CHAPTER 14

VIRTUAL REALITY—A NEW TECHNOLOGY FOR THE MECHANICAL ENGINEER

Tushar H. Dani
Rajit Gadh
Department of Mechanical Engineering
University of Wisconsin—Madison
Madison, Wisconsin

14.1 INTRODUCTION

In recent times, the term *virtual* has seen increasing usage in the mechanical engineering discipline as a qualifier to describe a broad range of technologies. Examples of usage include "*virtual reality,*" "*virtual prototyping,*" and "*virtual manufacturing.*" In this chapter, the meaning of the term *virtual reality* (VR) is explained and the associated hardware and software technology is described. Next, the role of virtual reality as a tool for the mechanical engineer in the design and manufacturing process is highlighted. Finally, the terms *virtual prototyping* and *virtual manufacturing* are discussed.

14.2 VIRTUAL REALITY

The term *virtual reality* is an oxymoron, as it translates to "reality that does not exist." In practice, however, it refers to a broad range of technologies that have become available in recent years to allow generation of synthetic computer-generated (and hence virtual) environments within which a person can interact with objects as if he or she were in the real world (reality).[1] In other instances, it is used as a qualifier to describe some computer applications, such as a virtual reality system for concept shape design or a virtual reality system for robot path planning.

Hence, the term by itself has no meaning unless it is used in the context of some technology or application. Keeping in mind this association of VR with technology, the next section deals with various elements of VR technology that have developed over the last few years. Note that even though the concept of VR has existed since the late 1980s, only in the last two to three years has it gained a lot of exposure in industry and the media. The main reason for this is that the VR technology has become available at an affordable price so as to be considered a viable tool for interactive design and analysis.

Mechanical Engineers' Handbook, 2nd ed., Edited by Myer Kutz.
ISBN 0-471-13007-9

Later, we will focus on VR applications, which allow such VR technology to be put to good use. In particular, a VR-based application is compared to a typical three-dimensional (3D) computer-aided-design (CAD) application to highlight the similarities and differences between them.

14.3 VR TECHNOLOGY

Typically, in the print media or television, images of VR include glove-type devices and/or so-called head mounted displays (HMDs). Though the glove and HMD are not the only devices that can be used in a virtual environment (VE), they do convey to the viewer the essential features associated with a VE: a high degree of immersion, and interactivity.

Immersion refers to the ability of the synthetic environment to cause the user to feel as if he or she is in a computer-generated virtual world. The immersive capabilities can be judged, for example, by the quality of graphics presented (how real does the scene look?) or by the types of devices used (HMD, for example). All VEs need not be immersive, as will become clearer from later sections.

Interactivity is determined by the extent to which the user can interact with the virtual world being presented and the ways he or she can interact with the virtual world: for example, how the user can interact with the VE (using the glove) and the speed with which the scene is updated in response to user actions. This display update rate becomes an important ergonomic factor, especially in immersive systems, where a lag between the user's actions and the scene displayed can cause nausea.

With reference to the typical glove/HMD combination, the glove-type device is used to replace the mouse/keyboard input and provides the interactivity, while the HMD is used to provide the immersion. Though the glove and head-mounted display combination are the most visible elements of a VR system, there are other components of a VR that must be considered. First, the glove and HMD are not the only devices that can be used in a VE. There are many other devices in the market that can be used for providing the 3D interactions capabilities. These are discussed in Section 14.3.1.

Second, the software in a VR system plays an equally important role in determining the behavior of the system, is discussed. A wide variety of software tools for VR system are described in Section 14.3.2.

Third, the need for real-time performance, combined with the need to interface with a wide range of devices, requires that special attention be paid to the architecture of a VR system. An example of a typical VR system architecture is provided in Section 14.3.1.

