recent development is the use of head-mounted displays (HMDs) to decrease the cost of wide field of view simulations. [McCarty, 1993]

Telepresence for military missions

Two fairly obvious reasons have driven the military to explore and employ telepresence in their operations; to reduce exposure to hazards and to increase stealth. Many aspects of combat operations are very hazardous, and they become even more dangerous if the combatant seeks to improve his performance. Prime examples of this principle are firing weapons and performing reconnaissance. To perform either of these tasks well takes time, and this is usually time when the combatant is exposed to hostile fire. Smart weapons and remotely- piloted vehicles (RPVs) were developed to address this problem.

Some smart weapons are autonomous, while others are remotely controlled after they are launched. This allows the shooter and weapon controller to launch the weapon and immediately seek cover, thus decreasing his exposure to return fire. In the case of RPVs, the person who controls the vehicle not only has the advantage of being in a safer place, but the RPV can be made smaller than a vehicle that would carry a man, thus making it more difficult for the enemy to detect.

Military information enhancement

In a dynamic combat environment, it is imperative to supply the pilot or tank commander with as much of the necessary information as possible while reducing the amount of distracting information. This goal led the Air Force to develop the head-up display (HUD) which optically combines critical information (altitude, airspeed, heading) with an unobstructed view through the forward windscreen of a fighter aircraft. With the HUD, the pilot never has to look down at his instruments. When the HUD is coupled with the aircraft's radar and other sensors, a synthetic image of an enemy aircraft can be displayed on the HUD to show the pilot where that aircraft is, even though the pilot may not be able to see the actual aircraft with his unaided eyes. This combination of real and virtual views of the outside world can be extended to nighttime operations. Using an infrared camera mounted in the nose of the aircraft, an enhanced view of the terrain ahead of the aircraft can be projected on the HUD. The effect is for the pilot to have a 'daylight' window through which he has both a real and an enhanced view of the nighttime terrain and sky.

In some cases, the pilot may need to focus totally on the virtual information and completely exclude the actual view. Work in this area has been pioneered by Thomas Furness III and others at Wright Laboratories, Wright-Patterson Air Force Base, Ohio. This work, dubbed the Super Cockpit, involved not only a virtual view of the outside world, but also of the cockpit itself, where the pilot would select and manipulate virtual controls using hand gestures. [Furness, 1986]

Anxiety Therapy

For years now, virtual environments have been used to treat anxiety problems with exposure therapy. Psychologists treat phobias and post traumatic stress disorder by exposing the patient to the thing that causes them anxiety and letting the anxiety dissipate on its own. But this proves difficult if your stressor is a battlefield in Iraq.

Enter virtual reality. Military psychologists use simulated Iraq war situations to treat soldiers. Other therapeutic VR uses include treating a fear of flying, fear of elevators, and even a "virtual nicotine craving" simulator for smoking addiction.



VR Training Programs

Virtual reality environments have also been used for training simulators. The earliest examples were flight simulators (most of us probably remember "Microsoft Flight Simulator"), but VR training has expanded beyond just that. There are many modern military examples, including Iraqi cultural situations and battlefield simulators for soldiers. Other examples include counter-terrorism, paratrooping, welding, and mining training sims.



Multiplayer Online Gaming

One result of virtual-reality research is the existence of entirely separate virtual worlds, inhabited entirely by the avatars of real world users. These worlds are sometimes referred to as massively multiplayer online games, and the World of Warcraft is the largest virtual gaming world in use now, with 11.5 million subscribers.

Another example is Second Life. The world of Second Life can't really be classified as a game, since the goal seems really just to be to wander around and interact with people, much like the real world. There is even a Second Life Shakespeare Company that performs Shakespeare's works within Second Life.

(Image: The Second Life Globe Theater, from Pathfinder Linden)



The Nintendo Wii

Probably the most successful cousin of virtual reality on the market today is the Nintendo Wii. The Wii owes its motion capture and intuitive interaction concepts to the virtual reality technologies of the past. The controller is basically a simplified version of the "virtual reality glove." Both the Wiimote and the Wii Fit offer users another way of interacting with their virtual environment without having to wear any bulky equipment.

(Image: a new take on Wii tennis by Mesq)



Medical Procedures

Modern medicine has also found many uses for virtual reality. Doctors can interact with virtual systems to practice procedures or to do tiny surgical procedures on a larger scale. Surgeons have also started using virtual "twins" of their patients, to practice for surgery before doing the actual procedure.

(Image: the Karlsruhe Endoscopic Surgery Trainer)



Project Natal

The latest entry in the virtual reality inspired gaming world is Project Natal, a new piece of technology under development now for the Xbox. Project Natal proposes a new way of interacting with games, and indeed with computer systems in general. In their demo video, they propose a system that requires no keyboard and no controller, where a user's voice and motions serve as their method for interacting with the system.

The demo video is impressive, but the technology has not been completed and released yet. When it does get released, however, virtual reality will take another giant step towards total immersion and common home usage.



The Cave

The term "CAVE" refers to any virtual reality system that uses multiple walls with multiple projectors to immerse users in a virtual world. The first CAVE was built in 1992 as a method of showing of scientific visualizations. Now, many universities have their own CAVE systems. The CAVE is used for visualizing data, for demonstrating 3D environments, and for virtually testing component parts of newly developed engineering projects.

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)

6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

UNIT 2- 3D AUDIO

3D AUDIO

A 3D audio system has the ability to position sounds all around a listener. The sounds are actually created by the loudspeakers (or headphones), but the listener's perception is that the sounds come from arbitrary points in space. This is similar to stereo panning in conventional stereo systems: sounds can be panned to locations between the two loudspeakers, creating virtual or "phantom" images of the sound where there is no loudspeaker. However, conventional stereo systems generally cannot position sounds to the sides or rear of the listener, nor above or below the listener. A 3D audio system attempts to do just that.

A lot of commercial audio products are described as having 3D capability, but in fact there is great disparity between the various technologies in use. Unfortunately, many of the weakest products are marketed with the most exaggerated claims. For example, a number of stereo multimedia speakers are marketed as having "3D" technology. These speakers incorporate a simple circuit that has the effect of widening the perceived soundfield of a stereo recording. That is, the sound images that would normally extend to the locations of the left and right speakers are widened to extend beyond the speakers. These systems should more properly be called stereo enhancement or "widening" systems. They have no ability to position individual sounds around a listener, nor do they have the ability to position sounds behind, above, or below the listener. We use the term 3D audio to describe a much more sophisticated system than can ideally position sounds anywhere around a listener.

Even within the field of what we would consider to be true 3D technology, there is a wide range of technologies in use, with corresponding variation in the performance and cost of the products. Compounding this is the fact that even the best technologies are subject to unavoidable limitations that guarantee that the performance of 3D audio will always fall a bit short of the marketing claims. This has created some confusion in the marketplace.

Nevertheless, 3D technology is rapidly improving, thanks in part to the increasing availability of inexpensive computational power.

To be able to create 3D sound we have to understand the methods that the human being locates sounds sources. At the end of the 19th century, Lord Rayleigh contributed a study addressing this point, and he was one of the pioneers in this field. His studies and others' led to the main points mentioned above. Here are some explanations for them:

Interaural Time Differences (ITD):

The interaural time differences are the least complex of the directional cues that the human auditory system depends on to locate sound sources. As mentioned above, the ITD is the difference in time-of-arrival between the left and right ears depending on the position of the sound source. In the following diagram the sound wave will reach the right ear before a considerable amount of time in which it will reach the left ear

The ITD is useful unless the sound source is in one of the following positions: Directly in front, directly behind, directly above or directly below). In these positions the sound will reach the left ear and the right one at exactly the same time.

Reproducing 3D Sound:

One of the theories that explain reproducing 3D sound in the virtual world states that a sound-producing device should be placed for every sound source to be created in the virtual world. It's obvious that this theory is so complicated and too costly as a sound source should be processed each time the user or one of the virtual objects move in the virtual world. To overcome this problem the time differences needed for determining the source can be processed automatically by our auditory system.

Sound Levels:

Researches found that the sound frequency level plays a major role in determining the sound source using the ITD. For high frequencies, the listener's head creates the "shadow effect" on the sound waves, which is another problem to be dealt with. Sounds with frequencies higher than 4000 Hz suffer from the shadow effect more than smaller frequencies. The reason for this is that the wave length of high frequencies is relatively short which causes more crashes between these waves and the listener's head which obstruct these waves from reaching on of the ears. For these kinds of sounds the IPD is used to determine the sound source.

Note: Research showed that for frequencies between 1000 Hz and 4000 Hz, the shadow effect depends mainly on the listener's perceptual threshold, in other words it differs from one listener to another.

The Engineering Basics of the ITD:

A huge amount of research literature tried to determine the actual delays required for reproducing ITDs. Probably the most accurate set of equations that represent this process is the following:

- For frequencies equal to or less than 4000 Hz, the ITD is calculated as follows:

$$\text{ITD} = (3 \times r \times \sin \emptyset) \div S \text{ Where:}$$

S : the speed of sound.

∅ : the sound source's deviation angle (given that " 0° " degree is the angle directly behind the listener's head, the " 90° " is the left ear's deviation angle and the " 270° " is the right ear's deviation angle).

r : the radius of the listener's head given in cm.

- For frequencies greater than 4000 Hz, the ITD is calculated as follows:

$$\text{ITD} = (2 \times r / S) \times \sin \emptyset$$

To fully understand these formulas, let's assume the followings:

The Speed Of Sound

The listener's head radius

The sound source's angle deviation of the listener's head

$S = 343 \text{ m/sec}$

$r = 9 \text{ cm}$

$\theta = 45^\circ$

In the given conditions, the ITD for frequencies less or equal to 4000 Hz is calculated as follows:

$$\text{ITD} = (3 \times 9 \times \sin 45) / 343 = 0.05566 \text{ sec}$$

This means that a delay of (0.05566 sec) has to occur between the instance the sound reaches the left ear and the instance the sound reaches the right ear.

To simulate this process in our virtual environment, we have to generate a sound from the right sound speaker and then generate the same sound from the left sound speaker after 0.05566 sec. It is clear that if the sound source was located on an angle deviation of " 45° " on the left of the listener, then the angle " θ " that should be used in the formula above is " 315° " and the result will be the same but in negative value.

How does 3D Audio Work?

To answer how 3D audio systems work, it is useful to start by considering how humans can localize sounds using only two ears. A sound generated in space creates a sound wave that propagates to the ears of the listener. When the sound is to the left of the listener, the sound reaches the left ear before the right ear, and thus the right ear signal is delayed with respect to the left ear signal. In addition,

the right ear signal will be attenuated because of “shadowing” by the head. Both ear signals are also subject to a complicated filtering process caused by acoustical interaction with the torso, head, and in particular, the pinna (external ear). The various folds in the pinna modify the frequency content of the signals, reinforcing some frequencies and attenuating others, in a manner that depends on the direction of the incident sound. Thus an ear acts like a complicated tone control that is direction dependent. We unconsciously use the time delay, amplitude difference, and tonal information at each ear to determine the location of the sound. These indicators are called sound localization “cues”. Sound localization

by human listeners has been studied extensively.

The transformation of sound from a point in space to the ear canal can be measured accurately; the measurements are called head-related transfer functions (HRTFs). The measurements are usually made by inserting miniature microphones into the ear canals of a human subject or a manikin. A measurement signal is played by a loudspeaker and recorded by the microphones. The recorded signals are then processed by a computer to derive a pair of HRTFs (for the left and right ears) corresponding to the sound source location. This process is diagrammed in figure 1. Each HRTF, typically consisting of several hundred numbers, describes the time delay, amplitude, and tonal transformation for the particular sound source location to the left or right ear of the subject. The measurement procedure is repeated for many locations of the sound source relative to the head, resulting in a database of hundreds of HRTFs that describe the sound transformation characteristics of a particular head.

A 3D audio system works by mimicking the process of natural hearing, essentially reproducing the sound localization cues at the ears of the listener. This is most easily done by using a pair of measured HRTFs as a specification for a pair of digital audio filters (equalizers). When a sound signal is processed by the digital filters and listened to over headphones, the sound localization cues for each ear are reproduced, and the listener should perceive the sound at the location specified by the HRTFs. This process is called binaural synthesis (binaural signals are defined as the signals at the ears of a listener).

Binaural synthesis works extremely well when the listener's own HRTFs are used to synthesize the localization cues. However, measuring HRTFs is a complicated procedure, so 3D audio systems typically use a single set of HRTFs previously measured from a particular human or manikin subject.

Localization performance generally suffers when a listener listens to directional cues synthesized from HRTFs measured from a different head, called non-individualized HRTFs. Human heads are all different sizes and shapes, and there is also great variation in the size and shape of individual pinna. This means that every individual has a different set of directional cues. The greatest differences are

in the tonal transformations at high frequencies caused by the pinna. It is clear we become accustomed to localizing with our own ears, and thus our localization

abilities are diminished when listening through another person's ears. Our uniqueness as individuals is the source of the greatest limitation of 3D technology. The use of non-individualized HRTFs results in two particular kinds of localization errors commonly seen with 3D audio systems: front/back confusions and elevation errors. A front/back confusion results when the listener perceives the sound to be in the front when it should be in back, and vice-versa. When 3D audio is reproduced over frontal loudspeakers, back to front confusions tend to be common, which simply means that some listeners may not be able to perceive sounds as being in the rear. In practice, this means that when panning a sound from the front, around to the side, and to the rear, the result will be perceived as a sound panning to the side and then back to the front.

