3D User Interfaces: New Directions and New Perspectives

Doug A. Bowman[*], Sabine Coquillart[†], Bernd Froehlich[‡], Michitaka Hirose[§], Yoshifumi Kitamura[**], Kiyoshi Kiyokawa[††], Wolfgang Stuerzlinger[‡‡]

ABSTRACT

Three-dimensional user interfaces (3D UIs) allow users to interact with virtual objects, environments, or information using direct 3D input in the physical and/or virtual space. With the advent of systems like the Nintendo Wii, 3D UIs have entered the mainstream. Thus, research on 3D UIs is more relevant than ever. In this article, the founders and organizers of the IEEE Symposium on 3D User Interfaces reflect on the current state-of-the-art in several key aspects of 3D UIs, and speculate on future research directions for 3D UIs as well.

KEYWORDS

3D user interfaces, input devices, bio-signal interfaces, pseudo-haptic, multi-display environments, collaborative 3D user interfaces, virtual reality.

INTRODUCTION

Unless you've been living underground for the last couple of years, you know that the Nintendo Wii has taken the world of gaming by storm. Wii consoles, games, and accessories fly off the shelves faster than they can be restocked, and enterprising resellers make a tidy profit hawking the Wii on eBay. Not only that, the Wii has brought a new demographic to gaming. Its appeal is not limited to males ages 15-30; instead, the games are enjoyed by moms, older adults, and whole families. The unique style of input employed by the Wii, and the types of games this input can be used for, make gaming on the Wii a unique experience.

What makes the Wii special is its three-dimensional user interface (3D UI). Not only does it make use of 3D graphics (like all modern gaming consoles), but it employs innovative spatial input devices that can sense how the user moves them. The gamer can swing his arm through space to roll a bowling ball, point directly at the screen to select a location, and punch the air to win a boxing match.

Although playing with the Wii is the first time that many people have seen or experienced a 3D UI, research in this area has been around for many years. Researchers in fields such as virtual and augmented reality, human-computer interaction, computer graphics, and human factors engineering have all wrestled with difficult questions about the design, evaluation, and application of 3D UIs.

What 3D interaction techniques work best for important tasks like navigation and manipulation? How should we design 3D input devices? What are the most appropriate mappings between 3D input devices, displays, and interaction techniques? How can many 3D techniques be integrated into a seamless 3D UI? All of these questions, and many others, make 3D UIs an exciting area with a wide variety of open issues.

As the example of the Wii demonstrates, 3D UI research is now more relevant than ever. The 3D UI community has been expanding and coalescing as a result – the IEEE Symposium on 3D User Interfaces, first held in 2006, is one piece of evidence for this. In this article, some of the leading experts in the field (the founders and organizers of the 3DUI Symposium) present a series of pieces on both the current state-of-the-art and future prospects for 3D UIs. We hope that this collection of short pieces will serve as an

---

[*] Doug A. Bowman, Center for Human-Computer Interaction, Virginia Tech, USA

[†] Sabine Coquillart, i3D-INRIA Grenoble Rhône-Alpes-LIG, France

[‡] Bernd Froehlich, Bauhaus-Universität Weimar, Germany

[§] Michitaka Hirose, The University of Tokyo, Japan

[**] Yoshifumi Kitamura, Osaka University, Japan

[††] Kiyoshi Kiyokawa, Cybermedia Center, Osaka University, Japan

[‡‡] Wolfgang Stuerzlinger, York University, Toronto, Canada

introduction to the field for those who are new to 3D UIs, and a source of interesting research problems for those who are already in the field.

The first four sections describe some of the *latest 3D UI research trends*. The design of 3D input devices, specifically the use of novel combinations of sensors, is the topic of the first section by Bernd Froehlich of Bauhaus University Weimar. Michitaka Hirose of the University of Tokyo describes the concept of using bio-signals (e.g., brain activity) as an input mechanism for 3D UIs. Providing haptic (touch) feedback in 3D UIs has been a very difficult topic; Sabine Coquillart of INRIA describes research on pseudo-haptic interfaces, in which haptic feedback is simulated by clever use of other sensory displays. In the last piece of this type, the application of 3D UIs to multi-display interfaces is explored by Yoshifumi Kitamura of Osaka University.

The next two sections provide *new perspectives on the design of 3D UIs*. Wolfgang Stuerzlinger of York University proposes eight guidelines for the design of next-generation 3D interaction techniques. Another challenge is the design of multi-user, collaborative 3D UIs; several strategies addressing this problem are presented by Kiyoshi Kiyokawa of Osaka University.

Finally, Doug Bowman of Virginia Tech provides a *new perspective on 3D UI research directions*. He suggests two broad strategies for increasing the impact of this research, both within and beyond the traditional 3D UI domain of virtual reality.

MULTI-SENSORY INPUT FOR IMPROVED CONTROL AND PERFORMANCE

Bernd Froehlich

Figure 1a: The user operates the GlobeFish with the left hand to navigate and manipulate objects with 6 degrees of freedom. The right hand is used for pointing and selection. 1b: The Two-4-Six consists of a Groovepad, operated by the thumb, and an analog rocker joystick operated by the pointer finger. The thumb controls 5 degrees of freedom navigation and the rocker is used for backward and forward motion.

Today three-dimensional interaction in games, CAD or 3D animation applications is mainly performed with the 2D mouse. Everything is mapped to the input from a single pointer and thus users have to learn the transformations of their input actions. The more complex these transformations are, however, the harder the training and performing can be. The current trend towards multi-touch interfaces at least acknowledges that humans tend to act with more than one finger at a time, but still this is just scratching the surface of the immersive experience that virtual environments will offer in future computer applications. What about grasping, turning, pushing, throwing, and jumping when interacting with computer applications? The success of the Wii, Nintendo's current game controller, shows that users want a more engaging computer experience. In particular, professional applications are still far away from providing sufficiently versatile user interfaces.

Task Driven Design of Interaction Techniques

For control and efficiency, the user's focus has to be kept on the task at hand. Graphical widgets, however, require the user's attention for keeping track of their current interaction state. This is not only cumbersome, but also allocates cognitive capacities that would otherwise be available for the task to be solved. To improve the situation we need to design not only software interfaces, but also sensor hardware that fits the specific requirements of spatial interaction. The visual paradigm "What you see is what you get" (WYSIWYG) should become an action-based approach following the idea: "What you do is what happens."

