Doug A. Bowman
bowman@vt.edu
Department of Computer Science
Virginia Polytechnic Institute and
State University

Ernst Kruijff
ernst.kruijff@archit.uni-weimar.de
InfAR-computer science in
architecture
Bauhaus Universitaet Weimar

Joseph J. LaViola, Jr.
jjl@cs.brown.edu
Computer Science Department
Brown University

Ivan Poupyrev
poup@csl.sony.co.jp
Sony Computer Science Lab
Tokyo, Japan

An Introduction to 3-D User Interface Design

Abstract

Three-dimensional user interface design is a critical component of any virtual environment (VE) application. In this paper, we present a broad overview of 3-D interaction and user interfaces. We discuss the effect of common VE hardware devices on user interaction, as well as interaction techniques for generic 3-D tasks and the use of traditional 2-D interaction styles in 3-D environments. We divide most user-interaction tasks into three categories: navigation, selection/manipulation, and system control. Throughout the paper, our focus is on presenting not only the available techniques but also practical guidelines for 3-D interaction design and widely held myths. Finally, we briefly discuss two approaches to 3-D interaction design and some example applications with complex 3-D interaction requirements. We also present an annotated online bibliography as a reference companion to this article.

1 Introduction

User interfaces (UIs) for computer applications are becoming more diverse. Mice, keyboards, windows, menus, and icons—the standard parts of traditional WIMP interfaces—are still prevalent, but nontraditional devices and interface components are proliferating rapidly. These include spatial input devices such as trackers, 3-D pointing devices, and whole-hand devices allowing gestural input. Three-dimensional, multisensory output technologies—such as stereoscopic projection displays, head-mounted displays (HMDs), spatial audio systems, and haptic devices—are also becoming more common.

With this new technology, new problems have also been revealed. People often find it inherently difficult to understand 3-D spaces and to perform actions in free space (Herndon, van Dam, & Gleicher, 1994). Although we live and act in a 3-D world, the physical world contains many more cues for understanding and constraints and affordances for action that cannot currently be represented accurately in a computer simulation. Therefore, great care must go into the design of user interfaces and interaction techniques for 3-D applications. It is clear that simply adapting traditional WIMP interaction styles to three dimensions does not provide a complete solution to this problem. Rather, novel 3-D user interfaces, based on real-world interaction or some other metaphor, must be developed.

This paper is a broad overview of the current state of the art in 3-D user interfaces and interaction. It summarizes some of the major components of tutorials and courses given by the authors at various conferences, including the 1999 Symposium on Virtual Reality Software and Technology. Our goals are

Presence, *Vol. 10, No. 1, February 2001,* 96–108

to describe some of the major interaction techniques and interface components that have been developed for 3-D systems, especially immersive virtual environments (VEs), and to provide the reader with some practical guidelines for the design of 3-D user interfaces in real-world applications.

It is impractical to list even a small fraction of the reference materials from which this overview is taken, but we refer readers to the "20th Century 3DUI Bibliography" (Poupyrev & Kruijff, 2000), a continuously updated and annotated list of references that is available online. This bibliography was created both by the authors and by other members of the 3DUI mailing list, an online community of researchers in this field.

2 Input and Output Devices

Input and output (I/O) devices are an important component in building 3-D user interfaces for VE applications. Interaction designers must have a thorough understanding of the ergonomics, advantages, and limitations of the devices used so they can find natural and intuitive mappings between interaction techniques and hardware.

Literally hundreds of different I/O devices are used in VE applications, and a thorough analysis of them is beyond the scope of this paper. (See Youngblut, Johnson, Nash, Wienclaw, and Will (1996) for a somewhat dated but thorough discussion on many I/O devices.) However, many of these devices can be classified into groups with common characteristics and interaction design criteria. In the following subsections, we examine these groups and discuss interface design issues within them.

2.1 Output Devices

We commonly use the term *display* to describe output. Although the most common displays are visual, there are other important displays for the auditory, haptic, tactile, and olfactory channels. In the context of 3-D user interfaces for VEs, visual displays can be roughly categorized into fully immersive and semi-immersive devices. Fully immersive displays—such as head-mounted displays (HMDs), arm-mounted displays, and virtual retinal displays—occlude the real world. (Augmented realty is an exception to this rule.) Because of this, physical objects require a graphical representation in the virtual world. In addition, more-complicated input devices may be difficult to use because they cannot be seen.

Semi-immersive displays—such as stereo monitors, workbenches, and surround-screen virtual reality systems—allow the user to see both the physical and virtual world. A number of interface issues arise with semi-immersive displays. For instance, because users can see their own hands in front of the display, they can inadvertently block out virtual objects that should appear to be closer than their hands. This problem occurs frequently during manipulation tasks and can be alleviated if the virtual object is offset from the hand so that it is always visible. Another problem relates to the use of stereo shutter glasses. Because these require emitters, any line-of-sight occlusion will disrupt the stereo effect. Interface designers should try to avoid interaction techniques that cause users to move their hands or physical objects in the line of sight of the glasses and emitters.

In addition to visual displays, auditory, tactile, and haptic displays are becoming more prominent in 3-D user interfaces for VEs. With auditory displays (Begault, 1994), the main interface goals are localization, the generation of 3-D sound, and sonification (transforming certain types of information into sound). Auditory output is especially useful in collaborative interfaces in which participants can get a sense for where others are in the virtual world. It can also be used to substitute for missing tactile or haptic feedback. For example, a sound could substitute for the feel of pressing a button. Haptic and tactile displays (Burdea, 1996) are an important component of 3-D user interfaces for VEs, and this is becoming an active area of research. Allowing users to touch and feel in the virtual world can be extremely powerful, especially for object manipulation and collision detection.