14.3.1 VR Hardware

The hardware in a VE consists of three components: the main processor, input devices, and output devices (Fig. 14.1). In the initial stages of VR technology development, in the 1990s, there was a limited choice of computer systems that could be used for VR applications. Currently, all major UNIX workstation vendors have specific platforms targeted to the VR market. These workstations usually have a enhanced graphics performance and specific hardware to support VR-type activity. However, with improvements in the processing speeds, of PCs, they are also becoming viable alternatives to more expensive UNIX-based systems. With prices much lower than their workstation counterparts, these are popular with VR enthusiasts and researchers (with limited budgets) alike. The popularity of the PC-based VR systems has spawned a whole range of affordable PC-based VR interaction devices, some examples of which are provided in this section.

Main Processor

The main processor or virtual environment generator[2] creates the virtual environment and handles the interactions with the user. It provides the computing power to run the various aspects of the virtual world simulation.

The first task of the virtual environment generator is to display the virtual world. An important factor to consider in the display process is the number of frames per second of the scene that can be displayed. Since the goal of a VE is to look and feel like a real environment, the main processor must be sufficiently powerful (computationally) to be able to render the scene at an acceptable frame rate. A measure of the speed of such a processor is the number of shaded polygons it can render per second. Typical speeds for UNIX-based Silicon Graphics machines range from 60,000 Tmesh/sec (Triangular Mesh) for an Indigo2XL to 1.6 million Tmesh/sec for a Power Onyx/12.[3]

The second task of the main processor is to interface with the different input and output devices that are so important in providing the interactiveness in the VE. Depending on the platform used, a wide range of input and output devices are available. A brief summary of such devices is provided in the next two sections. Detailed description of such devices and hardware can be found in Ref. 4.

Input Devices

Input devices provide the means for the user to interact with the virtual world. The virtual world, in turn, responds to the user's actions by sending feedback through various output devices, such as a visual display. Since the principal objective of a VE is to provide realistic interaction with the virtual

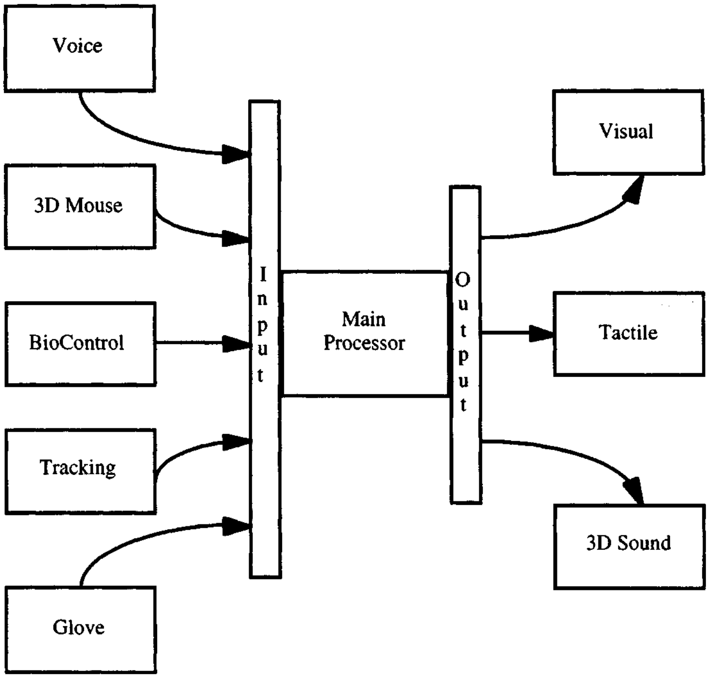

Fig. 14.1 Hardware in a VR system.

world, input devices play an important role in a VR system. The mouse/keyboard interaction is still used in some VR environments, but the new generation of 3D devices that provide the tools to reach into the 3D virtual world.

Based on their usage, input devices can be grouped into five categories: tracking, pointing, hand-input, voice-based, and devices based on bio-sensors. Of these, the first four types are typically used in VR systems. Note that of the devices described below, only the devices in the first three categories are used in VEs.