Designing human-computer interfaces in this way requires knowledge of various disciplines, including psychology, software engineering, product design and many others. The challenge is to find the best solution for a certain task instead of developing a workaround to enable the desired functionality within a given infrastructure. As an overview, our research proceeds in five main steps:

- Observing cognitive, perceptual, and motor performance of humans interacting within the physical world
- Modeling the cognitive, perceptual, and motor demands of a certain task in order to create interaction metaphors
- Developing sensors, low-level interfaces and device drivers to record human actions as input for computer applications
- Designing input devices – a combination of sensors assembled in an ergonomic way
- Implementing the designed interaction systems in prototypical applications in order to involve users in the development process
- Examining usability and adjusting the design

Since these aspects are interrelated, the whole design process is iterative. In the following paragraphs, we briefly address these topics and exemplify them with two of our input device designs and their design rationales.

Input Device Design

Each type of input sensor provides specific sensory feedback, depending on its construction and measured input parameters. For example, the counterforce of an elastic joystick perfectly matches the task of controlling velocity during navigation through a virtual environment. Props, such as Hinckley's doll's head [1] resemble a manipulated virtual object and are typically free-moving devices ideally suited for positioning the object in space. Such isotonic input mainly relies on proprioception and tactile feedback. Elastic and isometric input makes use of the human's force sensors (e.g. tendons, ligaments). Most input devices consist of an assembly of different types of input sensors. We use the term multi-sensory input to emphasize that each input sensor possibly relies on different human senses or sensors.

Another major factor in the design of interaction systems is the simultaneously available degrees of freedom (DOF) of the input controller versus the integral attributes of the task it is designed to control [2]. For example, if a task requires movement in all three dimensions, the input device should support these translations along multiple axes simultaneously. If instead only two dimensions are required, as for viewpoint orientation in space, the operational axes of the input device should be constrained to prevent unintentional actions.

Obviously, it would be uneconomical to design an individual controller for every different type of 3D task. For the design of new input devices, we examine various tasks within targeted applications and identify task-specific interactions and interaction sequences. These observations form the basis for our design decisions. A task's parameters define the types of sensor best employed for the manufacture of a relevant control device. The relative frequency of specific tasks and the transitions between tasks provide information about the sensors that should be incorporated in a device and how easy switching among those different sensors should be. Providing good combinations of simultaneously and separately available DOF through an ergonomic arrangement of various sensors still remains a considerable challenge. In addition – as for any physical tool – there are also design qualities like weight, shape and appearance that qualify input devices for certain uses. Our efforts to address these design concerns are illustrated by the following two input devices.

The Globefish

Manipulating objects in 3D is a central task in most digital content creation systems. We observed users performing this task while using an integrated six-degree-of-freedom (6-DOF) input device, the commercially available Spacemouse®. We found that users alternated between rotating and translating and rarely used both operations simultaneously. Thus we decided to build an input device which uses separate sensors for these two interaction modes and allows rapid switching between them. This is the central idea of our Globefish device [3], which consists of a custom 3-DOF trackball embedded in a spring-loaded frame (see Figure 1a). The Globefish trackball sensor measures the rotation of the ball, which is manipulated by the fingertips, and transforms the sensor reading into a corresponding rotation of a virtual object. Firming the grip on the trackball and pushing or pulling it in any direction controls the virtual object's translation along all spatial dimensions. In a user study we compared Globefish to the SpaceMouse for object positioning tasks [3]. For these types of tasks the Globefish clearly outperformed the SpaceMouse and most users clearly preferred the new device.

Motivated by these results, we are currently studying the usability of our device for viewpoint navigation. Since this is a more complex task, it cannot be evaluated as easily as manipulation performance. Navigation in large environments involves motor control and cognitive tasks. The motor behavior, referred to as travel, is the movement of the viewpoint from one location to another. The cognitive process, known as wayfinding, is the specification of a path through an environment. While traveling, wayfinding is mainly supported by regular rotations of the view to scan the environment passing by. For that purpose, the rotational degrees of freedom need to be controlled independently from other input channels. We believe that the Globefish's tangible separation of rotational input from translational input facilitates this environment scanning and thus wayfinding. Travel along a given path may be supported by different interaction metaphors.

The Groovepad and the Two-4-Six

The Groovepad is an input device which consists of a regular touchpad surrounded by an elastically suspended ring with joystick-like functionality. Both input sensors of the device are assembled such that they can be used separately but facilitate frequent and fluid switching between their different input characteristics. In addition, a tracking sensor may be embedded to facilitate pointing and selection tasks.

The Groovepad was originally developed for a hand-held input device, the Two-4-Six, used to navigate in three-dimensional graphics applications [4]. There, the touchpad was used for specifying position-controlled view orientation in a virtual world, while the elastic ring was used for rate-controlled viewpoint motion. Viewpoint rotation around the vertical axis was performed by moving the finger along the elastic ring (see Figure 1b) A separate analog rocker sensor operated with the index finger was used for movement in the depth dimension. We found that this sensor configuration worked well for basic navigation tasks in virtual buildings such as museums or exhibition halls.

Considering 2D desktop applications, we found that the Groovepad matches the required functionality for the increasingly popular zoomable interfaces quite well: The elastically suspended ring can be used as a tangible correspondence to the window frame of the application window. It is used for panning the workspace while the touchpad is used for pointing inside the window. Smooth circular gestures along the Groovepad ring can be used to specify the zoom factor. An initial user study showed that users performed better with the Groovepad than with regular touchpad interfaces. This was particularly the case for tasks that required frequent switches between panning the window and controlling the mouse pointer.

Summary

We have presented some of our ideas and rationales for designing 3D input devices with multiple degrees of freedom and novel combinations and configurations of sensors. Our user studies indicate that these devices perform well for a certain set of tasks and that they can compete with commercially available solutions. To a great extent, however, the design space for desktop as well as handheld devices is still unexplored. Further user studies based on carefully selected tasks and task combinations need to examine the advantages and disadvantages of various sensor combinations to further improve 3D UIs.

BIO-SIGNAL INTERFACE AS A NOVEL 3D UI

Michitaka Hirose

An important goal of VR technology is to allow intuitive interaction between the user and the virtual world. For example, in the early days of VR technology development, a user wearing a Dataglove with a Polhemus sensor could grasp and pick up a virtual 3D object. Now, however, we can imagine a much wider range of interactive channels. In this section, we will describe the use of bio-signals as an input channel for 3D UIs.

The bio-signal channel provides a different way for users to interact with a virtual environment, sometimes without the need for any physical motion. In addition, invisible states of the user, such as “intention”, can be directly measured.

The relatively new research field called brain-computer interface (BCI) is in this category. A variety of brain activities that may be used as an interface channel have been reported, such as visual evoked potential (VEP) [5], p300 evoked potential [6] and motor imagery [7].

SSVEP

A visual evoked potential (VEP) is a measurable signal that arises due to the stimulation of the visual cortex. In particular, steady-state VEP (SSVEP), which can be observed in the visual cortex when viewing a flickering stimulus with a frequency of more than 4Hz, is known to be a reliable signal.