2.2 Input Devices

A distinction must be made between input devices and interaction techniques. Input devices are just the physical tools used to implement various interaction techniques. In general, many different interaction techniques can be mapped onto a given input device. The question is how naturally, efficiently, and appropriately a given input device will work with a given technique. Input devices are also governed by the degrees of freedom (DOFs) they have. In general, an input device with a smaller number of DOFs can be used to emulate a device with more DOFs with the addition of buttons or modifier keys.

Input devices can be roughly categorized based on the types of events they generate. Devices that generate one event at a time based on the user are considered discrete-input devices. A button press or other action signals an event, which is usually a Boolean (up/down) value. Pinch gloves, developed by Fakespace, are an example of a discrete-input device (the user pinches two or more fingers to signal an event). In contrast to discrete-input devices, continuous-input devices generate a stream of events. Two of the most common continuous-input devices are position/orientation trackers and datagloves, which transmit bend angles of the fingers. Devices that combine both discrete and continuous events to form single, more-flexible devices are called combination or hybrid-input devices. Examples of hybrid devices include the Ring Mouse (a small device worn on the user's finger that combines ultrasonic tracking with two buttons) and pen-based tablets, which are becoming more popular in VE applications because they give users the ability to interact in two dimensions.

Speech input is unique because the "device" is the human voice. Speech provides a nice complement to other input devices and, as a result, it is a natural way to combine different modes of input (multimodal interaction). In general, speech input can be a valuable tool in 3-D user interfaces, especially when both the user's hands are occupied. A major myth of speech recognition is that having a good speech recognizer will solve all problems. Unfortunately, many other issues need to be considered when dealing with speech input. Letting the computer know when to listen to the user's voice is one such issue. A user may issue a speech command unintentionally when talking to another person. One of the best ways to avoid this problem is to use an implicit or invisible push-to-talk scheme (LaViola, 1999). A push-to-talk interface lets users tell the application when they are speaking to it. In order to maintain the naturalism of the speech interface, one can embed the "push" into existing interaction techniques so the user does not have the burden of remembering the signal to the application that a voice command is about to be issued.

3 Interaction Techniques

3.1 Navigation

The task of navigation is the most prevalent user action in most large-scale 3-D environments, and it presents challenges such as supporting spatial awareness, providing efficient and comfortable movement between distant locations, and making navigation lightweight so that users can focus on more-important tasks. We subdivide navigation into the motor component called *travel* and the cognitive component called *wayfinding*.

Navigation tasks can generally be classified into three categories. *Exploration* is navigation with no explicit target: the user is simply investigating the environment. *Search* tasks involve moving to a particular target location. Finally, *maneuvering* tasks are characterized by short-range, high-precision movement that are used to place the viewpoint at a more advantageous location for performing a particular task.

Travel is a conceptually simple task—the movement of the viewpoint from one location to another. Further, viewpoint orientation is usually handled in immersive VEs by head tracking, so only techniques for setting viewpoint position need be considered.

There are five common metaphors for travel interaction techniques, and most published interaction techniques for travel fit into one of these five categories:

- **Physical movement:** The user's body motion is used to travel through the environment. Examples include wide-area motion tracking, walking in