Tracking Devices. These devices are used in position and orientation tracking of a user's head and/or hand. These data are then used to update the virtual world scene. The tracker is sometimes also used to track the user's hand position (usually wearing a glove; see below) in space so that interactions with objects in the 3D world are possible. Tracking sensors based on mechanical, ultrasonic, magnetic, and optical systems are available. One example of such a device is the Ascension tracker.[5]

Point Input Devices. These devices have been adapted from the mouse/trackball technology to provide a more advanced form of data input. Included in this category is the 6-degree of freedom (6-dof) mouse and force ball. The 6-dof mouse functions like a normal mouse on the desktop but as a 6-dof device once lifted off the desktop. A force ball uses mechanical strains developed to measure the forces and torques the user applies in each of the possible three directions. An example of force ball-type technology is the SpaceBall. Another device that behaves like a 6-dof mouse is the Logitech Flying Mouse, which looks like a mouse but uses ultrasonic waves for tracking position in 3D space.

Glove-Type Devices. These consist of a wired cloth glove that is worn over the hand like a normal glove. Fiber-optical, electrical, or resistive sensors are used to measure the position of the joints of the fingers. The glove is used as a gestural input device in the VE. This usually requires the development of gesture-recognition software to interpret the gestures and translate them into commands the VR software can understand. The glove is typically used along with a tracking device that measures the position and orientation of the glove in 3D space. Note that some gloves do provide some rudimentary form of tracking and hence do not require the use of a separate tracking device. One example of such a glove is the PowerGlove[6] which is quite popular with VR home enthusiasts since it is very affordable. Other costlier and more sophisticated versions, such as the CyberGlove, are also available.

Biocontrollers. Biocontrollers process indirect activity, such as muscle movements and electrical signals produced as a consequence of muscle movement. As an example, dermal electrodes placed near the eye to detect muscle activity could be used to navigate through the virtual worlds by simple eye movements. Such devices are still in the testing and development stage and are not quite as popular as the devices mentioned earlier.

Audio Devices. Voice input provides a more convenient way for the user to interact with the VE by freeing his or her hands for use with other input devices. Such an input mechanism is very useful in a VR environment because it does not require any additional hardware, such as the glove or biocontrollers, to be physically attached to the user. Voice-recognition technology has evolved to the point where such software can be bought off the shelf. An example of such a software is VoiceAssist from SoundBlaster.

Output Devices

Output devices are used to provide the user with feedback about his or her actions in the VE. The ways in which the user can perceive the virtual world are limited to the five primary senses of sight, sound, touch, smell, and taste. Of these only the first three have been incorporated in commercial output devices. Visual output remains the primary source of feedback to the user, though sound can also be used to provide cues about object selection, collisions, etc.

Graphics. Two types of technologies are available for visual feedback. The first, HMD (head-mounted display), is mentioned in Section 14.3. It typically uses two liquid crystal display (LCD) screens to show independent views (one for each eye). The human brain puts these two images together to create a 3D view of the virtual world. Though head-mounted displays provide immersion, they currently suffer from poor resolution, poor image quality, and high cost. They are also quite cumbersome and uncomfortable to use for extended periods of time.

The second and much cheaper method is to use a stereo image display monitor and LCD shutter glasses. In this system, two images (as seen by each eye) of the virtual scene are show alternately at a very high rate on the monitor. An infrared transmitter coordinates this display rate to the frequency with which each of the glasses is blacked out. A 3D image is thus perceived by the user. One such popular device is the StereoGraphics EyeGlasses system.[7]

Audio. After sight, sound is the most important sensory channel for virtual experiences. It has the advantage of being a channel of communication that can be processed in parallel with visual information. The most apparent use is to provide auditory feedback to the user about his or her actions in the virtual world. An example is to provide audio cues if a collision occurs or an object is successfully selected. Three-dimensional sound, in which the different sounds would appear to come from separate locations, can be used to provide a more realistic VR experience. Since most workstations and PCs nowadays are equipped with sound cards, incorporating sound into the VE is thus not a difficult task.