Elevation errors are also common with 3D audio systems. In practice, when a sound is moved from directly to the right to directly overhead, this may be perceived as though the sound is moving from the right to directly in front. This is a typical manifestation of elevation errors, commonly observed when using loudspeakers. Elevation performance is much better when using headphones than when using loudspeakers because the high frequency cues are more faithfully reproduced.

The HRTFs used by Wave Arts 3D were measured from a Knowles Electronic Manikin for Acoustic Research (KEMAR). The measurements were made in MIT's anechoic chamber. The KEMAR is an anthropomorphic manikin whose dimensions were designed to equal those of a median human. The pinna used were molded from human pinna. In total, 710 measurements were made at different locations around the KEMAR. When synthesizing a

location that is not in the measured set, HRTFs from four adjacent locations are interpolated.

Notes

How does 3D audio work over loudspeakers?

When reproducing localization cues to a listener, it is important that the left and right audio channels remain separated, that is, the left ear signal should go to the listener's left ear only, and the right ear signal should go to the listener's right ear only. This is easy to achieve when the listener is using headphones. When using loudspeakers, however, there is significant "crosstalk" between each speaker and the opposite ear of the listener. A large portion of the left speaker signal will go to the right ear of the listener, and similarly a large portion of the right speaker signal will go to the left ear of the listener. In figure 3, the crosstalk paths are labeled ALR and ARL. The crosstalk severely degrades localization performance and must be eliminated.

Fortunately, it is possible to build an elaborate digital filter, called a "crosstalk canceller," that eliminates crosstalk. The crosstalk canceller adds a cancellation signal to each of the two channels of audio, such that when the listener is properly positioned between the loudspeakers, the crosstalk is acoustically cancelled at the listener's ears. The listener must be centered between the two loudspeakers in order for the crosstalk to be cancelled. In 3D audio parlance, the listener must be in the "sweet spot" to get the full 3D effect. Provided the listener is centered between the loudspeakers, crosstalk cancellation is relatively insensitive to front-back motions of the listener, however, crosstalk cancellation is degraded when the listener is off-center or not facing forward.

Loudspeaker 3D audio systems are extremely effective in desktop computing environments. This is because there is usually only a single listener (the computer user) who is almost always centered between the speakers and facing forward towards the monitor. Thus, the primary user gets the full 3D effect because the crosstalk is properly cancelled. In typical 3D audio applications, like video gaming, friends may gather around to watch. In this case, the best 3D audio effects are heard by others when they are also centered with respect to the

Notes

loudspeakers. Off-center listeners may not get the full effect, but they still hear a high quality stereo program with some spatial enhancements.

Many crosstalk cancellers are based on a highly simplified model of crosstalk, for example modeling crosstalk as a simple delay and attenuation process, or a delay and a lowpass filter (book, trans 3D audio). Other crosstalk cancellers have been based on a spherical head model. The crosstalk canceller used by Wave Arts 3D audio is based on actual HRTF measurements and thus accurately models the crosstalk that occurs with human listeners. For typical human listeners, the Wave Arts crosstalk canceller improves channel separation by about 20 dB in the 100 Hz to 6 kHz range. This may seem like a modest improvement, but in fact it is quite good. Even a small improvement in channel separation leads to a large improvement in localization performance. As with binaural synthesis, crosstalk cancellation performance is ultimately limited by the variation in the size and shape of human heads.

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

UNIT 3- 3D Graphics

3D graphics are graphics that use a three-dimensional representation of geometric data that is stored in the computer for the purposes of performing calculations and rendering images.

In computer graphics software, the distinction between 2D and 3D is occasionally blurred; 2D applications may use 3D techniques to achieve effects such as lighting, and primarily 3D may use 2D rendering techniques.

3D computer graphics are often referred to as 3D models. Apart from the rendered graphic, the model is contained within the graphical data file. However, there are differences. A 3D model is the mathematical representation of any three-dimensional object (either inanimate or living). A model is not technically a graphic until it is visually displayed. Due to 3D printing, 3D models are not confined to virtual space. A model can be displayed visually as a two-dimensional image through a process called 3D rendering, or used in non-graphical computer simulations and calculations.

Stereoscopy (also

called **stereoscopies** or **3-D**

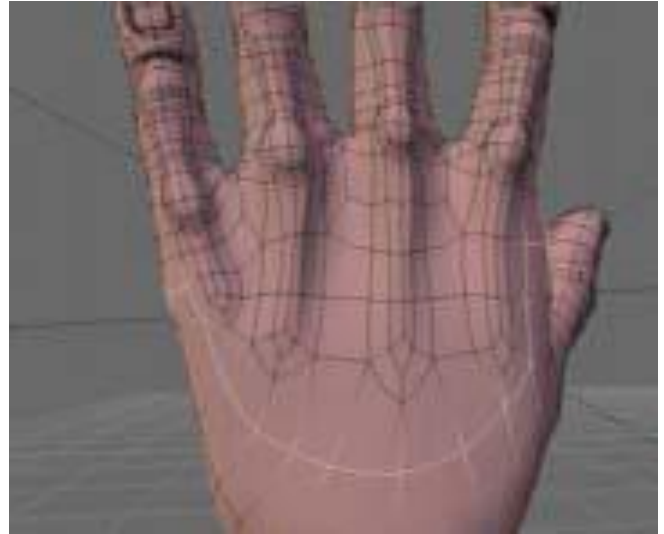
imaging) refers to a technique for creating or enhancing the illusion of depth in an image by presenting two offset images separately to the left and right eye of the viewer. These two-dimensional images are then combined in the brain to give the perception of 3-D depth. Besides the technique of freeviewing, which must be learned by the viewer, three strategies have been used to mechanically present different images to each eye: have the viewer

wear eyeglasses to combine separate images from two offset sources, have the viewer wear eyeglasses to filter offset images from a single source separated to each eye, or have the lightsource split the images directionally into the viewer's eyes (no glasses required; known as Autostereoscopy)

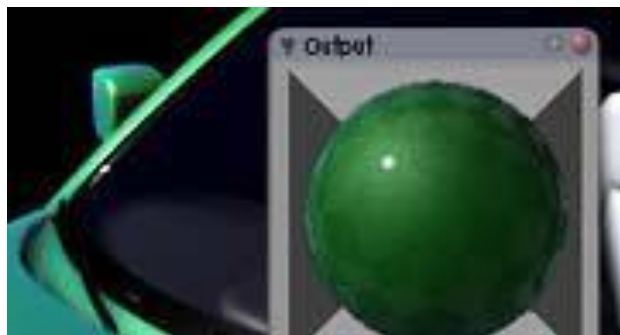
What is Virtual Reality: 3D Graphics

The availability of computer graphics workstations capable of real-time, 3D display at high frame rates is probably the key development behind the current push for realistic virtual environment. In fact, the generation and displaying of three-dimensional objects in a two-dimensional space is a complex process requiring considerable memory and processing power.

The main difference between 2D and 3D computer graphics is the use of 3D representation of geometric data (Schroeder, 1998). In a 2-dimensional graphic scene, each pixel (short for "Picture Element", a single point in a graphic image) has the following properties: position, color, and brightness. When we move to 3D graphics, each pixel requires a new property - depth - that indicates where the point lies on an imaginary Z-axis.



The process of adding depth to an image using a set of cross-sectional images - known as a volumetric dataset - is defined voxelization (from voxel: VOlume piXEL). When many 3-D pixels are combined, each with its own depth value, the result is a three-dimensional surface, called a texture. Once a texture has been defined, it can be wrapped around any 3-dimensional object.



In this context, a key role is played by rendering: the process of adding realism to a computer graphics by

adding three-dimensional qualities such as shadows and variations in color and shade (see Table 1 for the most common rendering techniques).



Rendering in virtual reality, is calculated and displayed in real time, at rates of approximately 20 to 120 frames per second. Animations for non-interactive media, such as video and film, are rendered much more slowly.

Rendering	Description
------------------	--------------------

scanline	is a rendering algorithm that works on a point-by-point basis: some point in a line is calculated, followed by successive points in the line. When the line is finished, rendering proceeds to the next line.
ray tracing	works by tracing the path taken by a ray of light through the scene, and calculating reflection, refraction, absorption of the ray whenever it intersects an object in the world.
radiosity	Unlike ray tracing, which tend to simulate light reflection only once off each surface, it simulates the multiple reflections of light around a scene, generally resulting in softer, more natural shadows and reflection

In the past, 3-D graphics was available only on powerful workstations, but now 3-D graphics accelerator are commonly found in personal computer. To handle the computational operations required by the rendering process a graphics accelerator contains its own memory and a specialized microprocessor.

Coordinate systems

- World coordinate
 1. 3D space defined using X, Y, and Z coordinates
 2. The origin is at 0,0,0
 3. In a right-handed coordinate system, -Z is in the distance

- Object coordinates
 1. Each object has its own coordinate system

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

BCA (DES)306- 3rd YEAR

VIRTUAL REALITY

[Type here]

Block 2

BCA(DES)-306

VIRTUAL REALITY

Copyright © 2018 KnowledgeManagementand Research Organization.

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by means of, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

Acknowledgment

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/ amendment in future edition of this book.

The content is developed by taking reference of online and print

Publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken “as is” and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.’



ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a

Successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual

Involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect,

Students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self- instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual- skills objectives may be met by designing instructions that make

use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive

Communication clues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to r e d u c e

This. Divide and to bring this Self Instructional Material as the b e s t

teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self- instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts.

Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline.

These could be in the form of students 'assignments, a research project

or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you
Expect.

All the best for your studies from our team!

All the best for your studies from our team!

UNIT 4 VR software

There are currently quite a number of different efforts to develop VR technology. Each of these projects have different goals and approaches to the overall VR technology. Large and small University labs have projects underway (UNC, Cornell, U.Rochester, etc.). ARPA , NIST, National Science Foundation and other

branches of the US Government are investing heavily in VR and other simulation technologies. There are industry supported laboratories too, like the Human Interface Technologies Laboratory (HITL) in Seattle and the Japanese NTT project. Many existing and startup companies are also building and selling world building tools (Autodesk, IBM', Sense8, VREAM).

There are two major categories for the available VR software: toolkits and authoring systems. Toolkits are programming libraries, generally for C or C++ that provide a set of functions with which a skilled programmer can create VR applications. Authoring systems are complete programs with graphical interfaces for creating worlds without resorting to detailed programming. These usually include some sort of scripting language in which to

describe complex actions, so they are not really non-programming, just much simpler programming. The programming libraries are generally more flexible and have faster renders than the authoring systems, but you must be a very skilled programmer to use them.

Freeware VR Programs

At the low end of the VR spectrum are the freeware products and garage or home-

brew VR hackers. There are currently a few fast rendering programs that have been released with source code and no charge. These programs are generally copyrighted freeware, which means that the original creators retain the copyright and commercial use is restricted. They are not polished commercial programs, and are often written by students. However, these programs exist to give people a very low cost entry into the VR world.

Rend386 is one such freeware library and world player written for 386/486 DOS systems. It was written by Dave Stampe and Bernie Roehl at the University of Waterloo, Canada. It creates images at a resolution of 320x200x256 and supports various extra devices such as the Mattel PowerGlove, LC shutter glasses, Split Screen stereo viewers etc. Rend386 is provided both as a complete world player and as a C source code. It does not provide a full authoring environment for world and object building. Dave and Bernie co-authored the book "Virtual Reality Creations" with John Eagan. It serves as the primary user documentation for Rend386 Version 5. There is also an electronic mail list for Rend386. Rend386 is available on the via ftp ([sunee.uwaterloo.ca](ftp://sunee.uwaterloo.ca)), CompuServe's CyberForum, and also from a large number of BBSes.

ACK3D is a freeware C programming library developed by Lary Meyer that provides a fast 'raycasting' renderer for PC systems. This technique restricts the user motion somewhat, but allows textures to be drawn at very impressive rates. The technique gained a fair bit of exposure with the Wolfenstein 3D series of shareware games. ACK3D can be found on the CompuServe Gamer's forum and also via ftp from [ftp.u.washington.edu](ftp://ftp.u.washington.edu) in the <pub/virtual-worlds/cheap-vr> area.

Notes

Gossamer

Gossamer is a freeware VR package for the Apple Macintosh system, written by Jon Blossom. Source code has not been released yet, but Jon has released a demo and a Think C library. Jon is currently working on a new version that will support file compatibility with Rend386 V5 and a more extensive user program. The current version is available on via ftp from ftp.apple.com in the directory pub/VR, and also on CompuServe's CyberForum.

Multiverse

It is a freeware UNIX based client/server system written by Robert Grant. It is a multi-user, non-immersive, X-Windows based Virtual Reality system, primarily focused on entertainment/research. It includes capabilities for setting up multi-person worlds and a client/server type world simulation over a local or long haul network. Multiverse source and binaries for several flavors of UNIX are available via anonymous ftp from medg.lcs.mit.edu in the directory pub/multiverse.

The **MRT**oolkit is a programming library for UNIX systems that is available at no cost from the University of Alberta, but the licensing agreement stipulates no commercial products may be made with it.