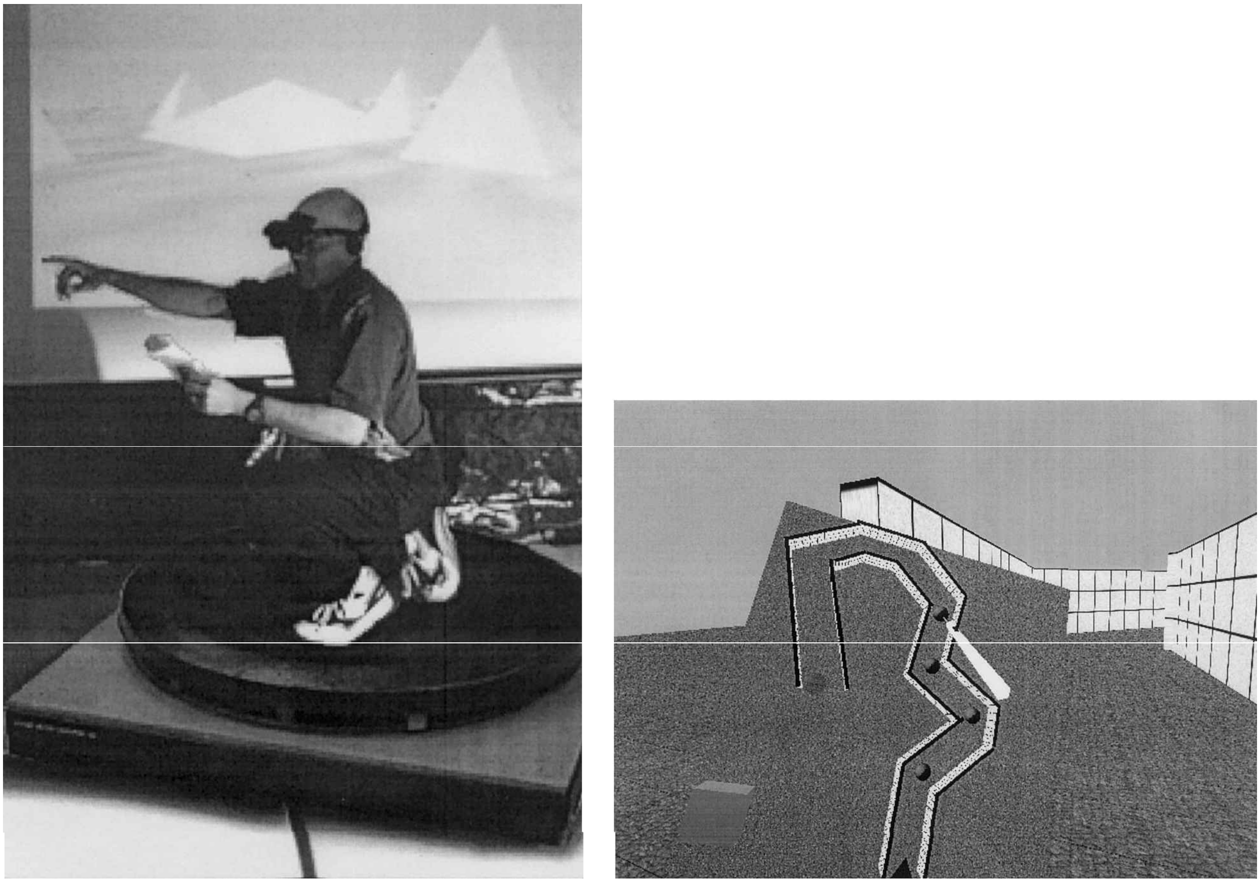

Figure 1. *Example travel techniques. Virtual Motion Controller, a physical movement travel technique (left). Placing markers on a map in a route-planning technique (right).*

place, and locomotion devices (figure 1, left) such as treadmills or stationary bicycles. Such techniques are appropriate when an enhanced sense of presence is required or when the application requires the user to experience physical exertion when traveling.

- **Manual viewpoint manipulation:** The user's hand motions are used to effect travel. For example, the user "grabs the air" and pulls himself along as if with a virtual rope. Another type of technique uses a selected object as a center point around which user motion can be specified. These techniques can be efficient and easy to learn, but they can also cause fatigue.
- **Steering:** Steering is the continuous specification of the direction of motion. This is the most common travel metaphor and includes techniques such as gaze-directed steering (wherein the user's head orientation determines the direction of travel) or pointing (in which hand orientation is used). Steering techniques are general and efficient.
- **Target-based travel:** The user specifies the destination, and the system handles the actual movement. This may take the form of "teleportation," in which the user jumps immediately to the new location, or, preferably, the system may perform some transitional movement between the starting point and the destination. Target-based techniques are very simple from the user's point of view.
- **Route planning:** The user specifies the path that should be taken through the environment, and the

system handles the actual movement. The user may manipulate icons or draw a path on a map of the space or in the actual environment in order to plan a route (figure 1, right). These techniques allow the user to control travel while he retains the ability to perform other tasks during motion.

Besides choosing a metaphor, other design issues for travel techniques include velocity control and the use of constraints or guides to aid travel. Bowman et al. have performed a variety of experimental evaluations (Bowman, Davis, Badre, & Hodges, 1999; Bowman & Hodges, 1997; Bowman, Johnson, & Hodges, 1999; Bowman, Koller, & Hodges, 1997), resulting in quantitative comparisons of techniques and serving as the basis for design guidelines. For example, designers should consider whether travel is used as an end unto itself or in the service of some other task. (Target-based techniques, because of their simplicity, will allow the user to focus on the more relevant task.) Another important guideline is that user training and instruction may be as important as the technique used. Users with sophisticated strategies will remain more spatially oriented than will those with naíve travel strategies.

Wayfinding, the counterpart of travel, can be described as the cognitive process of defining a path through an environment, thereby using and acquiring spatial knowledge to build up a cognitive map of an environment. Spatial knowledge consists of landmark, procedural, and survey knowledge (Thorndyke & Hayes-Roth, 1982). The usage and acquisition of spatial knowledge is influenced by such factors as the reference frame (first-person egocentric or third-person exocentric) and the travel technique.

Users should receive wayfinding support during VE travel. Individual users have a wide range of spatial abilities, and the extra degrees of freedom within a VE can easily cause disorientation. In the case of VE training, with the goal of transferring knowledge from the VE to the real world, the application and environment should be designed to support the transfer (using techniques discussed below); otherwise, the training session can easily become counterproductive.

Wayfinding support can be subdivided into user-centered and environment-centered support. User-centered support includes factors such as a large field of view, the inclusion of visual motion and vestibular (real-motion) cues, and nonvisual support such as audio. These cues may have a substantial impact on the sense of presence, but this requires further research. Environment-centered support can be partitioned into structural organization and cues. Structural organization defines how clearly different parts of the environments can be identified and related to other parts. A good example of structuring is the set of legibility techniques (Ingram & Benford, 1995). Cues consist of real-world wayfinding principles that are transferred to a VE. The most commonly applied cues are artificial, such as maps, compasses, and grids. This category could also include architectural cues like lighting, color, and texture, and natural environment cues like a horizon and atmospheric perspective.

Studies with maps (Darken & Cevik, 1999) in VEs have shown some wayfinding performance gains. Additional evaluations need to be performed to learn how the content of the environment influences the application of cues. Most cues have been applied to environments that are strongly related to real-world environments. In any case, wayfinding support needs to be carefully implemented to allow useful navigation through an environment.

3.2 Selection and Manipulation

Interaction techniques for 3-D manipulation in VEs should provide means to accomplish at least one of three basic tasks: object selection, object positioning, and object rotation. Because direct hand manipulation is a major interaction modality not only in the 3-D virtual world but also in natural physical environments, the design of interaction techniques for object selection and manipulation has a profound effect on the quality of the entire VE user interface.