Figure 2a:Virtual Buttons in the Virtual Environment. Left and Right Stimuli Were Presented. 2b: A montage to Collect Steady-State Visual Evoked Potentials. Three-Channel EEG Signals Were Recorded from PO7, PO8.

Our research laboratory has performed several experiments that use SSVEP as a virtual joystick to navigate a 3D virtual environment displayed in the CABIN, which is a virtual-reality room with a five-screen configuration [8][9]. As shown in Figure 2a, virtual buttons to select left and right were positioned in the virtual environment at a distance of 2.0m and a view angle of about 13 degrees. Subjects were asked to gaze at either button; the flickering frequency was set to 8.0Hz for the left-turn button and 6.0 Hz for the right-turn button. We requested the subject to look at the left button if they intended to turn left and vice versa.

A modular EEG cap system was used for measuring the EEG signal on the subject’s scalp. Three-channel EEG signals, PO7, PO8, and Oz (shown in Figure 2b) were used for generating the control signal. The EEG features were extracted from a linear combination of the voltages of the three signals as [VOz – (VPO7 + VPO8)/2]. Using support vector machines (SVMs) with a linear kernel, we were able to classify two states of brain activity – whether the subjects were focusing on the left button or the right button – with a success rate of about 70-80%.

Motor Imagery

If a motor imagery signal is fully recognized and classified, it can be used as an interface that reacts simply to thought. It is known that the mu-rhythm is suppressed by body movement. The mu-rhythm is an EEG component which has 8-12Hz frequency and is typically observed at somatosensory cortices. This suppression is called event-related desynchronization (ERD). Interestingly, ERD occurs even without actual motor movement. Common spatial patterns (CSPs), which are linear spatial filters, can extract the features of signal patterns such as multichannel EEG after a learning period [10].

Figure 3a: Experimental Setup. Visual Cue in Virtual Environment Was Presented. 3b: Three Significant Components of EEG Pattern Map after CSP Filtering during motor imagery tasks (Left and right hand).

Figure 3a shows our experimental setup. The subjects were asked to produce a motor imagery signal (they were asked to imagine tapping their left or right finger) upon the presentation of a cue. Sixteen-channel EEG signals were measured using electrodes installed on a head cap.

To process the EEG signal, after band-pass filtering (8-30 Hz) the CSP algorithm was applied, and a support vector machine (SVM) with a linear kernel was chosen for classification. As the learning data for the CSP algorithm, an EEG signal pattern without visual feedback (the case when motion imagery does not cause actual motion in the virtual environment) was used.

Figure 3b shows the three significant components (1-3) of the EEG pattern map after CSP filtering during phantom finger movements. The EEG pattern was produced by projecting the 16-channel electrode montage onto a rectangular map. As shown in the figure, when the subject images left-finger tapping, the left side of the hemisphere significantly contributes to the spatial patterns, whereas for right-finger tapping, the right side of the hemisphere contributes. This shows that ERD can be extracted successfully from motor imagery. On the basis of this result, SVMs were able to classify left and right commands successfully. A success rate of almost 80% was achieved [11].

Although it is reported that well-trained subjects can generate localized EEG even without CSP filtering, most people are not able to do this. Thus, a CSP filtering process is essential for this purpose.

PSEUDO-HAPTIC INTERFACES

Sabine Coquillart

Figure 4a: Simulation of stiffness[12]. 4b: Simulation of force field [14]. 4c: Haptic Wrists [13][PH9].

3D UIs increasingly require the integration of several input modalities and several types of sensory feedback. Together with visual and auditory sensations, haptics is one of the most important types of sensory feedback. Active force feedback requires the availability of a haptic device able to return forces. However, this type of device is not always available and often difficult to integrate within VR configurations because of the space taken up by the hardware components. In addition, in order to guarantee stiff and stable rendering, active haptic rendering requires both complex computations and a refresh rate of approximately 1KHz for the haptic loop. While active haptic feedback is necessary for some applications, a growing number of studies focus on alternative and lighter approaches such as sensory substitution [PH1], passive haptics [1], or pseudo-haptics. Pseudo-haptic systems can be defined as “systems providing haptic information generated, augmented, or modified by the influence of another sensory modality” [PH2]. This section introduces and discusses some of the most recent contributions regarding pseudo-haptics and presents some applications.

Pseudo-haptic simulation

Recently, a number of authors have attempted to propose light pseudo-haptic solutions based on visuo-haptic illusions. These methods exploit the domination of the visual over the haptic modality. The basic principle consists of perturbing the visual feedback of the hand representation (the cursor for instance), or of the representation of an object handled by the hand in order to induce the haptic sensation that would cause this perturbation. We proposed a simple example in [12], in which a mouse is controlling the displacement of a cube on a horizontal plane. A qualitative experiment showed that a perturbation of the speed of the cube’s displacement could be perceived as friction-like (friction/gravity/viscosity) feedback. Likewise, Lécuyer & al. PH3] and van Mensvoort [PH6] show that a similar setup with a mouse controlling a cursor can be used to produce sensations interpreted as elevations or material properties.

While the subjects experienced a sense of friction, gravity or viscosity, we noticed [12] that forces are more perceptible with a force sensor-based (e.g., Spaceball^{TM}) input device than with a mouse. We can make the hypothesis that the reaction force from the force sensor is more perceptible. This observation led us to propose a second class of pseudo-haptic solutions, based on force sensors. The first studies concerned the simulation of stiffness. We proposed [12] a pseudo-haptic virtual spring based on the coupling of a force sensor and a perturbed visual feedback. The displacement of the virtual spring (D_{virtual}) is deduced from the force applied by the user (F_{user}) and the virtual spring stiffness (K_{virtual}) using the well-known Hooke's law (see Equation 1).

$$F_{\text{user}} = K_{\text{virtual}} \times D_{\text{virtual}} \qquad (1)$$

A major advantage of the force sensor-based pseudo-haptic approach is its relevance to real parameters. A quantitative evaluation of this setup for the task of compliance discrimination between a real spring and a virtual one shows that subjects are able to discriminate successfully with a JND consistent with previous studies on manual compliance discrimination (Figure 4a).

Similar results can be observed with torques. In 2004 [PH4], we extended the concept of pseudo-haptic feedback of stiffness to torque feedback. Torque pseudo-haptic feedback is based on the coupling of visual feedback and the internal resistance of a force/torque sensor that passively reacts to the user's applied force. Results showed that torque pseudo-haptic feedback was successfully simulated, with a difference in performance between isometric and elastic input devices. This difference was also later detected for the simulation of pseudo-haptic stiffness. In another study, Dominjon et al. [PH5] showed that a perturbation of the visual feedback could also be used to modify mass or weight perception.