VEOS is another programming toolkit that provides a basis for VR development on networked UNIX machines. Source code is available from the Human Interface Technology Lab (HITL) at University of Washington. (<ftp.u.washington.edu>).

VR Programs for under \$200

There are a number of commercial VR programs that sell for under \$200. Many computer games that can be considered in this category, such as Wolfenstein 3D, but

these are often closed systems that do not allow much customizing or world building by the user.

Virtual Reality Studio (aka VR Studio, VRS) is a very low cost VR authoring system that does allow the user to define their own virtual worlds. This program is also known as "3D Construction Kit" in Europe. The program has a fairly nice graphical interface and includes a simple scripting language. It is available for about \$100 from Domark for PC and Amiga systems. Worlds created with the program can be freely distributed with a player program. There are a quite number of these worlds available from the BBSes, and other sources. Comuserve's Cyberforum has several in its libraries, like the company provided demo VRSDMO.ZIP (VRS.TXT gives a solution to the demo game). Version 2 of VR Studio was released in early 1993. It has many new features including a much enhanced scripting language and editor, but also an annoying number of bugs. The developers of VRS (Dimension International) are working hard to correct these.

Another entrant into the low cost market is the Lepton VR Data Modeling Toolkit. This package is a collection of C programming libraries for real-time 3d data modeling on DOS systems. Version 1.0 is scheduled to be released in Fal1993 and will cost approximately \$150.

For the Macintosh market there are the Qd3d, 3dPane, and SmartPane C++ libraries from ViviStar Consulting (\$192 for full package). These provide a full suite of 3D

graphics functions for popular Macintosh C++ compilers
as well as Think C 6.0.

VR Packages under \$1000

The next level of VR System is those costing between two hundred and one thousand dollars. There are some very excellent professional packages appearing in this price range in the last year. Most of these systems do not require any specialized hardware beyond the basic computer system.

VREAM is a complete VR authoring package for MS-DOS systems for about \$795 from VREAM, Inc.. It provides a nice GUI environment for creation of objects and worlds, as well as a fairly powerful scripting language. VREAM supports a very wide variety of input and output devices, including HMDs. Two versions of the runtime system are available at a much lower cost to provide only the playback ability. The lower cost runtime (under \$50) will work only with standard VGA display and mouse/joystick. The advanced runtime system supports more devices.

Virtus Walkthrough, from Virtus Corp., is available for both Mac and Windows systems. It provides a nice 3D modeling package and the ability to interactively control the viewpoint within the created worlds. However, it does not allow for interaction with the objects. The latest version Walkthrough Pro supports texture maps, including QuickTime movies.

Sense 8 has announced a \$795 programming library for Windows called World Tool Kit for Windows. This will be released late in 1993 as a DLL for Windows systems. It will work directly with standard SVGA displays and show worlds with texture

mapping either within a window or allow full screen display. The programming library will support DDE so a virtual world can be controlled from a spreadsheet, database or other program.

VR Software for over \$1000

The heavy duty professional VR software packages begin at about \$1000 and can go up dramatically. The hardware required to run these systems varies. Most

support a DOS environment with add in rendering cards like the i860 based SPEA Fireboard. A few work on SGI and other workstation system. There are also other packages available that run on vendor specific hardware configurations. The really high end packages require extremely expensive hardware "Image Generators" such as those used in flight simulators.

The Sense8 World Tool Kit (WTK) is probably the most widely used product of this type. It runs on a wide variety of platforms from i860 assisted PCs to high end SGI boxes. It has won several awards for excellence.

The Autodesk Cyberspace Development kit is another product in this range. It is a C++ library for MSDOS systems using the Metaware HighC/C++ compiler and Pharlap DOS 32bit extender. It supports VESA displays as well as several rendering accelerator boards (SPEA Fireboard, FVS Sapphire, Division's dView). I used this system for a few months and found it requires a strong background in C++ and a rendering accelerator card. VESA speeds were about 4 frames per second.

Straylight Corp. makes a package called PhotoVR that uses special rendering boards (Intel ActionMedia cards) to provide excellent texture mapped walkthrough environments.

Dimension International's Superscape VRT3 is a very powerful authoring system for virtual worlds. It

provides both a graphical environment for object and world creation and a lower level C library.

Division Ltd. sells a programming environment for VR called dVS. This package runs on SGI systems, IBM RS/6000 workstations and a proprietary Division

workstation. They also sell a complete world authoring and simulation program similar to VREAM and VRT3 called dVise.

Lightscape is a radiosity rendering package for creating realistically shaded walkthroughs from Lightscape Graphics Software. This product runs on high end workstations and is aimed primarily at architects and lighting designers.

There have been a number of other packages introduced recently for professional VR development. I do not have full information on all of them and suggest the interested reader follow up by reading either the AI Expert Special Report on Virtual Reality or perhaps by purchasing Sophistech's VR Sourcebook.

Commercial VR Programs

Virtual Reality Studio

(aka VR Studio, VRS) is a very low cost VR authoring system that does allow the user to define their own virtual worlds. This program is also known as "3D Construction Kit" in Europe. The program has a fairly nice graphical interface and includes a simple scripting language. Worlds created with the program can be freely distributed with a player program.

Virtual Reality uses immersive displays and tracking systems to put the user bodily inside interactive three dimensional worlds. At the Virtual Reality Studio, Dept.

of Media Study, UB, we use our systems to explore the design issues, aesthetics, and social ramifications of building responsive simulations for art, architecture, gaming, interactive fiction and education.

We are interested in working with graduate and undergraduate students who want to experiment with simulations, responsive installations, immersive and VR gaming, interactive fiction/drama, the warping of gaming paradigms for art, the expansion of interface beyond monitor and mouse.

VREAM

It is a complete VR authoring package for MS-DOS systems. It provides a nice GUI environment for creation of objects and worlds, as well as a fairly powerful scripting language. VREAM supports a very wide variety of input and output devices, including HMDs.

Virtus Walkthrough , from Virtus Corp., is available for both Mac and Windows systems. It provides a nice 3D modeling package and the ability to interactively control the viewpoint within the created worlds. However, it does not allow for interaction with the objects.

Straylight Corp. makes a package called **PhotoVR** that uses special rendering boards to provide excellent texture mapped walkthrough environments.

Lightscape is a radiosity rendering package for creating realistically shaded walkthroughs from Lightscape Graphics Software. This product runs on high end workstations and is aimed primarily at architects and lighting designers.

VRML

Most exciting is the ongoing development of VRML (Virtual Reality Modeling Language) on the World Wide Web. In addition to HTML (HyperText Markup Language), that has become a standard authoring tool for the creation of home pages, VRML provides three-dimensional worlds with integrated hyperlinks on the Web. Home pages become home spaces. The viewing of VRML models via a VRML plug-in for Web browsers is usually done on a graphics monitor under mouse-control and, therefore, not fully immersive. However, the syntax and data structure of VRML provide an excellent tool for the modeling of three-dimensional worlds that are functional and interactive and that can, ultimately, be transferred into fully immersive viewing systems. The current version VRML 2.0

has become an international ISO/IEC standard under the name VRML97.

Virtual Reality Modelling Language pronounced as vermal and its started to use in year 1995. Initially, this is also called by a name Virtual Really Markup language. This is a standard language which is used for interactive simulation with in the world wide web. It allows to represent 3-dimensional interactive vector graphics and a “virtual worlds” networked via internet and hyper linked with the world wide web.

VRML is text file format Which able to design vertices and edges for 3D polygon with any of the specified color. It is also used to perform UV mapping which create the 3D models of the 2D image. VRML files are commonly called with a name of “worlds” and have .wrl extension. VRML worlds contain the text format so that it easily compressed by using the gzip and compressed file transfer through internet more quickly.

VRML architecture include the Input processor, Simulation processor, Rendering Processor and World Database. Input Processor is used to control the input information to the computer e.g. Keyboard, mouse, 3D position trackers and a voice recognition system. Simulation Processor is the heart of Virtual Reality

system. It takes the user input along with a number of task that are programmed with it and determine the actions that will take place in the virtual environment. Rendering processor is used to create sensation that are output to the user. World Database is also known as World Description Files. It store the objects that describe the actions of all those objects.

VRML is successfully used in various field like Entertainment, Medicine, Manufacturing, Education and training. In Entertainment, it is helpful in designing a more attractive and exciting virtual environment. In

medicine, used to perform the practice test of surgery, surgery on remote patient and teach new skills in a safe, controlled environment. In manufacturing, used to make a low cost and highly efficient products. In education and training used to driving, flight, ship and tank simulators .

Note : A new, more comprehensive standard called X3D has been developed.

X3D

superceded VRML97. Since VRML97 is a subset of the X3D standard, VRML files can still be processed by newer X3D browsers

Key Concepts

- Virtual environments are realised by a player (browser) that interprets, executes, and presents VRML files
- A VRML file contains at least a header and nodes

- Nodes are arranged in a 'scene graph'
- Nodes have field and field values that describe their attributes
- Shape nodes are used to define shapes with appearance and geometry
- Predefined geometry nodes are Box, Cone, Cylinder and Sphere nodes
- Complex shapes can be built by grouping predefined shapes together and/or using a PointSet, IndexedLineSet, or a IndexedFaceSet, and/or using an Extrusion node.

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

UNIT 5 VR DISPLAYS

Head-mounted displays

The head-mounted display (HMD) was the first device providing its wearer with an immersive experience. Evans and Sutherland demonstrated a head-mounted stereo display already in 1965. It took more than 20 years before VPL Research introduced a commercially available HMD, the famous "EyePhone" system (1989)

An HMD uses small monitors placed in front of each eye which can provide stereo, binocular or monocular images. Stereo images are provided in a similar way to shutter glasses, in that a slightly different image is presented to each eye. The major difference is that the two screens are placed very close (50-70mm) to the eye, although the image, which the wearer focuses on, will be much further away because of the HMD optical system. Binocular images can be provided by displaying identical images on each screen and monocular images by using only one display screen.

The most commonly used displays are small Liquid Crystal Display (LCD) panels but more expensive HMDs use Cathode Ray Tubes (CRT) which increase the resolution of the image. The HMD design may partially or fully exclude the users view of the real world and enhances the field of view of the computer generated world. The advantage of this method is that the user is provided with a 360°; field of regard meaning that the user will receive a visual image if they turn their head to look in ANY direction.

All fully immersive systems will give a sense of presence that cannot be equalled by the other approaches discussed earlier, but the sense of immersion depends of several parameters including the field of view of the HMD, the resolution, the update rate, and contrast and illumination of the display.



A typical HMD houses two miniature display screens and an optical system that channels the images from the screens to the eyes, thereby, presenting a stereo view of a virtual world. A motion tracker continuously measures the position and orientation of the user's head and allows the image generating computer to adjust the scene representation to the current view. As a result, the viewer can look around and walk through the surrounding virtual environment.

To overcome the often uncomfortable intrusiveness of a head-mounted display, alternative concepts (e.g., BOOM and CAVE) for immersive viewing of virtual environments were developed.

Basic features of HMD

- Can display either stereo or mono images depending on type
 1. Stereo images (binocular disparity)
 2. Same image twice (binocular concordance)
 3. Single image (uniocular)

- May be totally immersive or see-through
- May include a built-in head-tracker
- May have built-in stereo headphones

Notes

Some related terms which affect the quality of HMD:

- Resolution: Number of pixels in the display.
- Field of View (FOV):
 1. Angle (in degrees) of the visual field.
 2. Varies from person to person, but typically 180 degrees (horizontal).
 3. 60 to 90 degrees is the usual minimum for a feeling of immersion
- Display technique: Usually LCD, but sometimes CRT
- Size/weight in combination with the factors above

LCD shutter glasses

The CAVE, ImmersaDesks, I-Wall and PARIS virtual reality (VR) systems developed at EVL display active stereo images by use of [StereoGraphics](#) Liquid Crystal Display (LCD) shutter glasses. The computer generates two images, one for the left and one for the right eye; the user wears the wireless glasses, which alternately block and pass images to create the stereo effect. The computer system generates left and right eye images sequentially. An infrared signal synchronizes the glasses to the computer images, so that the right image is

shown when the right lens is transparent and the left image is shown when the left lens is transparent. Infrared emitters are placed throughout and around the VR display environment, so that regardless of where the user 'looks' the glasses are fully functional.

The tracker, attached to the stereo glasses finds the position of the user's head, enabling the CAVE software to calculate and update the images from the user's perspective. In the CAVE, it is also necessary to synchronize in stereo the four projected video displays that make up the three walls and the floor.

The CAVE, ImmersaDesk, I-Wall and PARIS systems use active shutter glasses instead of the passive polarized glasses used in 3D cinemas for a variety of reasons. In 3D cinema, left and right eyes are multiplexed by the plane of polarization with one

polarization angle for the left and one polarization angle for the right eye. These systems are rear-projection displays, which tends to depolarize light as it passes through the screen material. This is especially true at the large viewing angles, 90 degrees and greater, that are common in EVL's VR systems.

Current research includes investigating polarize preserving screen materials for use with passive stereo, and the development of front-projection VR instrumentation suitable for this technology.