The classical approach to design manipulation techniques is to provide the user with a “virtual” hand—a 3-D cursor, often shaped like a human hand, whose movements correspond to the movements of the hand tracker (figure 2, left). Selection and manipulation sim-

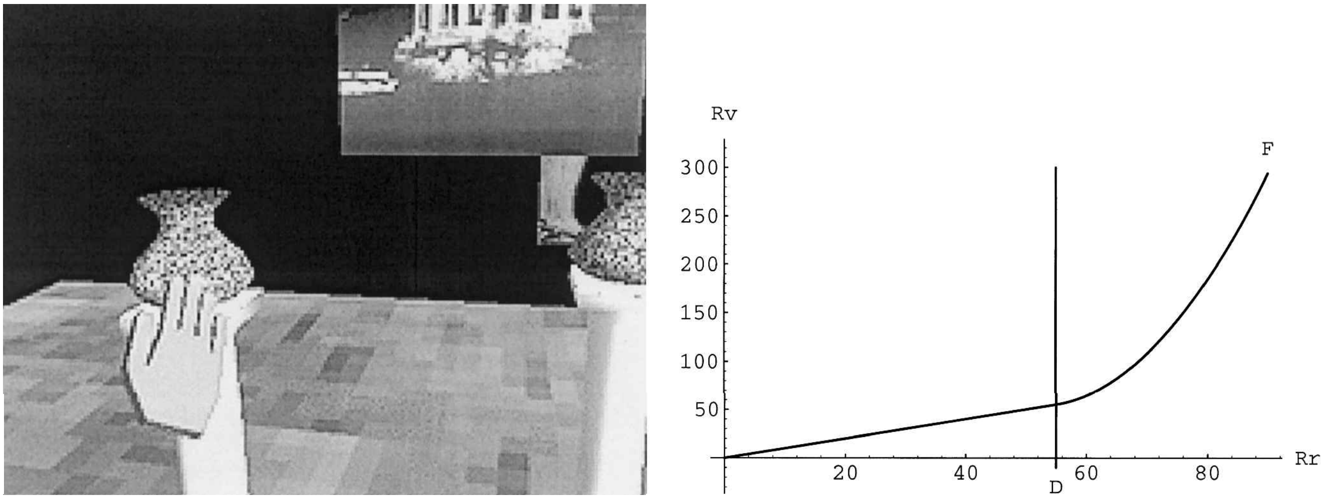

Figure 2. *"Classical" virtual hand technique (left) and nonlinear mapping function of Go-Go technique (right).*

ply involve touching an object, then positioning and orienting this virtual hand within the VE. The virtual hand technique is rather intuitive because it simulates a real-world interaction with objects, but only those objects within the area of reach can be picked up.

A number of techniques have been suggested to overcome this problem. The Go-Go technique (Poupyrev, Billinghurst, Weghorst, & Ichikawa, 1996) allows the extension of the user's reach by using a nonlinear mapping applied to the user's hand extension. When the user extends the hand farther than a threshold distance *D,* the mapping becomes nonlinear and the virtual arm "grows" (figure 2, right). Different mapping functions can be used to achieve different control-display gain between real and virtual hands (Bowman & Hodges, 1997).

The other common way to select and manipulate objects in VEs is to point at them using a virtual ray emanating from the virtual hand. When the virtual ray intersects an object, it can be picked up and manipulated (figure 3, left). Several variations of ray casting have been developed to help users in selecting very small or faraway objects. For example, the spotlight technique (Liang & Green, 1994) provides a conic selection volume, so that objects falling within the cone can be easily selected. However, when more than one object falls into the spotlight, further disambiguation of the target object is required. The aperture technique (Forsberg, Herndon, & Zeleznik, 1996) uses a conic pointer whose direction is defined by the location of the user's eye (estimated from the head location) and a hand sensor. The user can control the size of the selection volume simply by bringing the hand sensor closer or moving it farther away (figure 3, right). The image plane family of interaction techniques (Pierce et al., 1997) develops and extends this idea.

All the techniques described above provide users with tools that allow them to select or reach further in the immersive virtual world. An alternative approach would be to allow the user to manipulate the relative scale of the virtual world. One of the earliest uses of this approach was in the 3DM immersive modeler (Butterworth, Davidson, Hench, & Olano, 1992), in which users could "grow" or "shrink" themselves to manipulate objects of different sizes. The World-in-Miniature (WIM) technique (Stoakley, Conway, & Pausch, 1995) provides a handheld model of the VE. The user can then indirectly manipulate virtual objects by interacting with their representations in the WIM.

Because all manipulation techniques have particular strengths and weaknesses, a number of attempts have been made to integrate and combine their best features.

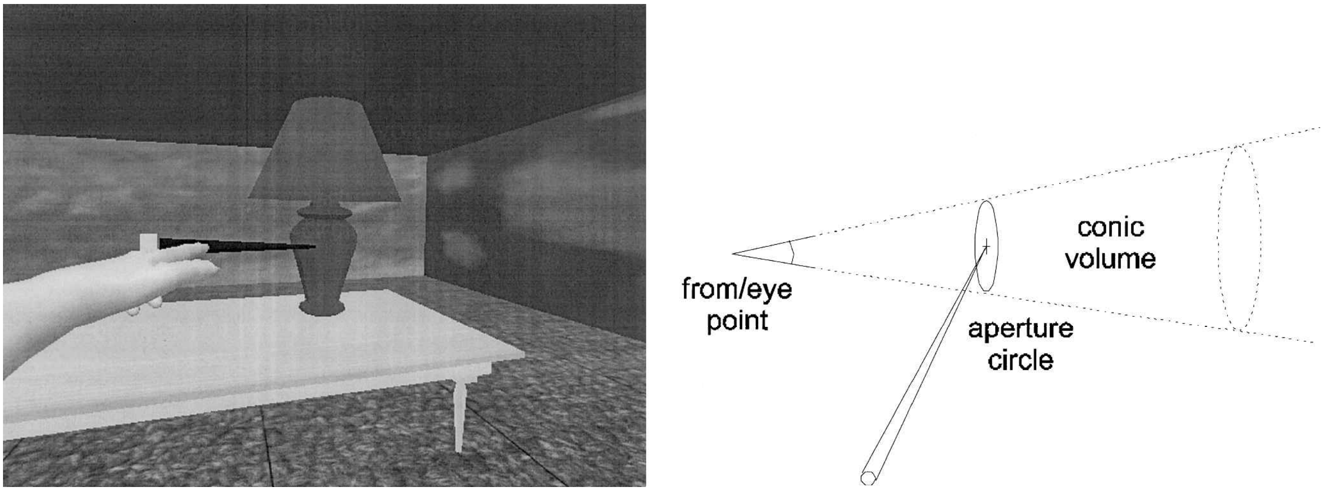

Figure 3. *Ray-casting technique (left) and aperture techniques (right).*

For example, the Virtual Tricorder (Wloka & Greenfield, 1995) combines ray casting for selection and manipulation with techniques for navigation and level-of-detail control within one universal tool. Other examples are the HOMER techniques (Bowman & Hodges, 1997), world scale grab (Mine, Brooks, & Sequin, 1997) and Voodoo Dolls (Pierce, Stearns, & Pausch, 1999).