More recently, we introduced a new deviceless pseudo-haptic concept based on a video see-through HMD augmented reality setup. HEMP – Hand-DisplacEMent-based Pseudo-Haptics aims to simulate haptic sensations by decoupling the visual feedback of the hand from its proprioceptive position [14]. HEMP was applied to the simulation of force fields by slowly moving the hand along the flow. Initial experiments showed a sensation of force perceived by the subjects (Figure 4b).

Applications

Several applications integrating pseudo-haptic feedback have already been developed. Our first applications were based on the mouse setup, while later applications were based on the force/torque sensor setup.

Developing pseudo-haptic applications based on the mouse setup is relatively simple. It can for instance be used for map navigation. A user can locate a landmark roughly while panning a map by making the pseudo-haptic mouse pointer lighter (faster) as the mouse approaches the landmark [PH7]. It can also be used to enhance GUIs. In [PH8] Mandryk et al. created "sticky widgets" to ease the user's access to some interface elements. The mouse slows down in the vicinity of these widgets, creating the illusion of stickiness. User studies compared several levels of stickiness and demonstrated the benefits of the proposed pseudo-haptic approach. Several other examples like games or the simulation of elevations can also be found in [PH6].

As seen above, forces are more perceptible with a force sensor than with a simple position sensor like a mouse. Several promising applications are based on this principle. Crison et al. [15] presented VTT, a pseudo-haptic virtual technical trainer for milling machines. The system uses a SpaceBall™ as a pseudo-haptic interface. The correlation between the user's force on the SpaceBall™ to move the tool and the speed of this move (depending on the resistance of the metal) as seen on the visual display generates the pseudo-haptic perception. While VTT is a fully pseudo-haptic application, it would also be interesting to combine pseudo-haptic solutions with actual haptic systems. CEIT [13][PH9] developed a smart 3-DOF haptic wrist by combining real haptic DOFs and pseudo-haptic DOFs (Figure 4c). The hand-roll DOF along the hand axis is controlled with pseudo-haptics, while the two other DOFs are haptically controlled. Following the results of Paljic et al., the pseudo-haptic DOF is made possible thanks to a force/torque sensor measuring the torque exerted by the user along the handle axis. The associated visual feedback makes it possible to control the pseudo-torque sensation returned to the user. If well integrated, mixing haptics and pseudo-haptics should be a very promising solution to avoid some limitations of haptic systems.

Conclusion

It has been shown that pseudo-haptics can be a viable alternative or complement to real haptics in certain situations. Pseudo-haptics can increase performance and the impression of reality while lowering the cost of haptic systems due to lower requirements regarding actuators and computational load. It is, however, still a young area of research, and a large number of questions remain open.

3D INTERFACES FOR MULTI-DISPLAY ENVIRONMENTS

Yoshifumi Kitamura

Figure 5a-b-c: The perspective cursor and perspective windows in a multi-display environment.

A variety of new display combinations are currently being incorporated into offices and meeting rooms. Examples of such displays are projection screens, wall-sized LCDs, and desktop and notebook PCs. We often use these multiple displays simultaneously during work. Moreover, digital tables are becoming popular in such environments. With the increasing amount of information produced by computers and the decreasing cost of display hardware, multi-display environments (MDEs) are becoming more and more common. We expect to work effectively by using multiple displays in such environments; however, there are important issues that prevent users from effectively taking advantage of all the available displays. MDEs include displays that can be at different locations from and different angles to the user. As a result, it can become very difficult to manage windows, read text, and manipulate objects. Therefore, we feel it is necessary to establish a sophisticated 3D interface for MDEs where the displays are stitched seamlessly and dynamically according to the users' viewpoints, and a user can interact with the many displays as if she is in front of an ordinary desktop GUI environment.

Problems in Multi-Display Environments

Ordinary GUI environments are designed with the assumption that the user sits in front of a stationary display perpendicular to her view; windows and data are rendered according to this assumption. Unfortunately, the perpendicularity assumption does not always hold in recent display environments. When the display plane is not perpendicular to the viewer (e.g., tabletop displays), when the display is flat and covers a large viewing angle (e.g., a large display seen from close proximity), or when the user moves around, the viewing angles become more oblique. The violation of the perpendicularity assumption results in increased difficulty in viewing, reading, and manipulating information due to perspective distortion [16].

The perspective problem becomes more crucial in MDEs than in single-display environments. Misunderstandings caused by perspective distortions may decrease the efficiency of collaboration. Moreover, if information extends to multiple displays, part of the information might not be visible, and will consequently be very difficult to interpret. On the other hand, even in MDEs users expect to use interaction techniques familiar in ordinary GUIs, instead of unfamiliar special ones. Therefore, techniques are necessary to provide MDE users with techniques that extend the ordinary GUI environment. The perspective cursor and perspective windows are two basic techniques that treat multi-displays as if they were part of one large virtual GUI environment.

The Perspective Cursor

Perspective can be defined as "the appearance to the eye of objects in respect to their relative distance and positions." (Merriam-Webster on-line dictionary). The Perspective Cursor is a cursor that moves beyond display boundaries seamlessly as if it is an ordinary desktop environment [17] (Figure 5a). The position and movement of the cursor is calculated from the viewpoint of the user based on the assumption that the system knows the spatial relationships between the user's viewpoint and the visible displays, so that the user perceives the movement of the cursor across displays as continuous, even when the actual movement of the cursor considered in 3D space is not.

The user controls the cursor on a virtual sphere around him, and the spherical nature of the cursor movement mapping makes it possible to point to areas where there is no display. Thus, users might lose the cursor in these spaces. A solution is a perspective variant of halos [18]. Halos are circles centered on the cursor that are big enough in radius to appear, at least partially, in at least one of the screens (a red arc in Figure 5b). By looking at the displayed part of the circle, its position and its curvature, the users can tell how far and in which direction the perspective cursor is located. When the cursor is just off of one display, the displayed arc section of the halo is highly curved, showing most of the circle. If the cursor is very far away, the arc seen will resemble a straight line.

Perspective Windows

The Perspective Window shows perspective-corrected information that the user observes seamlessly as if it is perpendicular to her, even if it is spread over several displays [19] (Figure 5b). Perspective windows display the same kind of content as traditional 2D windows (e.g., a Web browser or a text processor) but offer extra features derived from the perspective-aware capabilities of the system. The main difference between regular windows and perspective windows is that the latter provide optimal visibility to the user regardless of the angle of the display. The windows are rendered using a virtual plane that is perpendicular to the user in the center of the window, and then projected onto the display. If a window is displayed across more than one surface simultaneously, perspective can help reduce fracture. (Figure 5c).