Contact. This type of feedback could either be touch or force.[8] Such tactile feedback devices allow a user to feel forces and resistance of objects in the virtual environment. One method of simulating different textures for tactile feedback is to use electrical signals on the fingertips. Another approach has been to use inflatable air pockets in a glove to provide touch feedback. For force feedback, some kind of mechanical device (arm) is used to provide resistance as the user tries to manipulate objects in the virtual world. An example of such a device is the PHANToM haptic interface, which allows a user to "feel" virtual objects.[9]

14.3.2 VR Software

As should be clear from the preceding discussion, VR technology provides the tools for an enhanced level of interaction in three dimensions with the computer. The need for real-time performance while depicting complex virtual environments and the ability to interface to a wide variety of specialized devices require VR software to have features that are clearly not needed in typical computer applications. Existing approaches to VR content creation have typically taken the following approaches[10]: virtual world authoring tools and VR toolkits. A third category is the Virtual Reality Modeling Language (VRML) and the associated "viewers" which are rapidly becoming a standard way for users to share "virtual worlds" across the World Wide Web.

Virtual World Authoring and Playback Tools

One approach to designing VR applications is first to create the virtual world that the user will experience (including ascribing behavior to objects in that world) and then to use this as an input to a separate "playback" application. The "playback" is not strictly a playback in the sense that users are still allowed to move about and interact in the virtual world. An example of this would be a walk-through kind of application, where a static model of a house can be created (authored) and the user can then visualize and interact with it using VR devices (the playback application).

Authoring tools usually allow creation of virtual worlds using the mouse and keyboard and without requiring programs in C or C++. However, this ease of use comes at the cost of flexibility, in the sense that the user may not have complete control over the virtual world being played back. Yet such systems are popular when a high degree of user interaction, such as allowing the user to change the virtual environment on the fly, is not important to the application being developed and when programming in C or C++ is not desired. Examples of such tools are the SuperScape,[11] Virtus,[12] and VREAM[13] systems.

VR Toolkits

VR Toolkits usually consist of programming libraries in C or C++ that provide a set of functions that handle several aspects of the interaction within the virtual environment. They are usually used to develop custom VR applications with a higher degree of user interaction than the walk-through applications mentioned above. An example of this would be a VR-based driver training system, where in addition to the visual rendering, vehicle kinematics and dynamics must also be simulated.

In general, VR toolkits provide functions that include the handling of input/output devices and geometry creation facilities. The toolkits typically provide built-in device drivers for interfacing with a wide range of commercial input and output devices, thus saving the need for the programmer to be familiar with the characteristics of each device. They also provide rendering functions such as shading and texturing. In addition, the toolkits may also provide functions to create new types of objects or geometry interactively in the virtual environment. Examples of such toolkits include the dVise library[14] the WorldToolKit library,[15] and Autodesk's Cyberspace Development Kit.[16]

VRML

The Virtual Reality Modeling Language (VRML) is a relative newcomer in the field of VR software. It was originally conceptualized as a language for Internet-based VR applications but is gaining popularity as a possible tool for distributed design over the Internet and World Wide Web.

VRML is the language used to describe a virtual scene. The description thus created is then fed into a VRML viewer (or VRML browser) to view and interact with the scene. In some respects, VRML can be thought of as fitting into the category of virtual world authoring tools and playback discussed above. Though the attempt to integrate CAD into VRML is still in the initial phase, it certainly offers new and interesting possibilities. For example, different components of a product may be designed in physically different locations. All of these could be linked together (using the Internet) and viewed through a VRML viewer (with all the advantages of a 3D interactive environment), and any changes could be directed to the person in charge of designing that particular component. Further details on VRML can be found at the VRML site.[17]

14.4 VR SYSTEM ARCHITECTURE

To understand the architectural requirements of a VR system, it will be instructive to compare it with a standard 3D CAD application. A typical CAD software program consists of three basic components: the user input processing component, the application component, and the output component. The input processing component captures and processes the user input (typically from the mouse/keyboard) and provides these data to the application component. The application component allows the user to model and edit the geometry being designed until a satisfactory result is obtained. The output component provides a graphical representation of the model the user is creating (typically on a computer screen).