The variety of reported interaction techniques can be overwhelming for the developer. However, some general principles regarding the choice of manipulation techniques can be stated. None of the techniques can be identified as the "best": their performance is task and environment dependent. Often, nonrealistic techniques have better performance than those based on the real world. Finally, it is important to implement constraints and limit DOFs if possible. Evaluation of virtual manipulation techniques helps to quantify performance and is an important ongoing research area.

3.3 System Control

System control refers to a task in which a command is applied to change either the state of the system or the mode of interaction. The issuing of a command always includes the selection of an element from a set. Thus, some similarities can be seen between system control and object selection techniques.

In desktop applications, the use of commands has received much attention. Unfortunately, interaction styles used in desktop environments, like pulldown menus and command-line input, are not always usable within a VE. One of the basic problems of VE system control is that a normally one- or two-dimensional task becomes three-dimensional, which reduces the effectiveness of traditional techniques. For example, touching a menu item floating in space is much more difficult than selecting a menu item on the desktop, not only because the task has become 3-D, but also because the important constraint of the physical desk on which the mouse rests is missing. Evaluation results for system-control techniques are relatively sparse. Although many ad hoc implementations have been reported, system control has not been studied in a structured way.

We can categorize system-control techniques for immersive VEs into four groups, namely graphical menus (visual representations of commands), voice commands (menus accessed via voice), gestural interaction (command sets accessed via gesture), and tools (virtual objects with an implicit function or mode). Also, hybrid techniques exist that combine several of the types.

System control is often integrated within another universal interaction task. Due to this integration, one should avoid disturbing the flow of action of the main interaction task. The user should stay focused on the task. "Modeless" interaction (in which the mode changes are very natural) is ideal. One way of supporting easy access to a system-control interface is to use a natural spatial reference, such as a fixed position relative to the user's head or body, for placement of the interface. This guideline is mostly applicable to graphical menus, but tools also benefit from a strong spatial reference. Another method to allow a more seamless integration of system control into a flow of action is to use a multimodal system control interface.

After accessing a system-control interface, one has to select a command. When the set of functions is large, one needs to structure the items. This might be achieved by methods like context-sensitive menus, or by clearly communicating the hierarchy of items and (sub)menus.

Finally, the designer should try to prevent mode errors by providing the user with appropriate feedback during and after selection of a command. Mode errors can be highly disruptive to the flow of action in an application.

As in other areas, evaluations are needed which test different system-control techniques within realistic application domains. Further research is needed to determine the performance of these techniques and their applicability to different application situations.

4 Two-Dimensional Interaction in Three-Dimensional Environments

A common misconception of 3-D user interface design is that, because the applications usually contain 3-D worlds in which users can create, select, and manipulate 3-D objects, the interaction design space should utilize only 3-D interaction. In reality, 2-D interaction offers a number of distinct advantages over 3-D interaction techniques for certain tasks. If haptic or tactile displays are not present, 2-D interaction on a physical surface provides a sense of feedback that is especially useful for creating objects, writing, and annotating. As mentioned previously, most efficient selection techniques are essentially 2-D, although further manipulation may require a 3-D interaction technique. By taking advantage of the benefits of both 2-D and 3-D interaction techniques, we can create interfaces for 3-D applications that are easier to use and more intuitive for the user.

Seamless integration of 2-D and 3-D interaction techniques is a critical design consideration from both a physical and logical perspective. Physical integration is important because we do not want to make it difficult for users to switch between 2-D and 3-D devices. Logical integration is also important because we want the devices in the application to know whether they are used for 2-D or 3-D interaction. This contextual application-based information helps to reduce the user's cognitive load.

The 2-D/3-D interfaces can be roughly classified into three categories. Note that, in all categories, some type of physical surface is required for 2-D input. The distinguishing characteristic of these interfaces is how the physical 2-D surfaces are utilized. The first category covers applications that use fully immersive displays such as HMDs, where the user cannot physically see the 2-D surface. Here, the 2-D surface is usually a piece of tracked plastic or pen-based tablet, and users must have a graphical representation of the surface in order to interact with it in the virtual world. Examples of this type are the Virtual Notepad, a system for writing and annotating in VR (Poupyrev, Tomokazu, & Weghorst, 1998) and the Virtual Habitat (Bowman, Wineman, Hodges, & Allison, 1998).

The second category of 2-D/3-D interfaces cover applications that use semi-immersive displays such as workbenches. The physical 2-D interaction surface is usually either on top of the workbench display so users can directly interact with the display surface, or on a tracked, transparent tablet that users can hold in their hand. In the latter case, graphics are projected on the primary display but virtually appear as if they are on the surface of the tablet. Examples of this category are the ErgoDesk system (Forsberg et al., 1998), a modeling application using the physical display surface for a 2-D interface, and the Transparent Pad, a clear tablet that users can hold to interact with a landscaping design application (Schmalsteig, Encarnacao, & Szalzvari, 1999).

The third category uses separate 2-D display surfaces, such as handheld computers and pen-based LCD tablets. An example of this type of interface is the use of a Palm Pilot in a CAVE-like device for camera, environment, and geometry controls (Watsen, Darken, & Capps, 1999). In this area, there are many open research issues to explore, especially when dealing with wireless technology and the weight of these devices.