Main features of shutter glasses are:



- Display shows left and right images alternately, switching at high speed between images
- Stereoscopic image is seen when the display is viewed with special glasses Typical 'Fishtank VR'
- Particularly good for large audiences in a theatre

BOOM - Binocular Omni-Orientation Monitor

The BOOM (Binocular Omni-Orientation Monitor) from Fakespace is a head-coupled stereoscopic display device. Screens and optical system are housed in a box that is attached to a multi-link arm. The user looks into the box through two holes, sees the virtual world, and can guide the box to any position within the operational volume of the device. Head tracking is accomplished via sensors in the links of the arm that holds the box.

Sem 6 Virtual Reality

61

Notes



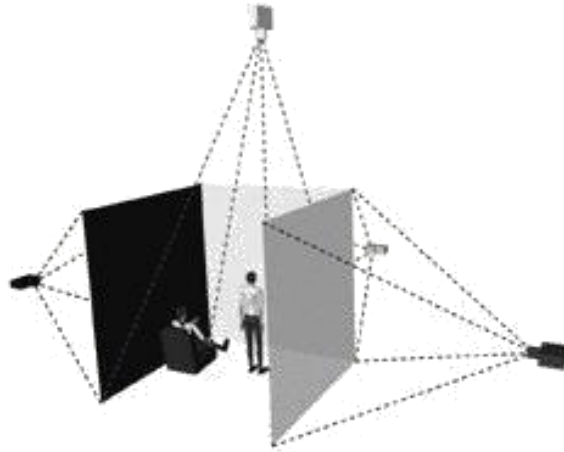
Note : The BOOM device has mostly disappeared from the market. Better, significantly improved HMD devices and projection-based systems like the CAVE have taken over.

- Uses a CRT to provide high-resolution display
- It is comfortable to use, since it does not have to be worn
- Has fast, accurate, built-in tracking

CAVE

The CAVE (Cave Automatic Virtual Environment) was developed at the University of Illinois at Chicago and provides the illusion of immersion by projecting stereo images on

the walls and floor of a room-sized cube. Several persons wearing lightweight stereo glasses can enter and walk freely inside the CAVE. A head tracking system continuously adjust the stereo projection to the current position of the leading viewer.



The CAVE is a surround-screen, surround-sound, projection-based virtual reality (VR) system. The illusion of immersion is created by projecting 3D computer graphics into a 10'x10'x9' cube composed of display screens that completely surround the viewer. It is coupled with head and hand tracking systems to produce the correct stereo perspective and to isolate the position and orientation of a 3D input device. A sound system provides audio feedback. The viewer explores the virtual world by moving around inside the cube and grabbing objects with a three-button, wand-like device.

Unlike users of the video-arcade type of VR system, CAVE dwellers do not wear helmets to experience VR. Instead, they put on lightweight stereo glasses and walk around inside the CAVE as they interact with virtual objects.

Multiple viewers often share virtual experiences and easily carry on discussions inside the CAVE, enabling researchers to exchange discoveries and ideas. One user is the active viewer, controlling the stereo projection reference point, while the rest of the users are passive viewers.

The CAVE was designed from the beginning to be a useful tool for scientific visualization; EVL's goal was to help scientists achieve discoveries faster, while matching the resolution, color and flicker-free qualities of high-end workstations. Most importantly, the CAVE can be coupled to remote data sources, supercomputers and scientific instruments via high-speed networks, a functionality that EVL, the National Center for Supercomputing Applications, and Argonne National Laboratory are jointly implementing.

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)

4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

UNIT SYSTEMS FOR VR

6 TRACKING

Virtual Reality Tracking Systems

The tracking devices are the main components for the VR systems. They interact with the system's processing unit. This relays to the system the orientation of the user's point of view. In systems which let a user to roam around within a physical space, the locality of the person can be detected with the help of trackers, along with his direction and speed.

The various types of systems used for tracking utilized in VR systems. These are as follows:-

- A six degree of freedom can be detected (**6-DOF**)
- Orientation consists of a yaw of an object, **roll and pitch.**
- These are nothing but the position of the objects within the x-y-z coordinates of a space, however, it is also the orientation of the object.

These however emphasizes that when a user wears a HMD then as the user looks up and down; left and right then the view also shifts. Whenever the user's head tilts, the angle of gaze changes. The trackers on the HMD describe to the CPU where you are staring while the right images are sent back to the screen of HMD.

All tracking system consists of a device that is capable of generating a signal and the signal is detected by the sensor. It also controls the unit, which is involved in the process of the signal and sends information to the

CPU. Some systems ask you to add the component of the sensor to the user (or the equipment of the user's). If this takes place, then you have to put the signal emitters at certain levels in the nearby environment. Differences can be easily noticed in some systems; with the emitters being worn by the users and covered by sensors, which are attached to the environment. The signals emitted from emitters to different sensors can take various shapes, including electromagnetic signals, optical signals, mechanical signals and acoustic signals.

Tracking devices are intrinsic components in any VR system. These devices communicate with the system's processing unit, telling it the orientation of a user's point of view. In systems that allow a user to move around within a physical space, trackers detect where the user is, the direction he is moving and his speed.

There are several different kinds of tracking systems used in VR systems, but all of them have a few things in common. They can detect six degrees of freedom (6-DOF) -- these are the object's position within the x, y and z coordinates of a space and the object's orientation. Orientation includes an object's yaw, pitch and roll.

From a user's perspective, this means that when you

wear an HMD, the view shifts as you look up, down, left and right. It also changes if you tilt your head at an angle or move your head forward or backward without changing the angle of your gaze. The trackers on the HMD tell the CPU where you are looking, and the CPU sends the right images to your HMD's screens.

Every tracking system has a device that generates a signal, a sensor that detects the signal and a control unit that processes the signal

and sends information to the CPU. Some systems require you to attach the sensor component to the user (or the user's equipment). In that kind of system, you place the signal emitters at fixed points in the environment. Some systems are the other way around, with the user wearing the emitters while surrounded by sensors attached to the environment.

The signals sent from emitters to sensors can take many forms, including electromagnetic signals, acoustic signals, optical signals and mechanical signals.

Each technology has its own set of advantages and disadvantages.

Main types of Tracking are:

Mechanical tracking systems

This tracking system is dependent on a physical link between a fixed reference point and the target. One of the many examples is that mechanical tracking system located in the VR field, which is indeed a BOOM display. A BOOM display, an HMD, is attached on the rear of a mechanical arm consisting 2 points of articulation. The detection of the orientation and position of the system is done through the arm. The rate of update is quite high with mechanical tracking systems, but the demerit is that they limit range of motion for a user.

Electromagnetic tracking systems

They calculate magnetic fields generated by passing an electric current simultaneously through 3 coiled wires. These wires are set up in a perpendicular manner to one another. These small turns to be an electromagnet. The system's sensor calculate how its magnetic field creates an impact on the other coils. The measurement shows the orientation and direction of the emitter. The responsiveness of an efficient electromagnetic tracking system is really good. They level of latency is quite low. The drawback is that whatever that can create a magnetic field, can come between the signals, which are sent to the sensors.

Optical tracking devices use light to measure a target's position and orientation. The signal emitter in an optical device typically consists of a set of infrared LEDs. The sensors are cameras that can sense the emitted infrared light. The LEDs light up in sequential pulses. The cameras record the pulsed signals and send information to the system's processing unit. The unit can then extrapolate the data to determine the position and orientation of the target. Optical systems have a fast upload rate, meaning latency issues are minimized. The system's disadvantages

are that the line of sight between a camera and an LED can be obscured, interfering with the tracking process. Ambient light or infrared radiation can also make a system less effective.

These devices use light to calculate a target's orientation along with position. The signal emitter typically includes a group of infrared LEDs. The sensors consist of nothing but only cameras. These cameras can understand the infrared light that has been emitted. The LEDs illuminates in a

fashion known as sequential pluses. The pulsed signal are recorded by the camera and then the information is sent to the processing unit of the system . Data can be extrapolated by this unit. This will estimated the position as well as orientation of the target. The upload rate of optical systems is quite fast which has in fact reduced the tenancy issue. The demerits of the systems ae that the line of sight between an LED and camera can be obscured, which interferes with the process of tracing. Infrared radiation or ambient light are also different ways that can make a system useless.

Acoustic

tracking systems emit and sense

ultrasonic sound waves to determine the position and orientation of a target. Most measure the time it takes for the ultrasonic sound to reach a sensor. Usually the sensors are stationary in the environment -- the user wears the ultrasonic emitters. The system calculates the position and orientation of the target based on the time it took for the sound to reach the sensors. Acoustic tracking systems have many disadvantages. Sound travels relatively slowly, so the rate of updates on a target's position is similarly slow. The environment can also adversely affect the system's efficiency because the speed of

sound through air can
change depending on the temperature, humidity or
barometric pressure in
the environment.

Typical Problems

- Can be uncomfortable to use if movement is restricted too much
- Precision is important if user actions are to be translated correctly into
- actions in the virtual environment
- Lag can be disorienting and cause discomfort

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

BCA (DES)306-

3rd

YEAR

VIRTUAL REALITY

[Type here]



**Babasaheb Ambedkar
Open University**

Block 3

BCA(DES)-306

VIRTUAL REALITY

Copyright © 2018 KnowledgeManagementand Research Organization.

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by means of, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

Acknowledgment

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/ amendment in future edition of this book.

The content is developed by taking reference of online and print

Publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken “as is” and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.’



ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a

Successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual

Involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect,

Students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self- instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual- skills objectives may be met by designing instructions that make

use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive

Communication clues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to r e d u c e

This. Divide and to bring this Self Instructional Material as the b e s t

teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self- instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts.

Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline.

These could be in the form of students 'assignments, a research project

or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you
Expect.

All the best for your studies from our team!

All the best for your studies from our team!

UNIT 7 HUMAN FACTORS IN VR

There are a number of factors that should be taken into consideration when designing the user interface of a virtual environment in order to minimise physical discomfort for the user

- Latency causes user discomfort (simulation sickness, nausea)
- Well design and optimised models reduce latency and improve rendering consistency
- There are a number of problems with stereoscopic viewing
- Eye strain problems?
- Long-term physical and psychological effects?

Virtual Reality Input Devices

Datagloves, wands, stairsteppers. These and other interface devices used in virtual environments serve as portals into virtual reality. Whether you are using the head-mounted displays, BOOM, or CAVE, such interface devices form a physical connection between you and the computer through which you interact with the images displayed.

You can pick up objects, navigate a plane through downtown Chicago, or join two molecules of protein. The varieties and applications in virtual reality are limitless.

Datagloves

Data gloves offer a simple means of gesturing commands to the computer. Rather than punching in commands on a keyboard, which can be tricky if you're wearing a head-mounted display or are operating the BOOM, you program the computer to change modes in response to the gestures you make with the datagloves.

Pointing upwards may mean zoom in; pointing down, zoom out. A shake of your fist may signal the computer to end the program. Some people program the computer to mimic their hand movements in the simulation; for instance, to see their hands while conducting a virtual symphony.

One type of dataglove has a web of fiber optic cables along its back. Changes in the amount of light transmitted to the computer by the cables signal how the joints of your fingers are bent. Once the dataglove has been calibrated to your hand, your gestures trigger pre-programmed commands.

Other gloves use strain sensors over the joints to detect movement. Yet others rely on mechanical sensors to measure your hand movements.

Some computer users have elaborated on the dataglove concept by creating facial sensors, even body suits. Not many scientists have climbed into these get ups, but animators have. Already, facial movement sensors hooked to computers are simplifying their job: animating cartoons.

Sem 6
Virtual Reality

72

Notes

Wands

Wands, the simplest of the interface devices, come in all shapes and variations. Most incorporate on-off buttons to control variables in a simulation or in the display of data. Others have knobs, dials, or joy sticks. Their design and manner of response are tailored to the application. For example, biologists sometimes use wands like scalpels to slice tissue samples from virtual brains.

Most wands operate with six degrees of freedom; that is, by pointing a wand at an object, you can change its position and orientation in any of six directions: forward or backward, up or down, or left or right. This versatility coupled with simplicity are the reasons for the wand's popularity.

VirtulSphere

If you don't have a CAVE system or the cash to drop on a DataSuit, there are still a few options available to you to provide users a way of getting around a virtual environment without using a wand or joystick. Researchers believe devices that allow for more natural navigation within a VR environment also increase the user's sense of immersion. With this in mind, engineers and scientists developed a few different systems for user navigation.

One system is the treadmill. A treadmill is useful because the user remains stationary with respect to the real world, but feels as if he is actually walking

through the virtual environment. Researchers have found it relatively simple to link a treadmill to a computer system so that a user's steps result in an appropriate adjustment in the system's graphics. An obvious limitation of normal treadmills is that you can only walk in two directions: backward or forward.

Some companies have developed omni-directional treadmills. These devices allow a user to step in any direction. Normal treadmills use a single motor, which exerts force either forward or backward relative to the user. Omni-directional treadmills use two motors -- from the user's perspective the treadmill can exert force forward, backward, left or right. With both motors working together, the treadmill can allow a user to walk in any direction he chooses on a walking surface wrapped around a complex system of belts and cables.