Future of Multi-Display Environments

The Perspective Cursor and Perspective Windows provide a perspective-correct GUI environment for viewing, reading, and manipulating information for each MDE user. Although only the visual displays have been considered with these techniques, it is easy to guess that various types of information presentation devices, such as audio, will also be included in future work. Future MDEs will be required to give the appropriate people the appropriate information by satisfactorily combining the available devices depending on the situation. For this purpose, adequate sensors will also be needed, and it will be important to measure such human dynamics as gestures (including pointing and gaze), physiological information (such as pulse-rate and body temperature), and brain waves. Furthermore, analyses of the conversations between people who share locations and communication processes will provide even more data that can be used to better align the necessary information. A future ideal MDE is expected to identify the situation in the room even if users do not do so explicitly, and provide the necessary information to eligible people in the room. Such a future environment is often called an “ambient information environment.” Toward this end, challenging studies of 3D UIs for MDEs will continue.

GUIDELINES FOR EASY-TO-USE 3D INTERACTION TECHNIQUES

Wolfgang Stuerzlinger

Three-dimensional user interfaces (3D UIs) in Virtual Reality (VR) are still in their infancy. Part of the issue is that 3D hardware technologies are currently still immature and setting up and keeping a VR system running on a daily basis has significant overhead. Another issue is that many user interface techniques are implemented as a thin layer on top of the mathematical foundations. One example is handles that allow only movement in the directions of the three major coordinate axes. The consequence is that only users with an understanding of the underlying mathematics can effectively use such a system. Finally, the biggest problem is that most VR systems require significant training before they can be used productively, which is arguably the biggest barrier to broad acceptance. In contrast, there are many 3D games and on-line virtual worlds that offer easy access to 3D content. Most people adapt quickly to the way such systems afford interaction with 3D worlds. Interestingly, most of these systems use 2D input devices for interaction, which involves the additional overhead of understanding the mapping of 2D mouse movements to 3D motions on the screen.

In the following I list a set of eight guidelines that are derived from user studies with novice participants (i.e. persons without VR knowledge), known results from VR research, perception, kinesiology, (2D) graphical user interfaces, as well as lessons learned from 3D games. These guidelines are designed to aid in driving 3D UIs forward towards broader accessibility and also form a basis for the next generation of 3D UI techniques.

Floating objects are the exception

In the real world, there are few floating objects and almost all objects are attached to other objects. To leverage the experience humans have in the real world, the correct default for any 3D system is to have all objects attach to other objects. In this context it is important to note that all professions performing “full” 3D tasks (astronauts, fighter pilots, divers, …) need substantial training to perform their work. Unfortunately, most current VR (& CAD) systems use a default of having every object float. In good 3D UIs users should be able to make individual objects float. But this should be the exception, not the rule! In other words, standard 3D UI techniques can and should be based on object attachment. Special user interface mechanisms can be provided to make objects stay in mid-air.

Objects do not interpenetrate

Solid objects cannot interpenetrate each other – including the viewer him/herself. Humans are used to this and deal with it every day. However, many VR systems allow object interpenetration by default. The issue is that interpenetration leads to confusing visual display, and many novice users cannot easily recover from such situations. One form of evidence here is the negative effect of users being “trapped” behind a wall in a game – most novices need help to recover from such a situation. Today, real-time collision detection and avoidance is easy to perform for large environments, e.g. with the help of graphics hardware [20]. An added benefit to collision detection and avoidance is that it enables sliding contact, which is an efficient way to position objects in the real world [21].

Interaction only with visible objects

There is strong evidence that users prefer to navigate to manipulate occluded objects [22]. This has several consequences. First, it points at the importance of easy navigation. Second, as the set of all visible objects can be fully described with a 2D manifold, 2D input is then sufficient to select an object. This is also documented by the success of ray-casting-based techniques relative to full 3D selection techniques [23]. Note that this also means that 2D input devices are sufficient to select objects in a 3D world – assuming that adequate 3D navigation techniques exist.

Perspective and Occlusion are the strongest depth cues

For manipulation of objects beyond arm’s length, perspective and occlusion are the strongest depth cues [24]. Assuming that there are no floating objects, these two cues are usually *sufficient* to accurately and quickly judge the 3D position of objects in an environment (unless optical illusions are involved). Although there is a clear value to stereo display, it matters most for objects that are fairly close to the viewer. Hence, stereo display of 3D environments is not always necessary. Last, but not least, there is evidence that most stereo technologies are far from mature and are tiresome and/or problematic if used on a daily basis [25].

People see the object, not the cursor

Research into primate vision has demonstrated that monkeys attend visually not only to the tip of a tool in their hand, but to the whole tool as well as the hand. One consequence of this is that the notion of a “cursor” is not necessarily the best choice for 3D UIs –a cursor is effectively a point, while an object covers an area in the visual field. The SESAME sliding technique [26] analyzes the visual area overlap between the manipulated object and the static scene to determine the position of a moving object. The user studies reported there demonstrate that users can easily use & learn such techniques and such methods provide clear performance benefits.

Full 3D rotations are not always necessary

Many common objects, such as chairs, desks, shelves, etc. have a clear “up” orientation. Other objects, such as hanging lamps, whiteboards, etc. have also clear orientations. Common to all these objects is that they are attached to other objects. However, this attachment also provides constraints for rotation – a chair is only on its side in exceptional cases. Consequently, providing a simple user interface to rotate an object around the axis afforded by that objects’ main attachment is a good design alternative for simple-to-use systems [27]. While there should be support for full 3D rotations, such modes can be secondary and do not need to be easily accessible all the time.

2D tasks are cognitively simpler than 3D

Most tasks in the real world are not fully 3D; they are 2D or 2½D tasks. Consider as an example that buildings consist of layers of 2D floor plans, both because they are easier to build this way as well as easier to navigate. Real 3D structures in buildings exist, but they are again the exception, not the rule. Consequently, most humans are used to dealing with 2D or 2½D and don’t have the training necessary to deal with full 3D problems. For 3D UIs this means that offering 2D alternatives for many tasks is an excellent way to increase usability.

2D input devices are advantageous

Input devices such as the Personal Interaction Panel, which use a pen on a 2D tablet to provide interaction within a Virtual Reality system, have been shown to be effective input devices for 3D worlds [28]. Also, constraining the input to 2D is often beneficial, as it combats hand fatigue and provides more accuracy.

Moreover, a comparison of input device specifications between mouse/pen-based systems and 3D technologies reveals that 2D technologies are one to two orders of magnitude more precise [29]. This work also shows initial evidence that this is one of the main reasons why 2D input devices outperform 3D technologies. Interestingly, the effect of a supporting surface is much less compared to the effect of increased resolution. Consequently, combinations such as using a tablet PC with accurate pen tracking combined with a 3D tracking system for off-slate interaction is a sensible approach.