5 Philosophies of 3-D Interaction Design

The techniques and guidelines presented in this paper come from years of work by many researchers in the field. Often, there is a debate as to the most effective methodology or philosophy for the development of good 3-D user interfaces. In this section, we compare and contrast an "artistic" philosophy of design with a systematic philosophy. In our view, these two strategies complement one another: a flash of insight (art) can lead to good systematic research, and vice versa. Both approaches can lead to novel and effective techniques, and principles and guidelines for design.

5.1 Artistic Philosophy

Shneiderman (1998) identified three "pillars of successful user interface design": guideline documents, user interface software tools, and expert review and usability testing. These pillars specify major sources of guided design decisions, such as the Macintosh Human Interface Guidelines or the interface API provided as part of the Macintosh operating system. Together they outline and implement the very basic elements of the desktop user interface, define their functionality, purpose, appropriate visual appearance and so forth. These and other guidelines and tools provide designers not only with basic interface building blocks, liberating them from the necessity to invent and implement them themselves, but also with a clear vision of how these interface blocks fit together and can be used to design user interfaces for a particular product.

Designing 3-D interfaces has not yet reached the same state of maturity. There are still no cohesive 3-D interface paradigms similar to the WIMP paradigm in desktop user interfaces. Although many 3-D interaction techniques have been reported in the literature and many human-factors issues of multidimensional interaction have been investigated, there is still no common vision on how these research results can be combined together to form a cohesive picture that would guide the interface designer in designing and building spatial interfaces. One of the reasons for this is that the design space in 3-D user interfaces is significantly larger than in 2-D, and large portions of it remain unexplored. Designers have to deal with a large variety of input and output devices and technology that is still rapidly developing: new displays and sensors that require innovation of appropriate interaction techniques and consequent reevaluation of accumulated design experience and knowledge.

How then can developers approach 3-D interface design? The successful 3-D interface design should be based, first, on existing research on human factors in computing systems. Second, it should reuse interaction techniques and ideas developed by researchers. Third, it has to employ creativity and simple approaches that can help to invent interfaces and new interaction techniques. Finally, it must use existing design models and strategies of 3-D interface design.

Examples of basic human-factors principles that can be directly used in 3-D interface design include research on two-handed interaction (for example, Hinckley, Pausch, Profitt, Patten, and Kassell (1997)), constraints, sensory feedback, and multimodal interaction, as well as basic interface-design guidelines that are applicable from 2-D interface design: the requirements of simplicity, consistency, error prevention, and so on.

Techniques for inventing 3-D interfaces can be broadly categorized as those borrowing from the real world and those based on "magic." Examples of realistic techniques include the exact simulation of the physical interaction between humans and environment, borrowing from other areas of human activities (such as movies and architecture), adapting real world tools in the form of 3-D widgets, and borrowing design ideas from 2-D user interfaces. The magic approach involves designing interaction techniques around cultural clichés and metaphors—the flying carpet and magic wand metaphors—as well as conscious violation of user assumptions (Pierce, personal communication, 2000).

Although these techniques allow us to design only short-term, and often ad hoc, solutions, they do provide a foundation for the scientific scrutiny that can analyze, categorize, and evaluate proposed ideas and techniques, giving rise to the systematic approach to interface design as discussed below.

5.2 Systematic Philosophy

The systematic approach to interaction design is characterized by a study of user tasks, existing interaction techniques, and characteristics of the user, environment, or system that might affect performance. In general, this approach is slow and methodical, with incremental improvements in performance rather than sudden leaps.

An important component of the systematic philosophy is classification, usually done in the form of taxonomies. By classifying, categorizing, and decomposing tasks and techniques, we gain a much deeper understanding of them and provide ourselves with a framework for the design of new techniques and the evaluation of existing ones. One type of taxonomy, based on a hierarchic decomposition of a task and then a listing of technique components for the lowest-level subtasks, lends itself to *guided design,* a way to use the taxonomy for the generation of new interaction technique designs. This approach can even be implemented in software so that new interaction techniques can be prototyped and tested extremely quickly.

A second crucial aspect of the systematic approach is evaluation. Although many think of design and evaluation as separate concepts, they are in fact tightly connected. The iterative design process used in standard HCI contexts applies to VEs as well. Evaluation of a design results in changes to that design, which can again be evaluated. In the area of 3-D interaction, basic evaluation of interaction techniques is still important, but formative and summative usability evaluation of VE applications must also be used (Hix et al., 1999).

6 Applications

To illustrate the practical usage of some of the techniques, devices, and concepts described in the earlier sections, we present some application case studies in this section.

6.1 Exploring Complex Data Visualizations with the Cubic Mouse

The use of complex visualization systems is widespread in application domains such as the automobile industry, the medical field, and the oil and gas industry. These visualization systems need to handle massive amounts of data, use appropriate visualization methods, and apply effective interaction techniques to inspect and manipulate data. Traditional input devices often form a bottleneck when interacting with such systems, due to their imprecision and need for regular mode changes. At the German National Research Center for Information Technology (GMD), research in visualization has led to a new input device called the Cubic Mouse (Froehlich & Plate, 2000).

The Cubic Mouse (figure 4) is a cube-shaped device with three rods going through the primary axes. Within the box, a magnetic sensor is placed to track orientation and position. The Cubic Mouse supports highly precise two-handed interaction with complex data sets. Generally, the entire device is used to create an appropriate viewpoint on a scene, while the rods are used to translate and rotate objects within the scene in a constrained way. During several informal and formal evaluations, users showed increased precision in comparison to two gloves or a stylus, and clearly preferred the Cubic Mouse for precise manipulations.