An alternative to a treadmill is a pressure mat. You may have seen pressure mats

used with video games like "Dance Dance Revolution." There are many kinds of pressure sensors, though the most common are electromechanical pressure sensors. An electromechanical pressure sensor is a relay that activates when pressure is applied to the sensor. When the circuit closes, an electric current runs through it, signaling the CPU to make changes to the graphic output sent to the user

The company VirtuSphere, Inc. offers a unique way for users to move around inside a virtual environment. It looks like a human-size hamster ball -- the user gets inside the sphere and walks around in it. The sphere rests on a stable platform that has several wheels resting against the sphere, allowing it to roll in any

Virtual Reality Sem 6

74

Notes

direction while staying in the same fixed position. Sensors in the wheels tell the CPU which way the user is walking, and the view within the user's HMD changes accordingly.

Within CAVE systems, some VR researchers are experimenting with a technique called passive haptics. "Haptics" refers to the sense of touch, so a haptic system is one that provides the user with physical feedback. A joystick with force-feedback technology is one example of a haptic interface device. Passive haptics are a little different in that they don't actively exert force against a user. Instead, passive haptics are objects that physically represent virtual elements in a VR environment. For instance, a real folding table might double as a virtual kitchen counter.

Having

something real to touch in a virtual environment enhances the user's sense of immersion and helps him navigate through the simulation.

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

UNIT 8 APPLICATIONS OF VR

As the technologies of virtual reality evolve, the applications of VR become literally unlimited. It is assumed that VR will reshape the interface between people and information technology by offering new ways for the communication of information, the visualization of processes, and the creative expression of ideas. Note that a virtual environment can represent any three-dimensional world that is either real or abstract. This includes real systems like buildings, landscapes, underwater shipwrecks, spacecrafts, archaeological excavation sites, human anatomy, sculptures, crime scene reconstructions, solar systems, and so on. Of special interest is the visual and sensual representation of abstract systems like magnetic fields, turbulent flow structures, molecular models, mathematical systems, auditorium acoustics, stock market behavior, population densities, information flows, and any other conceivable system including artistic and creative work of abstract nature. These virtual worlds can be animated, interactive, shared, and can expose behavior and functionality. Useful applications of VR include training in a variety of areas (military, medical, equipment operation, etc.), education, design evaluation (virtual prototyping), architectural walk-through, human factors and ergonomic studies, simulation of assembly sequences and maintenance tasks, assistance for the handicapped, study and treatment of phobias (e.g., fear of height), entertainment, and much more.

VIRTUAL REALITY:

Applications for Grand Challenges

Grand Challenge research is employing high-performance computing and communications to build more energy-efficient cars and airplane, to design better drugs, to improve military surveillance systems and environmental monitoring, to create new materials. The 34 Grand Challenge projects now underway range from explorations of molecules to studies on the origins of galaxies.

Cosmology

One of the largest, 3-dimensional simulations of the universe is helping scientists refine theories about the origins of galaxies. By digitally altering the mix of stellar gas, ordinary matter, and dark matter created soon after the Big Bang, cosmologists are searching for the correct formula for replicating the universe as it exists. Knowing how the structures we see today emerged from the fireball of

creation will reveal much about the future of the cosmos.

Can we really understand the cosmos, in all of its enormity and complexity? Or is cosmology just another example of humankind's hubris? There's no reason, in principle, why we shouldn't be able to uncover the secrets of the universe--its age, how its structures formed, even its fate. After all, "cosmos" is the Greek word for order, and although the first 10^{-43} second following the Big Bang is scientifically out of our reach, everything that happened after that instant followed the known laws of physics. The universe may be extraordinarily complex, but it is comprehensible.

Cosmologists' understanding of the universe is only as good as their observational data. Recent observations from powerful earth-based and space-borne telescopes, including the Hubble Space Telescope have yielded a wealth of data. These observations provide the means by which to evaluate competing cosmological models. All of the current theories are variants of the Big Bang Theory which differ only in the assumed global geometry of the universe (i.e., open, closed, or flat), the

amount and composition of the dark matter, and the spectrum of density fluctuations which give rise to structures.

Cosmologists are combining theory with data to build computer models which simulate the emergence and evolution of large scale structures from the minute density fluctuations in the nascent universe, and the formation of individual galaxies within the superstructures. The benchmark is, of course, all around us: how well do the models resemble the real universe?

Material sciences

Researchers are modeling the more than 400 different hydrogen-nitrogen chemical reactions in an internal combustion engine to design cooler, more efficient car engines. Other researchers are analyzing the properties of compounds in a race to discover the next generation of superconducting material. The winner will revolutionize power transmission and transportation.

Molecular biosciences

VR models of some of life's smallest structures are helping scientists decipher the precise mechanisms through which proteins communicate with each other, for instance, to bind antibody to antigen or signal a cell membrane to dilate. Knowledge of this kind is speeding the development of biological and industrial catalysts as well as therapeutic drugs.

Virtual Reality

Notes

Relativity

By simulating the gravitational ripples that would be generated if two black holes collided, researchers hope to confirm the existence of these elusive objects predicted as a consequence of Einstein's famous General Theory of Relativity. Should the simulated ripples precisely match gravitational waves detected by LIGO, an array of sensing devices that will become operational in 2000, not only could the existence of black holes be confirmed but also Einstein's 80-year-old theory finally will be vindicated.

Weather forecasting

When Hurricane Emily approached the Atlantic coast in 1993, a new hurricane model accurately predicted 48 hours in advance that the hurricane would turn sharply back out to sea off Cape Hatteras without making landfall. Predicting long-term weather patterns is one of the outcomes of the new monitoring and instrumentation tools being developed at part of the HPCC. Atmospheric scientists are also turning to advanced computing tools and virtual environments like the CAVE to calculate the behaviour of more local disturbances, particularly thunderstorms that spawn tornados.

VIRTUAL REALITY:

Applications for National Challenges

Manufacturing and industrial design

Major companies are maintaining their global competitiveness by designing products better and faster with virtual reality. General Motors, for instance, is using a virtual prototype rather than a physical model of one of its new model cars to evaluate the interior design for aesthetics, engineering, safety, and ergonomic features. Caterpillar refined its new backhoe and wheel loaders electronically with virtual reality technology.

Education, training, and life-long learning

Interactive computing and communications technology are ushering in a new era of education in which students and teachers, separated by a distance, are conducting research and performing experiments through high-speed connections that will eventually incorporate VR. High school students in Chicago are modeling weather patterns with the help of professors 100 miles away at the University of Illinois as part of a test project. Their experiences will doubtless shape curriculums for elementary and college students, business training programs, and continuing education.

Sem 6

81

Virtual

Reality

Notes

Environmental monitoring

For a glimpse of the complex circulation dynamics and ecosystems beneath the sea surface, researchers are simulating estuarine systems of the Chesapeake Bay. In the first phase, they are studying the effects of the Susquehanna, Potomac, and James Rivers on the salinity of the Bay.

Medicine

Computer modelling and simulation have become increasingly important in many scientific and technological disciplines owing to the wealth of computational power. Calculation of the behaviour of these computational models is increasingly replacing experiments on real world objects—for example, in the car industry (tests based on simulated crashes) or in the development of nuclear weapons—and is becoming an indispensable tool in the development of new products and procedures. Likewise, the development of techniques for acquiring data (for example, medical imaging) has enabled the easy generation of high resolution copies of real world objects from the computer's memory. The development of imaging technologies, such as magnetic resonance imaging, computed tomography, and ultrasound, has made the acquisition of highly detailed anatomical and partially functional models of three dimensional human anatomy a routine component of daily clinical practice.

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

Copyright © 2018 KnowledgeManagementand Research Organization.

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by means of, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

Acknowledgment

Every attempt has been made to trace the copyright holders of material reproduced in this book.

Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/ amendment in future edition of this book.

The content is developed by taking reference of online and print

Publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken “as is” and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.’



ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a

Successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student’s intellectual

Involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, Students ‘assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self- instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective.

These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual- skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive

Communication clues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce This. Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self- instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts.

Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline.

These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you
Expect.

All the best for your studies from our team!

All the best for your studies from our team!

UNIT 9 VRML PART 1

HISTORY OF VRML

VRML was an idea that was expressed in the First International Conference of the World Wide Web. Mark Pesce, Tony Parisi, Tim Berners-Lee, and David Ragget discussed VRML at this conference. VRML would have to be platform independent to be implemented on the internet. Also, the language had to be able to place objects in 3-D space, as well as include attributes such as shape, color, and size. Since VRML would be used in the internet, all platforms should be able to support it: UNIX workstations to humble desktop PC's. Silicon Graphics introduced the Open Inventor format, and it was widely accepted. Since then, slight changes were made, and VRML 1.0 was introduced in May 1995. A more clarified version, VRML 1.0c, was issued January 1996. Now, VRML 2, called Moving Worlds, is in full swing.

VRML is a standard file format for representing 3-dimensional (3D) interactive vector graphics, designed particularly with the World Wide Web in mind. It has been superseded by X3D.

VRML was initiated because of a need felt by the initiators to have a more human oriented interface for information and interaction. While I don't entirely agree blindly with the more human oriented part, it is interesting from a HCI point of view to investigate this for various applications.

Late in 1993, Mark Pesce, and Tony Parisi initiated workings that started VRML on its way. After a mailing-list was started to discuss the development of VRML, it was not long before a basis for VRML was chosen in the form of SGI's Open Inventor file-format. A subset was taken with some extensions to support networking. Although HTML was already well known and standardised, it was decided not to create VRML as an extension for HTML, but a

completely independent language. This way HTML and VRML could develop on their own and not hamper each other.

VRML 2.0 extends the possibilities of v1.0 by allowing properties of objects to be changed by interactions with the user or by using scripts to animate objects in time. Multi-user support is not directly supported, but it is explicitly mentioned in the specification as possible extension. It also has a more formal tone to it, which indicates that VRML is accepted as a serious extension to the Web.

FORMAT

85

VRML is a text file format where, e.g., vertices and edges for a 3D polygon can be specified along with the surface color, UV

mapped textures, shininess, transparency, and so on. URLs can be associated with graphical components so that a web browser might fetch a webpage or a new VRML file from the Internet when the user clicks on the specific graphical component.

Animations, sounds, lighting, and other aspects of the virtual world can interact with the user or may be triggered by external events such

as timers. A special Script Node allows the addition of program code (e.g., written in Java or JavaScript (ECMAScript)) to a VRML file.

VRML files are commonly called "worlds" and have the *.wrl extension (for example island.wrl). VRML files are in plain text and generally compresses well using gzip which is useful for transferring over the internet more quickly (some gzip compressed files use the *.wrz extension). Many 3D modeling programs can save objects and scenes in VRML format.

Emergence, Popularity, and Rival Technical Upgrade

The term VRML was coined by Dave Raggett in a paper called “Extending WWW to support Platform Independent Virtual Reality” submitted to the First World Wide Web Conference in 1994, and first discussed at the WWW94 VRML BOF established by Tim Berners-Lee, where Mark Pesce presented the Labyrinth demo he developed with Tony Parisi and Peter Kennard. In October 1995, at Internet World, Template Graphics Software (TGS) demonstrated a 3D/VRML plug-in for the beta release of Netscape 2.0 by Netscape Communications.

In 1997, a new version of the format was finalized, as VRML97 (also known as VRML2 or VRML 2.0), and became an ISO standard. VRML97 was used on the Internet on some personal homepages and sites such as "CyberTown", which offered 3D chat using Blaxxun Software. The format was championed by SGI's Cosmo Software; when SGI restructured in 1998 the division was sold to

Platinum Technologies, which was then taken over by Computer Associates, which did not develop or distribute the software. To fill the void a variety of proprietary Web 3D formats emerged over the next few years, including Microsoft Chrome and Adobe Atmosphere, neither of which is supported today. VRML's capabilities remained largely the same while realtime 3D graphics kept improving. The VRML Consortium changed its name to the Web3D Consortium, and began work on the successor to VRML—X3D.

SGI ran a web site at vrml.sgi.com on which was hosted a string of regular short performances of a character called "Floops" who was a VRML character in a VRML world. Floops was a creation of a company called "Protozoa".

H-Anim is a standard for animated Humanoids, which is based around VRML, and later X3D. The initial version 1.0 of the H-Anim standard was scheduled for submission at the end of March 1998.

VRML provoked much interest but has never seen much serious widespread use. One reason for this may have been the lack of available bandwidth. At the time of VRML's popularity, a majority of users, both business and personal, were using slow dial-up internet access. This had the unfortunate side effect of having users wait for extended periods of time only to find a blocky, ill-lit room with distorted text hanging in seemingly random locations.

VRML experimentation was primarily in education and research where an open specification is most valued. It has now been re-engineered as X3D.

The MPEG-4 Interactive Profile (ISO/IEC 14496) was based on VRML (now on X3D), and X3D is largely backward-compatible with it. VRML is also widely used as a file format for interchange of 3D models, particularly from CAD systems.

A free cross-platform runtime implementation of VRML is available in Open VRML. Its libraries can be used to add both VRML and X3D support to applications, and a GTK+ plug-in is available to render VRML/X3D worlds in web browsers.

In a September 17, 1998 Computer gram International essay, "VRML Versus Chromeffects: Microsoft Replies", Rachel Chalmerssharply criticized VRML as having "no integration with HTML", whereas "ChromEffects is based on 56 multimedia XML tags". She did not criticize X3D (though extended from VRML as well as XML), because X3D already had this integration. However, VRML survived Chromeffects, canceled in November 1998.