Conclusion

In summary, I argue that the next generation of 3D UIs can greatly benefit from user interface techniques that are adapted to how humans perceive and interact with the real world. Moreover, novel 3D user interfaces should leverage the strengths of existing technologies (for both input and output) as far as possible and avoid known weaknesses. This will maximize the chances for skill transfer and hence increase the usability of the developed techniques. In turn, this will lead to better 3D applications, a broader range of applications that use 3D productively, and overall increase the adoption of 3D UIs.

DESIGN CONSIDERATIONS FOR COLLABORATIVE 3D USER INTERFACES

Kiyoshi Kiyokawa

Figure 6a: The Bent Pick Ray in use [30]. 6b: Kiyokawa's occlusive HMD in use [34].

This section discusses design considerations for collaborative 3D UIs. A collaborative 3D UI has to accommodate many challenges inherent in its multi-user nature. These challenges include giving correct perspective views to every participant, handling multi-user object manipulation, and supporting natural awareness. To design a successful collaborative system, it is crucial to appreciate recent trends, trade-offs and limitations of interaction techniques and display technologies.

Multi-user Display Systems

One of the fundamental requirements for a collaborative 3D UI is that participants share the same virtual environment while observing it from their own individual perspectives. This simple requirement is not easily satisfied for situations where participants exist at the same real location. A conventional stereo projection system is unable to provide correct perspective views to more than one user. A user’s view is skewed and distorted if he or she observes the scene from a position different from the intended rendering position. In such an environment, different co-located users will perceive the same virtual object as if it is placed at several different locations.

Multi-viewpoint images composed of different image elements from multiple viewpoints partially solve this problem **[COL1**]. This technique projects interaction elements for each user in the correct position based on the tracked real input devices from the user’s point of view. Froehlich et al. have developed a more complete solution by multiplexing the cycle of stereo shuttering [COL2][30].

Bimber et al. [COL3] and Kitamura et al. [COL4] have developed other types of display systems for co-located 3D collaboration. These display systems provide stereoscopic images at the same physical location from different perspectives by separating viewing frustums using mirrors [COL3] or masking plates

[COL4]. One limitation of these types of display is that the effective working volume is fixed and relatively small.

A head mounted display (HMD) is another common display device for co-located collaboration. With a head tracking facility, virtual objects can be placed at arbitrary locations. Typical disadvantages of a HMD include more noticeable system latency, limited field-of-view (FOV) and peripheral vision, and ergonomic discomfort.

Multi-user Object Manipulation

Another fundamental requirement for a collaborative 3D UI is that participants manipulate virtual objects cooperatively. Most collaborative systems support simultaneous manipulation of different objects by different users [COL5]. However, a single virtual object can generally be manipulated by only one user at a time.

For more flexible collaborative activity, some recent studies attempt to allow simultaneous manipulation of a single object by multiple users. Ruddle et al. [COL6] proposed several solutions to combine the movements of two users to obtain the final movement of the virtual object. Riege et al. [30] described the Bent Pick Ray (see Figure 6a). In this technique, a straight line (pick ray) is rendered from the input device to the selected object for single-user manipulation, whereas a bent pick ray is rendered when the second user grabs the same object. The bent pick ray is still emitted from the input device, tangentially to the pointing direction, and touches the selected object at the point of intersection. Duval et al. presented the SkeweR technique [COL7], which allows simultaneous grabbing of any part of a virtual object by multiple users. The grabbed object's translation and rotation is determined by considering the positions of those 'crushing points.' Pinho et al. [31] proposed yet another technique for multi-user interaction – splitting the DOF of the task among the users. Evaluations showed that this technique could be useful when users have two complementary views.

Level of Immersion

The ability to be immersed in a virtual environment is a major asset of virtual reality (VR). The level of immersion significantly affects the participants' sense of co-presence and facilitates mutual understanding of the shared environment. Heldal et al. demonstrated that people working in distributed immersive environments felt more present and were more effective than those in non-immersive settings [32]. The performance results for the best immersive setting were close to the performance results for a face-to-face setting with real objects.

Immersive Projection Technologies (IPT) are commonly used to provide a high level of immersion. However, building an IPT system that supports multi-user observation is costly. In comparison to IPT, a HMD generally provides a lower level of immersion due to limited FOV. Although some closed HMDs provide wide horizontal FOV larger than 180 degrees, the typical horizontal FOV of a see-through HMD remains 60 degrees or less. Kiyokawa has recently developed a projective HMD that could potentially provide over 180 degrees of horizontal FOV as well as see-through capability [COL8]. They are currently developing an immersive co-located collaboration system using this display.

Face-to-face Arrangement

Co-located collaboration with a wall-type display forces users to observe the virtual environment from the same direction. In this case, it is difficult to see both the shared information on the screen and other participants at the same time. The space-multiplexed stereo displays mentioned above support a face-to-face arrangement. A face-to-face setup supports natural communication among participants because they support non-verbal communication cues such as facial expressions, poses, gestures and viewing directions [COL9]. A see-through HMD also supports natural awareness among co-located participants, as well as allowing virtual objects to be placed even between participants in midair. Kiyokawa et al. found that collaboration efficiency is improved by showing the shared information in the space between participants using an optical see-through HMD [COL10]. They also found that by presenting the shared information in midair between participants, their communication became more natural, social, and easier compared with conventional wall-screen and tabletop configurations [33].

Heterogeneous Perspectives

A virtual environment supports free control over users' viewpoints and scaling factors in their individual reference frames. This flexibility is beneficial to collaborative activities too, enabling configurations impossible in the real world, such as shared perspectives [COL11] and multi-scale perspectives [COL12]. In a standard co-located situation where direct visibility of other participants is supported, such flexible perspective settings are not applicable. In this sense, collaborative VR and collaborative augmented reality (AR) are complementary to each other in terms of flexibility and awareness support. For this reason, a number of collaborative 3D systems [COL5][COL13][COL14] support both AR and VR workspaces, and transition between them to maximize user flexibility.

3D collaboration involving a shared real object is difficult to support with independent observation among participants since any motion of the shared object will affect every participant's view. Yamamoto et al. [COL15] partially tackled this problem by using replica objects, and allowed independent observation of the 'same' tangible entity.

Occlusion Handling

True occlusion capability not only enhances visual realism but also improves collaboration experience. For example, to point to a virtual object, a user's finger is expected to occlude the object. On the other hand, to present a virtual object in between multiple users, those users behind the object should be occluded by it. With projection-based systems, it is difficult to handle such occlusion phenomena. With the actual depth information of a real scene, a video see-through HMD is suitable for handling occlusion at the expense of degraded real world visibility. Closer virtual objects are simply overlaid onto the real image, while further virtual objects are left unrendered. Normal optical see-through HMDs are, however, unable to handle occlusion phenomena due to the nature of optical combiners. Kiyokawa's optical see-through HMD tackles this problem and achieves both occlusion handling and natural visibility of co-located participants [34] (see Figure 6b).