For a VR system, components similar to those in CAD software can be identified. One major difference between a traditional CAD system and a VR-based application system is obviously the input and output devices provided. Keeping in mind the need for realism, it is imperative to maintain a reasonable performance for the VR application. Here "performance" refers to the response of the virtual environment to the user's actions. For example, if there is too much lag between the time a person moves his or her hand and the time the image of the hand is updated on the display, the user will get disoriented very quickly.

One way to overcome this difficulty is to maintain a high frame rate (i.e., number of screen updates per second) for providing the graphical output. This can be achieved by distributing the input processing, geometric modeling, and output processing tasks amongst different processors. The reason for distributing the tasks is to reduce the computational load on the main processor (Fig. 14.2).

Typical approaches adopted are to run the input and output processing component on another processor (Windows-based PC or a Macintosh) while doing the display on the main processor. In addition to reducing the computational workload on the main processor, another benefit of running the input component on a PC is that there are a wide variety of devices available for the PC platform, as opposed to the UNIX platform. This also has an important practical advantage in that a much

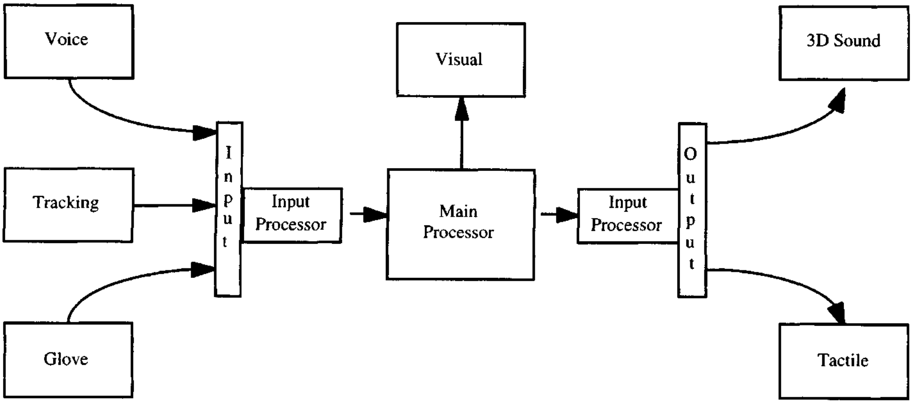

Fig. 14.2 VR system architecture.

wider (and cheaper) range of devices is available for a PC or Macintosh than for its workstation counterparts.

14.5 THREE-DIMENSIONAL COMPUTER GRAPHICS vs. VR

We will now consider virtual reality in the context of applications. So far it has been stressed that VR applications must provide interactive and immersive environments. However, a typical CAD application is interactive (although based on using a 2D mouse) and can be "used" with StereoGlasses (to provide immersion), and thus can be considered to meet the requirements for a VR system. Yet such CAD systems are not referred to as VR systems, despite providing a "virtual" world where the designer can in effect create objects in 3D.

Thus, there seems to be some kind of basic level of interaction and immersion that must be met before a system can be classified as a VR system (Fig. 14.3). The boundaries between a 3D application and VR are not very clear, but in general, a VR application will require 3D input devices (as opposed to a mouse device) and will also provide enhanced feedback, either sound- or contact-type, in addition to the display (typically stereoscopic). On the basis of the level of realism intended (often proportional to cost) and hardware used, VR systems can be classified as *immersive* or *desktop*.[18]

14.5.1 Immersive VR Systems

In an immersive system, the user's field of view is completely surrounded by a synthetic, computer-generated 3D environment. Such a system is useful in an application in which it is important that the user perceives that he or she is part of the virtual environment; for example, an application that allows a student driver to obtain training in a virtual environment. VR devices such as head-position trackers, head-mounted displays, and data gloves are commonly used in such systems to give a feeling of immersion. As the aim of these systems is to provide realism to the user, they require the use of very high-speed computers and other expensive hardware. Typical examples of such applications include virtual walk-throughs of buildings or driving a virtual vehicle.[2]

Fig. 14.3 CAD vs. Desktop vs. Immersive Systems.