In the 2000s, many companies like Bit management improved the quality level of virtual effects in VRML to the quality level of DirectX9.0c, but at the expense of using proprietary solutions. All main features like game modeling are already complete. They include multi-pass render with low level setting for Z-buffer, BlendOp, AlphaOp, Stencil, Multi-texture, Shader with HLSL and GLSL support, realtime Render To Texture, Multi Render Target (MRT) and PostProcessing.

87

Many demos shows that VRML already supports lightmap, normalmap, SSAO, CSM and Realtime Environment Reflection along with other virtual effects.

FEATURES OF VRML

- **Interactive 3D**

This means that you can manipulate 3D data sets in an easy and standardised method via the web

- **Standards**

VRML is rapidly becoming the standard method for handling virtual reality across the web. The VRML format is a subset of Silicon Graphic's OpenInventor format.

- VRML is clearly a very sophisticated technique for delivering chemical content over the web. In the ACCVIP team we are giving active consideration to this technique and a number of other techniques that have come into use since our project first started. We have however not included any VRML materials into the ACCVIP modules except for

this brief introduction. There are still some problems about the use of VRML. Helper applications and plugins such as Cosmo Player are not as widely available for all platforms as, for example, RasMol and Chime. They need fairly powerful platforms to run. In addition pages containing a lot of VRML material are slow to load. Widespread use of VRML at this time would mean that our materials are not as widely useable as they are in their current form. Things may be different in the future.

APPLICATION OF VRML IN INDUSTRY

- CAD users are pushing around 100+ MB. My desire is to push VRML into the simulation space where they routinely have databases representing the world or a very detailed region. I think a lack of binary format is a big issue. The combination of XML plus binary formats gives us both answers they need. But they must be 1:1 convertible, that's a key issue.

88

- A big chunk of the VRML applications written have been local, so delivery size has not been that big of an issue. No streaming needed, just deliver the whole thing and run.
- VRML can successfully be used in construction industry applications and give benefits including:
 - Easily model structures to be used in a web-based user interface.
 - Web-based user interface is easily accessed by all project participants
 - VRML can represent steel structures in an object-like fashion.

- Object-like VRML representations are easier to update or to extend. VRML is used in Games, Entertainment, and Education

Cyberspace

As with any new technology, the possible uses of its application are not fully explored or invented yet. One example of a possible use is discussed in “Cyberspace” , by the Labyrinth group who view VRML as a way to create a virtual space in which users can navigate. In this vision, the environment is viewed as the medium to find and navigate information. It is my belief that this is only useful in very few applications, because if this was the “normal” interface, it would require a whole new “culture” to develop in order to find the information. In a 3D environment you need cues to know when to make a right turn towards the information you're looking for, which should be 3D-objects that are metaphorical of the information. The problem of metaphors is that they are usually culturally dependent (“Real” culture, that is!), which becomes a problem in a Global interface where people of all kinds of different cultures interact with each other.

This interface would seem most useful as an environment for Computer Supported Cooperative Work (CSCW), because people convey so much information through body language that would probably not get through a video conferencing system. Assuming people already know each other and are from the same culture, the chances of misunderstandings and cultural differences would be much lower. To fully use the interface for this purpose, the "level 1" interface must support a "full body" hardware interface (or something close to that or with similar functionality).

Previews of 3D situations

• Seats in a baseball Stadium

One example of a use for VRML, I came across on the Web was that of a ticket-sales web page for a baseball stadium. The user wanting to buy a ticket would enter a few variables (like price-range of the tickets, date of the event and how many seats are needed) in a form and after submitting that information, receive a VRML-world of the stadium, with viewpoints (predefined camera positions on the world) onto the field to see which seats he would prefer. At every camera location would be an object to order the tickets for those seats or go to the next viewpoint.

• Experiencing a set of directions for a real situation

One example of this use is BigBook3D (VRML yellow-pages for San Francisco, but I couldn't really get it to work properly (because of my slow computer), so I won't discuss it in any detail.

The user searches for a particular business using a normal HTML form, and the system positions the viewpoint of the VRML close the real location (on the 3D map). It is also possible to just "look around in the city".

Another idea, based on this use of VRML and a Dutch web-site for the hospitals, which uses (although not with VRML) the concept of walking through the virtual hospital. For a specific hospital you could ask directions for the room you need to be and be guided through the virtual hospital by directional arrows, movement of the viewpoint and receive hints about the directional markers available in the hospital. This would put patients significantly at ease, when they are unfamiliar with the hospital and save it saves the assistants from having to give directions to everyone entering the building. (I imagine this interface to be available both via Internet and on a computer console at the entrance)

• Dynamically generated worlds to represent design problems Dome-generator.

This is the only one I've seen so far that could be seen as useful in this context. It creates Stratodesic domes from a specification entered in a HTML-form. The only right way to view these domes correctly is in some kind of 3D mode, VRML is almost perfect for the job. Someone interested in creating a Stratodesic dome can just enter some values, see if that was what he/she meant and go back and try again if it wasn't.

90

VRML as interface to information about 3D objects

- **A technical manual of a motorcycle**

I have not found an example of this on the web, but I can image this might be useful. You could start with a complete motorcycle in a VRML-world and when you click on a part, you get (in different “frames”) a web-page describing the part and information about its maintenance and fault-detection and another VRML-world with just the part so one can manoeuvre in and around it.

- **VRML Weather Center**

On this web-page you see the world with satellites around it, each projecting a cone of “vision” to the Earth, representing its current view on the world. When you click on the cone you get the latest image from the satellite that “projects” it. This makes it easy to select which is the image you want to see and all the information (from weather satellites) is navigable from one source.

Games and other Entertainment

- **(Multiplayer) games like Doom**
- **Virtual chat-rooms**

In this context the idea of an Avatar (a virtual representation of the user in the Virtual Environment) in a VRML world is a good example of an application of VRML for human communication. In some environments it can be useful or even necessary to assume a shape or figure that makes one completely unidentifiable. (This could be used for anonymous counseling or just entertainment).

The following quote comes from the VRML hypermail forum

Aaron McMahon makes some interesting remarks about the potential for modeling your own avatar. (Omitted for brevity.) Okay, here's how I imagine mechanisms to work Sometime In The Future(tm). My apologies if this is already painfully evident - call me a newbie, if it helps soothe furrowed brows :).

Your avatar enters a room. As it serves a VRML room description at you, the server queries your browser for a VRML avatar. The avatar description contains - apart from the visual information - a number of "key points" analogous to those neat black-and-white roundels you see on crash dummies (but not visible), which represent the intersection of discrete elements of the avatar structure (arms, legs, eyes, head, torso, whatever). The elements are encapsulated and the relationship between the key points and the elements is described. The avatar movement can

91

now be described purely in terms of the key points. The description can be as basic or complex as it needs to be. The server and client will need to negotiate a suitable scaling.

Your browser also receives the avatar descriptions of other people in the area (with key point info &c.). As you move within the room, your browser passes telemetry information on your key points to the server, which rebroadcasts them (alternative: the server has identified the other visitors, and you send telemetry directly to all interested parties). You receive telemetry for the other visitors, and your browser renders their avatars according to their key point descriptions.

The telemetry is obtained either by interpolation of movement (simulating walking) or from VR hardware directly operated (data gloves and all the other clunky stuff).

(Drawing from Mark's interpolation of a scene from Snow Crash, an idea of how this all could work:-)

A and B are avatars. A's user wants to give a file to B's user. A approaches B. [A's user tags a file for transfer, identifies B as sole permissible target.] A picks a token from a "utility belt" and holds it outstretched to B. B reaches out and takes the token, which changes colour to indicate the start of transfer, [FTP transfer starts] and again to signal completion [the transfer is now complete] (or failure). B puts the token in its utility belt [places the file in an "in-tray" for later filing].

Especially this last example gives an insight into the tendency to go to far in inventing uses for VRML. Of course this could be a way to transfer a file, but it could be done quicker and more intuitively by using a direct text or windows based interface, or at least a method without the hassle of extending an arm and giving/taking it. (Unless the interface with the user is such that the action of extending an arm to give something is at least as intuitive as doing just that in the real world (e.g. with a dataglove)).

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)
4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

UNIT 10 VRML PART 2

VRML stands for Virtual Reality Modeling Language. It was designed to create a more "friendly" environment for the World Wide Web. VRML incorporates 3-D shapes, colors, textures, and sounds to produce a "virtual world" that a user could walk and fly through. VRML is an interpreted language.

LANGUAGE SPECIFICATION

- VRML defines a set of objects useful for doing 3D graphics. These objects are called Nodes.
- Nodes are arranged in hierarchical structures called **scene graphs**. Scene graphs are a collection of nodes; the scene graph defines an ordering for the nodes. The scene graph has a notion of state -- nodes earlier in the world can affect nodes that appear later in the world.
- A node has the following characteristics:
 - **What kind of object it is.** A node might be a cube, a sphere, a texture map, a transformation, etc.
 - **The parameters that distinguish this node from other nodes of the same type.** For example, each Sphere node might have a different radius, and different texture maps nodes will certainly contain different images to use as the texture maps. These parameters are called Fields. A node can have 0 or more fields.
 - **A name to identify this node.** Being able to name nodes and refer to them elsewhere is very powerful; it allows a world's author to give hints to applications using the world about what is in the world, and creates possibilities for very powerful scripting

extensions. Nodes do not have to be named, but if they are named, they can have only one name.

- **Child nodes.** Object hierarchy is implemented by allowing some types of nodes to contain other nodes. Parent nodes traverse their children in order during rendering. Nodes that may have children are referred to as **group nodes**. Group nodes can have zero or more children.
- Blanks, tabs, newlines and carriage returns are whitespace characters wherever they appear outside of string fields. One or more whitespace characters separate the syntactical entities in VRML files, where necessary.
- After the required header, a VRML file contains exactly one VRML node. That node may of course be a group node, containing any number of other nodes.
- VRML is case-sensitive.

94

- Node names must not begin with a digit, and must not contain spaces or control characters, single or double quote characters, backslashes, curly braces, the sharp (#) character, the plus (+) character or the period character.
- Field names start with lower case letters, Node types start with upper case. The remainder of the characters may be any printable ASCII (21H-7EH) except curly braces {}, square brackets [], single ' or double " quotes, sharp #, backslash \\ plus +, period. Or ampersand &.

COORDINATE SYSTEM

- VRML uses a Cartesian, right-handed, 3-dimensional coordinate system. By default, objects are projected onto a 2-dimensional device by projecting them in the direction of the positive Z-axis, with the positive X-axis to the right and the positive Y-axis up.
- The standard unit for lengths and distances specified is meters. The standard unit for angles is radians.

- VRML worlds may contain an arbitrary number of local coordinate systems, defined by modeling transformations using Translate, Rotate, Scale, Transform, and MatrixTransform nodes. Given a vertex V and a series of transformations such as:

Translation { translation T } Rotation { rotation R } Scale { scaleFactor S }

Coordinate3 { point V } PointSet { numPoints 1 }

- The vertex is transformed into world-space to get v' by applying the transformations in the following order:

$V' = T \cdot R \cdot S \cdot V$ (if you think of vertices as column vectors) OR

$V' = V \cdot S \cdot R \cdot T$ (if you think of vertices as row vectors)

- VRML also has a "world" coordinate system as well as a viewing or "Camera" coordinate system. The various local coordinate transformations map objects into the world coordinate system. This is where the scene is assembled. The scene is then viewed through a camera, introducing another conceptual coordinate system. Nothing in VRML is specified using these coordinates. Having a clear model of the object, world and camera spaces will help authors.

FIELDS

- There are two general classes of fields; fields that contain a single value and fields that contain multiple values. Single-valued fields all have names that begin with "SF", multiple-valued fields have names that begin with "MF".

- Multiple-valued fields are written as a series of values separated by commas, all enclosed in square brackets. If the field has zero values then only the square brackets ("[]") are written. A comma may optionally follow the last. If the field has exactly one value, the brackets may be omitted and just the value written.

1

[1,] [1]

NODES

- VRML defines several different classes of nodes. Most of the nodes can be classified into one of three categories; shape, property or group. Shape nodes define the geometry in the world. Property nodes affect the way shapes are drawn. And grouping nodes gather other nodes together, allowing collections of nodes to be treated as a single object.
- Nodes may contain zero or more fields. Each node type defines the type, name and default value for each of its fields. The default value for the field is used if a value for the field is not specified in the VRML file. The order in which the fields of a node are read is not important.
- Here are the 36 nodes grouped by type. The first group is the shape nodes. These specify geometry:
 - AsciiText, Cone, Cube, Cylinder, IndexedFaceSet, IndexedLineSet, PointSet, Sphere
- The second group are the properties. These can be further grouped into properties of the geometry and its appearance and matrix or transform properties:
 - Coordinate3, FontStyle, Info, Material, MaterialBinding, Normal, NormalBinding, Texture2, Texture2Transform, TextureCoordinate2, ShapeHints
 - MatrixTransform, Rotation, Scale, Transform, Translation

These are the group nodes:

- Separator, Switch, WWWAnchor, LOD

The following nodes do not fit into any category.