Conclusion

According to recent technology advancement and new interaction techniques, the design domain of a collaborative 3D UI is rapidly expanding. There is, however, no single 'right' configuration for all arbitrary conditions. I hope the issues discussed here give some insights into finding an appropriate design for the target application.

3D USER INTERFACES FOR VR AND BEYOND VR

Doug A. Bowman

Figure 7a: Gesture-based menu selection on a remote display. 7b: High-precision 3D selection and manipulation on a large, high-resolution display.

Traditionally, research in 3D user interfaces has been tightly connected to the field of virtual reality (VR). Immersive VR systems such as the CAVE have an inherent need for 3D UIs because they display 3D visual environments, use non-traditional input devices, and require users to stand up and walk around. Even though the definition of 3D interaction – human-computer interaction that takes place in a three-dimensional spatial context [35] – does not mention VR, most 3D UI research has assumed the use of VR technology. Almost all of the common 3D interaction techniques for tasks such as navigation, selection, and manipulation were designed and developed in the context of VR systems [35].

What impact has this research had? Since 3D interaction in VR has been extensively studied, has that resulted in widespread adoption of complex, but usable, 3D UIs for VR applications? Frankly, the impact has been smaller than one might expect. Although researchers have been successful at demonstrating 3D UIs with high levels of both functionality and usability, most real-world VR applications have very simple interfaces. At least for now, the results of years of 3D UI research are not highly visible in the world of VR. In this section, we offer two strategies, or proposed research directions, that could increase the impact of 3D UI research both within and beyond VR.

3D UIs For VR: Demonstrating the Benefits of 3D Interaction

Why don't we see more examples of complex 3D UIs in VR applications? After an initial period of unrealistic hype and expectations in the 1990s, it became clear that immersive VR technology and knowledge were not yet mature enough for most real-world applications. A few applications that depend on a unique user experience and that do not require ultra-realistic visual imagery, such as entertainment, phobia therapy, and vehicle simulation, became highly successful [36]. However, envisioned immersive VR applications with more stringent requirements, such as architectural design, classroom education, and military command-and-control, remained research prototypes in most cases.

But this situation may be changing. As VR technology and our understanding of how to use it have matured, there has been a resurgence of interest in immersive VR. The National Academy of Engineering in the U.S. recently named "Enhance Virtual Reality" as one of the top 14 grand challenges for engineering in the 21st century [37]. Many industries are again exploring the use of immersive VR for productive work, and even consumer applications such as immersive gaming are possible.

In this changing environment, there is again an opportunity for 3D UI research to have a major impact on complex real-world VR applications. But there is a twist: many of the new applications make use of "semi-immersive" technologies such as single large stereoscopic projection screens or tracked handheld displays, rather than traditional VR technologies such as head-mounted displays and CAVEs. In these semi-immersive systems, developers often have a choice between 3D UIs and desktop-style UIs that use 2D input devices and standard 2D widgets. Desktop-style interfaces are attractive because of their broad familiarity and availability.

If 3D UI research is to make an impact on these "new VR" applications, the research community must provide evidence that 3D UIs have significant advantages over desktop-style interaction. This is our first proposed research direction for 3D UIs: *demonstrating the benefits of 3D interaction*. Although we now know a great deal about how to design efficient, usable 3D UIs, we don't know enough about when 3D UIs should be employed. Compared to other types of UIs, what benefits do 3D UIs provide in terms of user productivity, accuracy, or satisfaction?

To answer these questions, obviously, we need empirical studies that compare 3D interaction to other styles of interaction. Such experiments are difficult to design. Simply comparing two existing interfaces does not provide generalizable results, but crafting similar 3D and 2D interfaces from scratch may result in bias toward one or the other. In our prior work on this topic, we have taken a controlled approach, using the same display, environment, and task for each interface, and working to optimize the design of each interface before comparing them. We found that for distant six-degree-of-freedom manipulation tasks, 3D interaction techniques were significantly faster and more accurate than comparable desktop-style techniques [38]. Further studies of this type are needed.

3D UIs Beyond VR: Expanding the Focus of 3D UI Research

As we noted above, the definition of 3D interaction is not limited to interaction with VR. But research on 3D UIs for non-VR systems has been limited. Thus, our second proposed research direction is to *expand the area of focus for 3D UI research* – to actively look for areas where interaction is problematic, and determine whether 3D UIs are a good fit.

The most obvious example comes from the world of gaming. Typical gaming systems use TVs or computer monitors to display 2D or 3D graphics, and handheld controllers, mice, or keyboards for input. But the wildly-successful Nintendo Wii has showed that games are not limited to this type of interaction; indeed, adding spatial interaction can dramatically change and improve the gameplay experience. Clearly, there is an opportunity here for 3D UI research to help determine what forms of 3D interaction will be best suited for use by gamers.

Another opportunity relates to the growing number of large displays found in public places (e.g., airport information displays), in visualization centers (e.g., high-resolution display walls), and in homes (e.g., home theater setups). Often, these displays are passive, simply providing visual information to people near the display. But in some instances, users desire or need to interact with the information on the display. In the airport, for example, a traveler may wish to get more detailed information about the status of his flight. In a visualization center, analysts are required to zoom, pan, rotate, query, cluster, or annotate the visualized datasets. In the home, users need to control all of their multimedia devices. In addition, most of these large displays are *remote*, meaning that the user cannot easily walk up and touch the display. They also need to support users who are walking past the display, sitting on a sofa, or standing in front of the display. These characteristics make desktop-style or touch-screen interfaces impractical.

With spatial input in 3D UIs, users can interact with these large remote displays while standing up and walking around, while distant from the display, and without requiring traditional 2D input devices. With advances in computer vision, users may not require any input device at all, instead interacting directly with freehand gestures in empty space. In our work, for example, we are investigating the design of spatial gestures for menu selection on remote displays [39] (Figure 7a) and for precise object selection and manipulation on high-resolution remote displays (Figure 7b). Note that in all of these examples, immersive VR displays are not being used, and in fact, most of them are displaying only 2D data. We believe this is a promising area for 3D UI researchers to explore.

Summary

The 3D UI research community has a broad base of knowledge about the design and evaluation of 3D interaction, but this has not yet resulted in a high level of real-world impact. We suggest that researchers should empirically demonstrate the benefits of 3D interaction to determine when 3D UIs should be applied, and that the community should expand its focus into new areas such as gaming and remote display interfaces. We believe that these strategies will increase the impact of 3D UI research and make it more relevant to real-world problems.