14.5.2 Desktop VR Systems

Desktop VR systems are typically more economical than immersive systems. Desktop systems let users view and interact with entities in a 3D environment using a stereo display monitor and stereo glasses. Such systems are adequate for tasks where immersion is not essential, such as CAD. For interaction with the 3D world, devices like the Spaceball and/or gloves can be used. Since desktop-based VR environments do not need devices such as head-mounted displays, they are simpler and cheaper to implement than immersive systems. Additional features such as voice recognition capability and sound output can further enhance the usability of a desktop system without requiring the use of significant additional hardware.

14.5.3 Hybrid Systems

A new category of VR systems, which can be classified as hybrid systems, attempts to preserve the benefits of HMD-based systems, such as higher degree of immersion, with the comfort of desktop VR systems. Since HMD's can be cumbersome to use, hybrid systems provide immersion by using projectors to display the computer images (usually stereoscopic) on a large screen. These can be in either a vertical (wall) or horizontal (table) configuration. As in desktop systems, they typically require the user to wear the lightweight LCD glasses and use standard position trackers to track the user's head and hand position. Examples of this approach are the CAVE[19] system developed at the University of Chicago, Virtual Workbench[20] developed by Fakespace Inc., and the Virtual Design Studio project at University of Wisconsin-Madison.[21] A comprehensive list of projection-based VR systems can be found at the Projected VR Systems site.[22]

14.6 VR FOR MECHANICAL ENGINEERING

Until recently, the usability of CAD systems has been constrained by the lack of appropriate hardware devices to interact with computer models in three dimensions and the lack of software that exploits the advantages of this new generation of devices. Thus, the user interface for CAD programs has remained essentially the mouse/keyboard/menu paradigm.

The availability of VR technology has allowed the user interface to expand beyond the realm of the mouse and keyboard. Consequently, new software has evolved that allows the usage of such devices in various tasks. The VR systems for mechanical engineering can be divided into two categories, depending on the amount of interactivity possible between the user and the design environment: (1) those that support visualization and (2) those that allow design activity of some type. A summary of how VR is being used in other CAD/CAM applications can be found at the National Institute of Standards and Technology (NIST) World Wide Web site.[23]

14.6.1 Enhanced Visualization

Enhanced visualization systems allow the user to view the CAD model in a 3D environment to get a better idea of the shape features of the parts and their relationships. Models created in existing CAD systems are "imported," after an appropriate translation process, into a VR environment. Once the part is imported into the VR environment, 3D interaction devices such as gloves and 3D display monitors can be used to examine the models in a "true" 3D environment.

Enhanced visualization systems typically use 3D navigational devices such as Spaceball, flying mouse, etc., and stereo monitors with shutter eyeglasses, to allow enhanced visualization of a product or prototype. The Mitre corporation[24] has developed several virtual environments, including the Microdesigner, which enable a designer to review 3D designs. Researchers at Sun Microsystems[25] have developed a Virtual Lathe with which a user can view the action of a cutting tool and control the tool in 3D. Other examples of such work include the VENUS project[26] and research at Clemson University.[27]

14.6.2 VR–CAD

The second category of software allows design activity in the VR environment. The advantage of design activity (as opposed to just visualization) in a VR environment is that the designer is no longer limited to a traditional 2D interface when making 3D designs. Such systems use a variety of input devices (gloves, 3D navigation devices, etc.) to provide a 3D interface for design and interaction. In addition, they also support alternative methods of user input, such as voice and gestures.