96

- OrthographicCamera, PerspectiveCamera
- DirectionalLight, PointLight, SpotLight
- WWWInline

CONE

- This node represents a simple cone whose central axis is aligned with the y-axis. By default, the cone is centered at (0,0,0) and has a size of 1 in all three directions. The cone has a radius of 1 at the bottom and a height of 2, with its apex at 1 and its bottom at -1. The cone has three parts: the sides and the bottom.
- The cone is drawn with the current texture and material.
- When a texture is applied to a cone, it is applied differently to the sides and bottom. On the sides, the texture wraps counterclockwise (from above) starting at the back of the cone. The texture has a vertical line at the back, intersecting the yz-plane. For the bottom, a circle is cut from the texture square and applied to the cone's base circle. The texture appears right side up when the top of the cone is rotated towards the y-axis.

PARTS

SIDES	The conical part
BOTTOM	
M	The bottom circular face
ALL	All parts

```

Cone {
  field      SFFloat bottomRadius 1
  field      SFFloat height 2
  field      SFBool side TRUE
  field      SFBool bottom TRUE
}

```

SHAPES AND GEOMETRY

- The Cone specifies only the geometry(shape) of an object not the appearance(color).

VRML also has an Appearance node. Once you have geometry and appearance you have enough to define a VRML object.

```

Shape {
  exposedField SFNode appearance NULL exposedField SFNode geometry NULL
}

```

- Here exposedField is like field. This field is of type SFNode that is a data type that holds another node. It is seen in lot of places in VRML but here it just gives someplace to put your geometry and appearance. The default value is NULL for both these fields which is a special value meaning this

97

value contains nothing. A NULL appearance makes whatever geometry is in the Shape pure wire so it shows up.

```

Shape{
  geometry Cone{ }
}

```

MATERIAL

- This node defines the current surface material properties for all subsequent shapes. Material sets several components of the current material during traversal. Different

shapes interpret materials with multiple values differently. To bind materials to shapes, you use a `MaterialBinding` node.

- The fields in the `Material` node determine the way light reflects off of an object to create color:
 - The `ambientColor` reflects ambient light evenly from all parts of an object regardless of viewing and lighting angles.
 - The `diffuseColor` reflects all VRML light sources depending on the angle of the surface with respect to the light source. The more directly the surface faces the light, the more diffuse light reflects.
 - The `specularColor` and `shininess` determine the specular highlights, e.g., the shiny spots on an apple. When the angle from the light to the surface is close to the angle from the surface to the viewer, the `specularColor` is added to the diffuse and ambient color calculations. Lower `shininess` values produce soft glows, while higher values result in sharper, smaller highlights.
 - Emissive color models "glowing" objects. This can be useful for displaying radiosity-based models (where the light energy of the room is computed explicitly), or for displaying scientific data.
 - Transparency is how "clear" the object is, with 1.0 being completely transparent, and 0.0 completely opaque.
- The lighting parameters defined by the `Material` node are the same parameters defined by the OpenGL lighting model. Several of the OpenGL parameters (such as light attenuation factors) are also left unspecified in VRML. Also note that OpenGL specifies the specular exponent as a non-normalized 0-128 value, which is specified as a normalized 0-1 value in VRML (simply multiplying the VRML "shininess" value by 128 to translate to the OpenGL specular exponent parameter).
- You assume that there is an implicit white ambient light of intensity 0.2 in any VRML scene.

- For rendering systems that do not support the full OpenGL lighting model, the following simpler lighting model is recommended:
- A transparency value of 0 is completely opaque; a value of 1 is completely transparent. Applications need not support partial transparency, but should support at least fully transparent and fully opaque surfaces, treating transparency values ≥ 0.5 as fully transparent.
- Specifying only emissiveColors and no diffuse, specular, emissive, or ambient colors is the way to specify pre-computed lighting. It is expected that browsers will be able to recognize this as a special case and optimize their computations.

Material {

ambientColor [] diffuseColor [] specularColor [] emissiveColor [0.1 0.1 0.2, 0.5 0.8 0.8]
}

- Rendering systems which do not support specular color may nevertheless support a specular intensity. This should be derived by taking the dot product of the specified RGB specular value with the vector [.32 .57 .11]. This adjusts the color value to compensate for the variable sensitivity of the eye to colors.
- If a system supports ambient intensity but not color, the same thing should be done with the ambient color values to generate the ambient intensity. If a rendering system does not support per-object ambient values, it should set the ambient value for the entire scene at the average ambient value of all objects.
- It is also expected that simpler rendering systems will be unable to support both lit and unlit objects in the same world. Many VRML implementations will support only either multiple diffuse colors with a single value for all other fields, or multiple emissive colors with one transparency value and NO (empty, '[]') values for all other fields. More complicated uses of the Material node should be avoided.

Appearance {

exposedField SFNode **material** NULL

exposedField SFNode **texture** NULL

exposedField SFNode **textureTransform** NULL

}

Material {

exposedField SFFloat **ambientIntensity 0.2**

exposedField SFCOLOR **diffuseColor 0.8 0.8 0.8**

exposedField SFCOLOR **emissiveColor 0 0 0**

99

exposedField SFFloat **shininess 0.2**

exposedField SFCOLOR **specularColor 0 0 0**

exposedField SFFloat **transparency 0**

}

Example:

Shape{

appearance Appearance{

material Material{

diffuseColor 1 0 0

}

}

geometry Cone{ }

}

VRML FIELD TYPES

Type	Description
SFBool	The Boolean value TRUE or FALSE

SFFloat	A 32 bit floating point value
SFInt32	A 32 bit signed integer
SFTime	An absolute or relative time value
SFVec2f	A pair of floating point values usually denoted as u, v because they are most often used to represent texture coordinates
SFVec3f	Three floating-point values usually denoted as x, y, z because they are used most often

100

	to represent a 3D position.
SFColor	Three floating point values each between zero and one; representing the red, green, and blue components

	<p>and blue components of a color.</p>
SFRotation	<p>Four floating-point values. The first three values represent an axis with 0,0,0 being the other point on the axis line and the fourth value represents the angle of rotation in radians around the axis.</p>
SFImage	<p>A 2D image with one to four color components allowing representation of monochrome to full-color images with transparency.</p>
SFString	<p>A UTF8 (international character) string</p>
SFNode	<p>A container for a VRML node</p>

MFFloat	An array of SFFloat values
MFInt32	An array of SFInt32 values
MFVec2f	An array of SFVec2f values
MFVec3f	An array of SFVec3f values

101

	SFVec3f values
MFColor	An array of SFColor values
MFRotation	An array of SFRotation values
MFString SFString values	An array of SFString values

Example:

```
#VRML V2.0 utf8
```

```
Shape {
  geometry Cone { }
}
```

Example:

```
#VRML V2.0 utf8
```

```
Shape {  
  appearance Appearance {  
    material Material {  
      diffuseColor 1 0 0  
    }  
  }  
  geometry Cone { }  
}
```

Example

```
#VRML V2.0 utf8.0  
Shape{  
  appearance Appearance {  
    material Material {  
      diffuseColor 0.606383 0.432091 0  
    }  
  }  
  
  geometry Cone { height 3 }  
  
  102  
}
```

```
Shape{  
  appearance Appearance{  
    material Material{  
      diffuseColor 0 0.49931 1
```



```

    exposedField MFNode    children      []
    exposedField SFRotation rotation    0 0 1 0
    exposedField SFVec3f    scale        1 1 1
    exposedField SFRotation scaleOrientation 0 0 1 0
    exposedField SFVec3f    translation  0 0 0
    field          SFVec3f    bboxCenter  0 0 0

```

103

```

field          SFVec3f    bboxSize      -1 -1 -1
}

```

- The Transform node has a children field that can hold a Shape node. This node is different because it is an MFNode rather than an SFNode field. That means the children field can have more than one node in a list. If more than one node is placed in the children field the node must be enclosed in a [] bracket.
- The other fields of Transform change the position, size and orientation of the nodes in the children field. The fields controlling these transformations are center, rotation, scale, scaleOrientation and translation.
- The translation field allows you to move the children of the Transform away from their default location of 0 0 0.

Example: A TRANSFORMED CONE

```
#VRML V2.0 utf8
```

```

Transform {
translation 4 5 0
children Shape {

```

```
appearance Appearance {
material Material {
diffuseColor 1 0 0
}
}
geometry Cone { }
}
}
```

Example: VRML SCENE WITH RED AND GREEN CONE

```
#VRML V2.0 utf8
```

```
Transform {
translation 4 5 0
children Shape {
appearance Appearance {
material Material {
diffuseColor 1 0 0
```

```
104
}
}
geometry Cone { }
}
}
```

```
Shape {
appearance Appearance {
material Material {
diffuseColor 0 1 0
}
}
```

```
}  
geometry Cone { }  
}
```

Example: VRML SCENE WITH A RED, GREEN AND BLUE CONE

```
#VRML V2.0 utf8
```

```
Transform {  
translation -2 0 -6
```

```
children [  
Transform {  
translation 4 5 0
```

```
children Shape {  
appearance Appearance {  
material Material {  
diffuseColor 1 0 0  
}  
}  
geometry Cone { }  
}  
}
```

```
Shape {  
appearance Appearance {  
material Material {  
diffuseColor 0 1 0  
}  
}
```



```
geometry Cone { }  
}
```

```
105  
]  
}
```

```
Shape {  
  appearance Appearance {  
    material Material {  
      diffuseColor 0 0 1  
    }  
  }  
  geometry Cone { }  
}
```

SCALE AND ROTATION

- **SCALE:** The Transform node can also alter the size of its children and their orientation. The size is changed with the scale field, which is an SFVec3f just like the translation field. But in this case the three values independently change the size of the children relative to the three axes. The default 1 1 1 makes the children their normal size.

Example:

```
#VRML V2.0 utf8  
Transform {  
  scale 3 1 1  
  children Shape {  
    appearance Appearance {  
      material Material {  
        diffuseColor 0 1 0
```

```

}
}
geometry Cone { }
}
}

```

- **Rotation:** The orientation of children of a Transform is changes with the oration field, which is an SFRotation field type.

- Take a soda can and stick a pencil through the middle of the can horizontally. Now spin the pencil and the soda can will spin around the center of the pencil. Take the pencil out and try to push it through again but this time at a 45 degrees angel. Now when you join the pencil the can does something interesting. If you keep the pencil at the same 45 degrees angle you started with the

106
can starts upright, spins onto its side, goes back to begin upright again and so on.

- The pencil forms the axis of rotation for the soda can. Whatever direction the pencil goes through the can, that is the axis about which the can will rotate. In fact the pencil does not even heave to go through the can. If it does not the can will orbit around the pencil as if attached to it with a string. This is how the SFRotation value is specified with four numbers. The first three define the axis of rotation and the fourth defines the angle the object should rotate relative to its initial position. Here is an example:

```
rotation 0 1 0 1.57
```

- Here the object will rotate counterclockwise when looking from the top and 90 degree about a vertical axis. To find the actual axis you draw a line form 0 0 0 to the axis value in the SFRotation. In this case the line goes form 0 0 to 0 1 0 which goes along the Y-axis.

The angle is in radians. There are 6.28 radians in a circle. 3.14 is 180 degree, 1.57 is 90 degree and 0.78 is 45 degree.

RIGHT HAND RULE:

- Rotations in VRML follow the right hand rule. The axis in this example goes from 0 0 0 to 0 1 0. That is the positive Y direction. Every axis has a direction. Now if you grasp this axis line with you're right hand and your thumb is pointing in the direction of the axis and if you look at the axis with your thumb pointing toward you your fingers curve around the axis in the counterclockwise direction. This is the direction an SFRotation will rotate an object.

Example:

```
#VRML V2.0 utf8
Transform {
rotation 0 0 1 0.78
children Shape {
appearance Appearance {
material Material {
diffuseColor 1 0 0
}
}
geometry Cone { }
}

107
}
```

- **center:** The center field sets the point about which the rotation occurs. As already mentioned the axis of rotation is the line that goes through the origin, 0 0 0 and the axis part of the SFRotation. Changing the center field from its default of 0 0 0 will cause the

rotation axis to still have the direction of the original axis but it will go through the center instead of the origin. This is often useful if you want to rotate an object along its edge such as the base of the cone.

Example:

```
#VRML V2.0 utf8
```

```
Viewpoint {  
  position 0 -4 10  
}
```

```
Transform {  
  translation 0 -2 0  
  scale 3 1 1  
  scaleOrientation 0 1 0 -0.78  
  rotation 0 0 1 0.78  
  center 0 1 0
```

```
  children Shape {  
    appearance Appearance {  
      material Material {  
        diffuseColor 1 0 0  
      }  
    }  
    geometry Cone { }  
  }
```

```
Shape {  
  appearance Appearance {
```

```

material Material {
diffuseColor 0 1 0
}
}
geometry Cone { }
}

```

108

COMPOSITE HIERARCHICAL TRANSFORMATIONS

- Composition is the act of putting smaller pieces together to make a larger whole. The human body provides an excellent example of composition. When you move your forearm it hinges at the elbow and everything from your forearm to your fingertips moves together. When you move your wrist your hand and fingers move and so on all the way down to the last knuckle on your fingers. You can think of your fingers as children of your hand, which is a child of your arm, which is a child of your torso. Your body is a big scene graph.