6.2 Multimodal Interfaces in VEs

Multimodal interaction can be defined as the combination of multiple input and output modalities

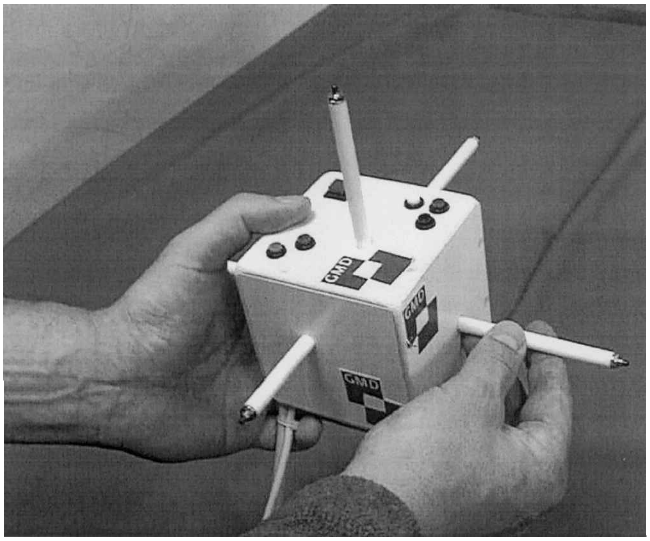

Figure 4. *The Cubic Mouse (copyright B. Froehlich, GMD).*

to provide users with a richer set of interactions compared to traditional unimodal interfaces. Output modalities can be combined to allow for sensory substitution and parallel processing of information. There are several basic ways to combine input modalities (Martin, 1998), including *complementarity* (two or more input modalities complement each other when they combine to issue a single command), *concurrency* (two or more input modalities are concurrent when they issue different commands that overlap in time), and *transfer* (two input modalities transfer information when one receives information from another and uses this information to complete a given task).

These multimodal input styles have been implemented in the context of VR-based scientific visualization applications. In particular, the Multimodal Scientific Visualization Tool (MSVT) allows users to view and interact with fluid flow about a data set by utilizing hand gesture and voice input (LaViola, 2000). MSVT uses input combination styles such as complementarity and transfer for the creation, manipulation, and removal of visualization tools. Users can ask the application for a particular tool while showing it where to place the tool. In addition, users can concurrently manipulate the data set and visualization tools while issuing voice commands for remembering important viewpoints and recording animations. These types of interactions present a more natural interface to the user.

6.3 VEs for Design Education

Architectural design is often touted as an application area for which virtual environments should be ideal. Beyond walkthroughs and validation environments, however, there are few applications of immersive design. The Virtual Habitat (Bowman et al., 1998) is a constrained design space focused on education, rather than a generic design environment aimed at professionals.

The goal of the Virtual Habitat is to allow students to apply the principles of environmental design. Students are immersed in a virtual zoo exhibit and use interactive design tools to make modifications to the terrain, the locations of rocks and trees, and the location and viewing angle of the visitor viewpoints. The task of immersive design is made possible because students do not have to start with a blank environment and because the design tools are constrained by the specific tasks of exhibit design and by the principles of good environmental design.

The Virtual Habitat exhibits complex interaction. Users must navigate, select and manipulate objects, and control the display of information. The system supports different types of tasks and user preferences by providing at least two ways to do most tasks: a direct technique in which the action takes place in the environment, and an indirect technique that uses a 2-D interface displayed on a tablet. The interaction techniques were chosen based on the results of experiments comparing various travel, selection, and manipulation techniques. A usability study has shown this application to perform very well in its intended domain. First-time users of the system all completed novel designs during sessions of one hour or less.

7 Conclusions

Three-dimensional interaction research will only continue to grow in importance as VEs become more complex and widespread. We have given a brief introduction to the field and some of the issues involved in designing effective 3-D user interfaces. For further information, readers should consult the references in the 3DUI bibliography (Poupyrev & Kruijff, 2000) or join (by emailing the authors) the 3DUI mailing list, which is devoted to the discussion of this research area.

The principles of traditional user interface design can often be applied directly to 3-D user interfaces but 3-D interaction has its own unique characteristics that require new design guidelines, some of which have been discussed above. Several generic principles for 3-D user interface design can be stated based on the current state of the art in this area. These include:

- Consider "magic" interfaces in place of "natural" ones when tasks require productivity and efficiency. Natural interaction should be used when a replication of the physical world is important.
- Choose interaction techniques based on the requirements of the application. The same set of techniques will not work well in every situation.
- Limit the required degrees of freedom for input whenever possible, and provide physical or virtual constraints to help guide user input.
- Take advantage of the increased flexibility afforded by whole-body input, multimodal I/O, and novel input devices, but also give the user structure and support in managing the increased complexity.

There are still many unanswered questions and research issues in the field of 3-D interaction. Some of the most important unresolved issues are:

- Will a standard for 3-D interfaces ever be practical?
- How must interaction change between different display devices (for example, HMD and CAVE)?
- How can VEs best support complex system-control tasks (such as very large sets of menu items)?
- How can we determine which 3-D interaction techniques work best for a given application?
- How do various 3-D interaction styles affect the user's sense of presence?

We hope that this overview helps to encourage further work on 3-D interaction techniques, user interface metaphors, and applications. We stress that researchers must continue to perform evaluations of usability and performance, and that further guidelines for the development of effective 3-D interfaces are necessary. Such research has a direct impact on the acceptance of VEs as real-world tools, and on the levels of user satisfaction and productivity in VE applications.

References

Begault, D. (1994). *3D Sound for Virtual Reality and Multimedia*. Academic Press.

Bowman, D., Davis, E., Badre, A., & Hodges, L. (1999). Maintaining spatial orientation during travel in an immersive virtual environment. *Presence: Teleoperators and Virtual Environments, 8*(6), 618–631.

Bowman, D., & Hodges, L. (1997). An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. *Proceedings of the ACM Symposium on Interactive 3D Graphics,* 35–38.