Examples of VR–CAD systems include the DesignSpace[28] system, currently under development at Stanford University. It allows conceptual design and assembly, using voice and gestures in a networked virtual environment. Another system that allows design is the Virtual Workshop[29] developed at MIT, which allows parts to be created in a virtual metal and woodworking shop. Other systems include the 3-Draw system[30] and JDCAD system.[31] The 3-Draw system uses a 3D input device to let the designer sketch out ideas in three dimensions. The JDCAD system uses a pair of 3D input devices and 3D user interface menus to allow design of components.

The authors are currently developing a system called Conceptual Virtual Design System (COVIRDS), a VR system that allows the designer to create concept shape designs in a 3D environment. COVIRDS[32] is designed to solve some of the limitations of existing CAD systems. It has an intuitive interface so that designers without CAD system expertise can use the computer to create concept shapes using natural interaction mechanisms, such as voice commands and gestures.

14.7 VIRTUAL PROTOTYPING/MANUFACTURING AND VR

The terms *virtual prototyping* and *virtual manufacturing* are commonly used in academia and industry and can be easily confused with virtual reality (technology or applications). *Virtual,* as used in virtual prototyping or virtual manufacturing, refers to the use of a computer to make a prototype or aid in manufacturing a product. The discussion below applies both to virtual prototyping and virtual manufacturing.

Virtual prototyping refers to the design and analysis of a product without actually making a physical prototype of the part. *Virtual* here refers to the fact that the result of the design is not yet created in its final form, only a visual representation of the object that is presented to the user for observation, analysis, and manipulation. This prototype does not necessarily have all the features of the final product, but has enough of the key features to allow testing of the product design against the product requirements.

The simplest example of virtual prototyping tool is a 3D CAD system that allows a user to design, create, and analyze a part. However, since a 3D model is difficult to visualize on a 2D screen, one approach that has developed is to use a VR-based design and visualization system. The VR-based CAD system (as discussed in Section 14.6.2) allows changes to the "virtual prototype" to be made instantaneously, thus allowing the designer to experiment with different shapes in a short period of time. The importance of getting an optimum design lies in the fact that once the concept design is decided 60–70% of the costs of a product are committed. A poor design decision may result in increasing the downstream (committed) cost significantly. Hence, VR can be used as a tool to facilitate virtual prototyping and manufacturing. However, note that virtual prototyping or manufacturing does not require the use of VR. For more details on virtual manufacturing/prototyping see Ref. 33.

REFERENCES

1. N. I. Durlach and A. S. Mavor (eds.), "Virtual Reality: Scientific and Technological Challenges," in *National Research Council*, National Academic Press, Washington, DC, 1995.
2. R. S. Kalawsky, *The Science of Virtual Reality and Virtual Environments*, Addison-Wesley, New York, 1993.
3. Silicon Graphics Computer Systems, Periodic Table, WWW URL: ftp://ftp.sgi.com/sgi/Periodic Table.ps.Z.
4. K. Pimental and K. Teixeira, *Virtual Reality: Through the New Looking Glass*, Intel/Windcrest/McGraw-Hill, New York, 1993.
5. Ascension Technology Home Page, WWW URL: http://world.std.com/~ascen
6. Mattel PowerGlove Home Page, WWW URL: http://www.spies.com/jet/vr/vr.html
7. StereoGraphics Corp. Home Page, WWW URL: http://www.stereographics.com
8. M. A. Gigante, "Virtual Reality: Enabling Technologies," in *Virtual Reality Systems*, E. A. Earnshaw (ed.), Academic Press, 1993.
9. T. H. Massie and J. K. Salisbury, "The PHANToM Haptic Interface: A Device for Probing Virtual Objects," in *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interface for Virtual Environment and Teleoperator Systems*, Chicago, November 1994, WWW URL: http://www.mit.edu:8001/people/proven/Phantom/
10. J. Isdale, "What Is Virtual Reality," On-line Document, WWW URL: ftp://ftp.hitl.washington.edu/pub/scivw/WWW/scivw.html
11. Superscape Home Page, WWW URL: http://www.superscape.com/index.html
12. Virtus Corp. Home Page, WWW URL: http://www.virtus.com
13. VREAM V1.1 Webview and VR Creator (VREAM Inc.) Home Page, WWW URL: http://www.vream.com/vream/vream.html
14. Division Ltd. Home Page, WWW URL: http//www.division.com
15. Sense8 Corp. Home Page, WWW URL: http://www.sense8.com
16. Autodesk CDK Home Page, WWW URL: http://www.autodesk.com/prod/mm/cyber.htm
17. Virtual Reality Modeling Language, WWW URL: http://www.wired.com/vrml
18. L. Jacobson, "Virtual Reality: A Status Report," *AI Expert* **6** (8), 26–33 (August 1991).
19. The CAVE System—http://evlweb.eecs.uic.edu/pape/CAVE.
20. The Virtual WorkBench—http://www.fakespace.com/new_pro.html