GROUPS AND TRANSFORMS

The simplest grouping node is Group

```

Group{
eventIn          MFNode          addChildren
eventIn          MFNode          removeChildren
exposedField     MFNode          children      []
    field        SFVec3f         bboxCenter   0 0 0
    field        SFVec3f         bboxSize    -1 -1 -1
}

```

- The Group node has two other fields `bboxCenter` and `bboxSize`. These fields appear in the Transform node as well. They allow you to give a hint to the VRML browser about the total size and location of the children of this node. The `bboxSize` field is set to the size of a box surrounding the children in all three dimensions and the `bboxCenter` specifies the center of this box. If this box is not specified for set to its default size of `-1 -1 -1`, the VRML browser must compute the value itself.

BASIC GEOMETRIC PRIMITIVES

Box{

```
field SFVec3f          size 2 2 2
}
```

Sphere{

```
field SFFloat         radius      1
}
```

109

Cylinder{

```
field SFBool         bottom      TRUE
field SFFloat        height      2
field SFFloat        radius      1
field SFBool         side        TRUE
field SFBool         top         TRUE
}
```

Example:

```
#VRML V2.0 utf8
```

```
Viewpoint {
```

```
position 0 -3 4
```

```

orientation 1 0 0 0.6
}
Shape {
appearance Appearance {
material Material {
diffuseColor 1 0 0
}
}
geometry Cylinder {
bottom FALSE
top FALSE
}
}

```

```

Shape {
appearance Appearance {
material Material {
diffuseColor 0 0 1
}
}
geometry Cylinder {
side FALSE
}
}

```

USING DEF/USE TO REUSE PARTS OF THE SCENE

- If you want the world made up of 16 blue spheres arranged in a square pattern you would need 16 sphere shapes and 16 Transform nodes to position them. But VRML gives you tools to avoid this duplication of work DEF and USE. These two modifiers can be

placed wherever a node name is allowed. The DEF keyword is followed by a name and then a node description. This node now has a name defined for it. Legal names can be as short as one character long, but must start with a letter. Following the first letter may be any letter, number or any of the characters given below

SPECIAL CHARACTERS ALLOWED IN A VRML DEF NAME:

Symbol	Name
!	Exclamation point
\$	Dollar
%	Percent
&	Ampersand
(Left parenthesis
)	Right parenthesis
*	Asterisk
/	Forward slash
:	Colon
;	Semicolon
<	Less than
=	Equal
>	Greater than
?	Question mark
@	Commercial at
^	Hat
_	Underscore
`	Accent grave
	Vertical bar
~	Tilde

Example:

```
Shape{  
  appearance DEF Red Appearance{  
    material Material { diffuseColor 1 0 0 }  
  }  
  geometry Box{ }  
}
```

```
  geometry Box{ }
```

```
}
```

```
Shape{  
  appearance USE Red  
  geometry Cylinder { }  
}
```

111

Example:

```
#VRML V2.0 utf8
```

```
DEF RowOfSpheres Group {
```

```
  children [
```

```
    Transform {
```

```
      translation -6 -6 0
```

```
    children Shape {
```

```
      appearance Appearance {
```

```
        material Material {
```

```
          diffuseColor 0 0 1
```

```
        }
```

```
      }
```

```
    geometry Sphere { }
```

}

}

Transform {

translation -2 -6 0

children Shape {

appearance Appearance {

material Material {

diffuseColor 0 0 1

}

}

geometry Sphere { }

}

}

Transform {

translation 2 -6 0

children Shape {

appearance Appearance {

material Material {

diffuseColor 0 0 1

}

}

geometry Sphere { }

}

}

Transform {

translation 6 -6 0

```
children Shape {  
  appearance Appearance {  
    material Material {  
      diffuseColor 0 0 1  
    }  
  }  
  geometry Sphere { }  
}  
]  
}
```

```
Transform {  
  translation 0 4 0  
  children USE RowOfSpheres  
}
```

```
Transform {  
  translation 0 8 0  
  children USE RowOfSpheres  
}
```

```
Transform {  
  translation 0 12 0  
  children USE RowOfSpheres  
}
```

OBJECT APPEARANCE

- The appearance node of VRML is used to apply color to a geometric model. This includes 2D images applied to the object ways to adjust the shininess and brightness of an object and even ways to apply a different color at each vertex of the object.

Textures

- You can add color by wrapping a 2D image called a texture around the surface of an object. The image follows the contours of the object as needed. In VRML any JPEG image can be used as a texture. Textures are added using the ImageTexture node.

113

```
ImageTexture{
  eExposedField          MFString  url          []
  field                  SFBool    repeatS      TRUE
  field                  SFBool    repeatT      TRUE
}
```

Example:

```
#VRML V2.0 utf8

Transform {
  translation -2 0 0

  children Shape {
    appearance DEF A Appearance {
      material Material { }
      texture ImageTexture {
        url "WildLife.gif"
```

```

}
}
geometry Box { }
}
}

```

```

Transform {
translation 2 0 0

```

```

children Shape {
appearance USE A
geometry Sphere { }
}
}

```

DEFAULT TEXTURE MAPPING FOR ALL GEOMETRIC SHAPES

Shape	Mapping
Box	One copy of the texture on each field
Cone	Wrap once around the base with the seam at the back. Squeeze all points at the top. Image is cut out at the lower-left corner and placed on the bottom cap, appearing right side up when bottom is tilted toward you.
Cylinder	Wrap once around horizontal diameter with the seam at the back. Image is cut out at the lower-left corner and

114

	placed on the top and bottom caps appearing right side up when the bottom or top is tilted toward you.
ElevationGrid	One copy of the texture is laid on the grid

Extrusion	Wrap once around cross-section diameter, horizontally. Place once along pine vertically.
IndexedFaceSet	Mapping is determined by measuring the dimensions along the major axes. The longest dimension is used to determine the horizontal mapping. The second longest dimension determines the vertical mapping.
IndexedLineSet	No textures are applied to this shape
PointSet	No textures are applied to this shape
Sphere	Wrap once around the horizontal diameter with the seam at the back. Squeeze image at the top and bottom of the sphere to a single point at the top and bottom
Text	Image is cut out to fit a single copy of the texture over the entire block of text.

TRANSFORMING TEXTURES

- There are two ways to change the default way textures are applied to objects. The first uses the TextureTransform node.

```

TextureTransform{
    exposedField      SFVec2f      center      0
    exposedField      SFFloat      rotation     0
    exposedField      SFVec2f      scale         1
    exposedField      SFVec2f      translation   0
}

```

- The TextureTransform node is placed in the textureTransform field of the Appearance node. It scales, rotates, translates the texture on the object. Setting the scale field to 2 2 will scale the texture s and t coordinates of the texture by 2. Rather than making the texture larger on the object it makes it smaller. This is because rather than scaling the texture image up and then applying the larger texture to the object it scales the s and t coordinates used to map the texture to the object so the texture has to shrink down to fit on an object of the same size. The same thing happens with rotation and translation. A translation field of 0.5 0.5 will place a point in the center of the

115

texture at the lower-left corner of the object. The appearance is that the texture has shifted by a value of $-0.5 -0.5$.

Example:

```
#VRML V2.0 utf8
```

```
Transform {  
translation -6 0 0
```

```

children Shape {
  appearance Appearance {
    material DEF M Material { }
    texture DEF T ImageTexture { url "fish.jpg" }
  }
  geometry DEF IFS IndexedFaceSet {
    coord Coordinate {
      point [ -1 -1 0, 1 -1 0, 1 1 0, -1 1 0 ]
    }
    coordIndex [ 0, 1, 2, 3 ]
  }
}

```

```

Transform {
  translation -2 0 0
  children Shape {
    appearance Appearance {
      material USE M
      texture USE T
      textureTransform TextureTransform { scale 2 2 }
    }
    geometry USE IFS
  }
}

```

```

116
}

```

```

Transform {
  translation 2 0 0
  children Shape {
    appearance Appearance {

```



```

material USE M
texture USE T
textureTransform TextureTransform { translation 0.5 0.5 }
}
geometry USE IFS
}
}

```

```

Transform {
translation 6 0 0
children Shape {
appearance Appearance {
material USE M
texture USE T
textureTransform TextureTransform { rotation 0.78 }
}
geometry USE IFS
}
}

```

CHANGING TEXT FONT STYLE

- The text node displays a common looking font with a height of 1 by default. But the FontStyle node can be used to change this and much more.

```

FontStyle{
field      SFString      family      "SERIF"
field      SFBool        horizontal   TRUE
field      MFString     justify     "BEGIN"
field      SFString     language    ""
field      SFBool        leftToRight TRUE

```

```

        field      SFFloat      size      1.0
        field      SFFloat      spacing   1.0
        field      SFString     style    “PLAIN”
        field      SFBool      topToBottom  TRUE
    }

```

117

- The FontStyle node is placed in the fontStyle field of the Text node. The family and style fields allow you to change the look of the font. They both take SFString values in which you can place one of several strings that select some general appearance characteristics of the font. The family field can have the string SERIF for a serif font, SANS for a sans-serif font or TYPEWRITER for a fixed pitch font. A sans-serif font is simpler and many systems use a font called Helvetica for it. A fixed pitch font is one where all characters are the same width. The style field modifies the font family's appearance. The allowed values here are PLAIN for a font that is both heavy slanted. These apply to the family attribute to create 12 possible font styles.
- The size field allows you to change the height of the font. By default the size is 1, which means the font should be tall enough so that lines of text can be spaced vertically at 1 unit spacing. This does not set an absolute height for every character but every font has a notion of how tall it must be to look acceptable when lines are spaced at this distance. This line spacing can be adjusted from this default with the spacing field. This value is multiplied by the size field to determine how much distance there will be from the top of one line of text to the next. Here is a simple example of these four fields in action.

Example:

```

#VRML V2.0 utf8
Transform {
translation -16 0 0

```

```
children Shape{
  geometry Text{
    string [ "VRML", "is the wave", "of the future" ]
    fontStyle FontStyle {
    }
  }
}
```

```
Transform {
  translation -9 0 0
  children Shape{
    geometry Text{
      string [ "VRML", "is the wave", "of the future" ]
      fontStyle FontStyle {
        family "SANS"
        style "BOLDITALIC"
      }
    }
  }
}
```

```
118
}
```

```
Transform {
  translation -2 0 0
  children Shape{
    geometry Text{
      string [ "VRML", "is the wave", "of the future" ]
      fontStyle FontStyle {
        size 2
      }
    }
  }
}
```

```

Transform {
translation 10 0 0
children Shape{
geometry Text{
string [ "VRML", "is the wave", "of the future" ]
fontStyle FontStyle {
size 2
spacing 1.5
}
}
}
}

```

Thus we don't need to forget that Virtual reality can be defined as an upcoming technology that makes users feel in a **Virtual Environment** (VE) by using computer hardware and software. It was originally conceived as a digitally created space which humans could access by donning special computer equipments. It enables people to deal with information more easily. VR provides a different way to see and experience information, one that is dynamic and immediate. For example, in a computer game, user's joystick motions are tracked and the objects in the game are moved according to the joystick movements. In the same way a simulated, three-dimensional world is created around the user in which he/she could interact with objects, people, and environments. Typically three-dimensional life-sized images with support of audio devices are presented around the user and the perspective is modified in accordance with the user input (generally head or eye movements). Many devices along with the computers are used to create a virtual environment.

To enter in a VE, a user dons special gloves, earphones, and goggles, all of which send their output to the computer systems. The virtual environments are intended to replace the real world environment with the digital one and the human senses are immersed in the VE. Immersion is an experience of losing oneself in the VE and shutting out all cues from the physical world. A Virtual Environment can be created on different extents

depending on the computer based platform ranging from a cell phone screen to a desktop monitor or a fully **Immersive Virtual Environment (IVE)**.

119

The tracking and rendering turns the whole process more immersive and interactive than the traditional media like televisions and video games. The user actions result in immediate and observable impact on the content of virtual environment. Following are the main components of a virtual environment:

1. The visual displays that immerse the user in the virtual world and block out contradictory sensory impressions from the real world.
2. The graphics rendering system that generates the ever changing images at 20 to 30 frames per second.
3. A tracking system that continuously informs the position and orientation of the user's movements.
4. The database construction and maintenance system to build and maintain a detailed and realistic model of the virtual world.
5. A sound system that can produce high quality directional sounds and simulated sound fields.
6. Devices like tracked gloves with pushbuttons to enable users to specify their interactions with the virtual objects.

Further Reading

1. Virtual Reality: The Revolutionary Technology of Computer-Generated Artificial Worlds - and How It Promises to Transform Society by Howard Rheingold (Aug 15, 1992)
2. Infinite Reality: The Hidden Blueprint of Our Virtual Lives (P.S.) by Jim Blascovich and Jeremy Bailenson (Feb 14, 2012)
3. Understanding Virtual Reality: Interface, Application, and Design (The Morgan Kaufmann Series in Computer Graphics) by William R. Sherman and Alan B. Craig (2002)

4. Virtual Reality Technology, Second Edition with CD-ROM by Grigore C. Burdea and Philippe Coiffet (Jun 2003)
5. Developing Virtual Reality Applications: Foundations of Effective Design by Alan B. Craig, William R. Sherman and Jeffrey D. Will (Aug 7, 2009)
6. Virtual Reality by Darren G. Burton (Oct 15, 2010)
7. The Metaphysics of Virtual Reality by Michael Heim (Oct 27, 1994)
8. Virtual Reality by David Harrison, Mark Jacques and Harrison (Jan 1995)
9. The End of Hardware, 3rd Edition: Augmented Reality and Beyond by Rolf R. Hainich (Apr 27, 2009)