Bowman, D., Johnson, D., & Hodges, L. (1999). Testbed evaluation of VE interaction techniques. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology,* 26–33.

Bowman, D., Wineman, J., Hodges, L., & Allison, D. (1998). Designing animal habitats within an immersive VE. *IEEE Computer Graphics & Applications, 18*(5), 9–13.

Bowman, D. A., Koller, D., & Hodges, L. F. (1997). Travel in immersive virtual environments: An evaluation of viewpoint motion control techniques. *Proceedings of the Virtual Reality Annual International Symposium,* 45–52.

Burdea, G. (1996). *Force and Touch Feedback for Virtual Reality.* Wiley Interscience.

Butterworth, J., Davidson, A., Hench, S., & Olano, M. (1992). 3DM: A three dimensional modeler using a head-mounted display. *Proceedings of the ACM Symposium on Interactive 3D Graphics,* 135–138.

Darken, R., & Cevik, H. (1999). Map usage in virtual environments: Orientation issues. *Proceedings of IEEE Virtual Reality,* 133–140.

Forsberg, A., Herndon, K., & Zeleznik, R. (1996). Aperture based selection for immersive virtual environments. *Proceedings of the ACM Symposium on User Interface Software and Technology,* 95–96.

Forsberg, A., LaViola, J., & Zeleznik, R. (1998). ErgoDesk: A framework for two and three dimensional interaction at the ActiveDesk. *Proceedings of the Second International Immersive Projection Technology Workshop.*

Froehlich, B., & Plate, J. (2000). The Cubic Mouse: A new device for three-dimensional input. *Proceedings of ACM CHI.*

Herndon, K., van Dam, A., & Gleicher, M. (1994). The challenges of 3D interaction. *SIGCHI Bulletin, 26*(4), 36–43.

Hinckley, K., Pausch, R., Profitt, D., Patten, J., & Kassell, N. (1997). Cooperative bimanual action. *Proceedings of CHI: Human Factors in Computing Systems,* 27–34.

Hix, D., Swan, J., Gabbard, J., McGee, M., Durbin, J., & King, T. (1999). User-centered design and evaluation of a real-time battlefield visualization virtual environment. *Proceedings of IEEE Virtual Reality,* 96–103.

Ingram, R., & Benford, S. (1995). Legibility enhancement for information visualization. *Proceedings of IEEE Visualization,* 209–216.

LaViola, J. (1999). Whole-hand and speech input in virtual environments. Unpublished master's thesis, Department of Computer Science, Brown University, CS-99-15.

LaViola, J. (2000). MSVT: A multimodal scientific visualization tool. Proceedings of the Third IASTED International Conference on Computer Graphics and Imaging (forthcoming).

Liang, J., & Green, M. (1994). JDCAD: A highly interactive 3D modeling system. *Computer & Graphics, 4*(18), 499–506.

Martin, J. (1998). TYCOON: Theoretical framework and software tools for multimodal interfaces. In John Lee (Ed.), *Intelligence and Multimodality in Multimedia Interfaces.* AAAI Press.

Mine, M., Brooks, F., & Sequin, C. (1997). Moving objects in space: Exploiting proprioception in virtual environment interaction. *Proceedings of ACM SIGGRAPH,* 19–26.

Pierce, J. (2000). Personal communication.

Pierce, J., Forsberg, A., Conway, M., Hong, S., Zeleznik, R., & Mine, M. (1997). Image plane interaction techniques in 3D immersive environments. *Proceedings of the ACM Symposium on Interactive 3D Graphics,* 39–44.

Pierce, J., Stearns, B., & Pausch, R. (1999). Voodoo Dolls: Seamless interaction at multiple scales in virtual environments. *Proceedings of the ACM Symposium on Interactive 3D Graphics,* 141–146.

Poupyrev, I., Billinghurst, M., Weghorst, S., & Ichikawa, T. (1996). The Go-Go interaction technique: Non-linear mapping for direct manipulation in VR. *Proceedings of the ACM Symposium on User Interface Software and Technology,* 79–80.

Poupyrev, I., & Kruijff, E. (2000). *20th Century 3DUI Bibliography: Annotated Bibliography of 3D User Interfaces of the 20th Century.* Available at: http://www.mic.atr.co.jp/~poup/3dui/3duibib.htm.

Poupyrev, I., Tomokazu, N., & Weghorst, S. (1998). Virtual notepad: Handwriting in immersive VR. *Proceedings of the Virtual Reality Annual International Symposium,* 126–132.

Schmalsteig, D., Encarnacao, L., & Szalzvari, Z. (1999). Using transparent props for interaction with the virtual table. *Proceedings of the ACM Symposium on Interactive 3D Graphics,* 147–154.

Shneiderman, B. (1998). *Designing the User Interface* (3rd ed.). Reading, MA: Addison Wesley Longman.

Stoakley, R., Conway, M., & Pausch, R. (1995). Virtual reality on a WIM: Interactive worlds in miniature. *Proceedings of CHI: Human Factors in Computing Systems,* 265–272.

Thorndyke, P., & Hayes-Roth, B. (1982). Differences in spatial knowledge obtained from maps and navigation. *Cognitive Psychology, 141982,* 560–589.

Watsen, K., Darken, R., & Capps, M. (1999). A handheld computer as an interaction device to a virtual environment. *Proceedings of the Third Immersive Projection Technology Workshop.*

Wloka, M., & Greenfield, E. (1995). The virtual tricorder: A unified interface for virtual reality. *Proceedings of the ACM Symposium on User Interface Software and Technology,* 39–40.

Youngblut, C., Johnson, R., Nash, S., Wienclaw, R., & Will, C. (1996). Review of virtual environment interface technology (IDA Paper P-3186, Log: H96-001239). Institute for Defense Analysis.