21. Virtual Design Studio—http://icarve.me.wisc.edu/groups/virtual
22. Projected VR—http://evlweb.eecs.uic.edu/pape/CAVE/projectedVR.html
23. S. Ressler, "Virtual Reality for Manufacturing—Case Studies," WWW URL: http://nemo.ncsl.nist.gov/sressler/projects/mfg/
24. P. T. Breen Jr., "The Reality of Fantasy: Real Applications for Virtual Environments," *Information Display,* **11** (8), 15–18 (1992).
25. T. Studt, "REALITY: From Toys to Research Tools," *R&D Magazine* (March 1993).
26. VENUS, Virtual Prototyping Project, CERN, Switzerland, WWW URL: http://sgvenus.cern.ch/VENUS/vr_project.html
27. D. Fadel et al., "A Link between Virtual and Physical Prototyping," in *Proceeedings of the SME Rapid Prototyping and Manufacturing Conference*, Detroit, May 2–4, 1995, WWW URL: http://fantasia.eng.clemson.edu/vr/research.html
28. W. L. Chapin, T. A. Lacey, and L. Leifer, "DesignSpace — A Manual Interaction Environment for Computer-Aided Design," in *Proceedings of the ACM SIGCHI 1994 Conference: CHI'94 Human Factors in Computing Systems*, Boston, WWW URL: http://gummo.stanford.edu/html/DesignSpace/home.html
29. J. W. Barrus and W. Flowers, "The Virtual Workshop: A Simulated Environment for Mechanical Design," in *SIGGRAPH '94 Proceedings.*
30. E. Sachs, A. Roberts, and D. Stoops, "3-Draw: A Tool for Designing 3D Shapes," *IEEE Computer Graphics and Applications* **11** (11), 18–26 (1991).
31. J. Liang, "JDCAD: A Highly Interactive 3D Modeling System," *Computers and Graphics* **18**, 499–506 (1994).
32. T. H. Dani and R. Gadh, "COVIRDS : A Conceptual Virtual Design System," in *Proceedings of the Computers in Engineering Conference and the Engineering Database Symposium of the ASME*, Boston, 1995, WWW URL: http://icarve.me.wisc.edu/groups/virtual/
33. W. E. Alzheimer, M. Shahinpoor, and S. L. Stanton (eds.), "Virtual Manufacturing," in *Proceedings of the 2nd Agile Manufacturing Conference (AMC '95)*, Albuquerque, NM, March 16–17, 1995.

BIBLIOGRAPHY

Human Interface Technology Lab Home Page (T. Emerson, Librarian), University of Washington, Seattle, WWW URL: http://www.hitl.washington.edu/people/diderot

Virtual IO Home Page, WWW URL: http://www.vio.com

K. Warwick, J. Gray, and D. Roberts (eds.), *Virtual Reality in Engineering*, Institution of Electrical Engineers, London, 1993.

Bowman, D., Conceptual Design Space Project, WWW URL: http://www.cc.gatech.edu/gvu/virtual/CDS/

Deitz, D., "Educating Engineers for the Digital Age," *Mechanical Engineering* **117** (9), 77–80 (September 1995).