Acknowledgments

Bernd Froehlich's section is joint work with his colleague Anke Huckauf and his PhD students Alexander Kulik, André Kunert, and Jan Hochstrate. Special thanks go to Alexander Kulik for help with this article.

Sabine Coquillart thanks her PhD students (Anatole Lécuyer, Alexis Paljic, and Andreas Pusch) and her colleagues for their contributions to the work presented in this article.

The guidelines presented by Wolfgang Stuerzlinger are joint work with graduate students Tim Salzman, Graham Smith, Ji-Young Oh, and Rob Teather.

Doug Bowman acknowledges the contributions of his graduate students (Ryan McMahan, Regis Kopper, Mara Silva, Tao Ni, and Yi Wang) and his colleagues in the 3D UI community to the ideas in this article.

REFERENCES

Multi-Sensory Input for Improved Control and Performance

[1] Hinckley, K., Pausch, R., Goble, J.C., and Kassell, N.F., "A Survey of Design Issues in Spatial Input". *In Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 213–222, 1994.

[2] Jacob, R., Sibert, L., McFarlane, D., and Mullen, M. 1994. "Integrality and separability of input devices". *In ACM Transactions on Computer-Human Interaction*, 1(1), 3-26.

[3] Froehlich, B., Hochstrate, J., Skuk, V., Huckauf, A. "The GlobeFish and the GlobeMouse: Two New Six Degree of Freedom Input Devices for Graphics Applications", *In Proceedings of the ACM CHI* 2006, pp.191-199, 2006.

[4] Kulik, A. and Blach, R. and Froehlich, B., "Two-4-Six" - A Handheld Device for 3D-Presentations", *In Proceedings of the IEEE Symposium on 3D User Interfaces*, 2006, pp. 167--170

Bio-Signal Interface as a Novel 3D UI

[5] Middendorf M. et al. "Brain computer interfaces based on the steady state visual evoked response", *IEEE Trans. on Rehabilitation Engineering*, 8 (2), (2008), 211-214

[6] Farewell L.A. and Donchin E. "Taking off the top of your head: toward a mental prothesis utilizing event-related brain potentials", *Electroenceph clin Neurophysiol*, 70, (1988) 510-523

[7] Pfurtscheller G. and Neuper C. "Motor imagery activates primary sensorimotor area in man", *Neurosci Lett*, 239, (1997), 65-68

[8] Touyama H., Aotsuka M. and Hirose M.: "A pilot study on virtual camera control via steady-state VEP in immersing virtual environment", *In IASTED*, (2008), 611-065

[9] Hirose M. et al. "Development and evaluation of the CABIN immersive multiscreen display", *In Systems and Computers in Japan*, 30 (1), (1999), 13-22

[10] Wang Y. et al. "Common spatial pattern method for channel selection in motor imagery based brain-computer interface", *In IEEE Eng Med Biol Soc*, 5. (2005), 5392-5.

[11] Fujisawa J., Touyama H. and Hirose M.: “EEG based navigation of immersing virtual environments using common spatial patterns”, *In IEEE International Conference on Virtual Reality*, 2008, 251-252.

Pseudo-haptic Interfaces

[12] Lécuyer A., Coquillart, S., Kheddar A., RichardO. and Coiffet P., "Pseudo-Haptic Feedback : Can Isometric Input Devices Simulate Force Feedback?", *In IEEE International Conference on Virtual Reality*, New Brunswick, USA, 2000.

[13] Martín, J., Savall, J., Díaz, I., Hernantes, J.,Borro, D. "Haptic Wrists: An Alternative Design Strategy Based on User Perception," (submitted to) *Journal of Computing & Information Science in Engineering (CISE)*.

[14] Pusch A., Martin O. and Coquillart S., "HEMP – Hand Displacement-based Pseudo-Haptics: A Study of a Force Field Application”, *In IEEE Symposium on 3D User Interfaces*, Reno, USA, 2008.

[15] Crison F., Lécuyer A., Mellet-D’Huart D., Burkhardt J.M., Michel G., and Dautin J.L., "Virtual Technical Trainer: Learning How to Use Milling Machines with Multi-Sensory Feedback in Virtual Reality", *Proceedings of IEEE International Conference on Virtual Reality - IEEE VR,* Bonn, Germany, 2005

[PHx] see http://.

3D Interfaces for Multi-Display Environments

[16] Wigdor, D., Shen, C., Forlines, C., Balakrishnan, R.. “Perception of elementary graphical elements in tabletop and multi-surface environments”. Proc. of CHI, pp. 473-482 (2007).

[17] Nacenta, M.A., Sallam, S., Champoux, B., Subramanian, S., Gutwin, C., “Perspective cursor: perspective-based interaction for multi-display environments”. Proc. of CHI, pp. 289-298 (2006).

[18] Baudisch, P., and Rosenholtz, R., Halo: a technique for visualizing off-screen locations. Proc. of CHI, pp. 481-488 (2003).

[19] Nacenta, M., Sakurai, S., Yamaguchi, T., Miki, Y., Itoh, Y., Kitamura, Y., Subramanian, S., and Gutwin, C., “E-conic: a perspective-aware interface for multi-display environments”, *In Proc. of ACM Symposium on User interface Software and Technology (UIST)*, pp. 279-288 (2007).

Guidelines for Easy-to-Use 3D Interaction Techniques

[20] N. Govindaraju, S. Redon, M. Lin, and D. Manocha. CULLIDE: interactive collision detection between complex models in large environments using graphics hardware. SIGGRAPH Workshop on Graphics Hardware, pp. 25-32, 2003.

[21] Kitamura Y., Kishino F. “Consolidated manipulation of virtual and real objects”, *In Virtual Reality Software and Technology 1997*, pp. 133-138.

[22] Ware C., Lowther K.. “Selection Using a One-Eyed Cursor in a Fish Tank VR Environment”. *In ACM Transactions on Computer-Human Interaction*, 4, 4, pp. 309-322, 1997.

[23] Poupyrev I., Weghorst S., Billinghurst M. and Ichikawa T., “Egocentric object manipulation in virtual environments: empirical evaluation of interaction techniques”. *In Eurographics ’98*, pp 41-52.

[24] Wickins C., Hollands J., Spatial displays, in Engineering psychology and human performance, Prentice-Hall 3rd Edition, 1999.

[25] Z. Wartell, L. F. Hodges, W. Ribarsky, A geometric comparison of algorithms for fusion control in stereoscopic HTDs. IEEE TVCG, 8, 129-143, 2002.

[26] Oh J.-Y., Stuerzlinger W. “Moving Objects with 2D Input Devices in CAD Systems and Desktop Virtual Environments*”. In Proc. of Graphics Interface 2005*, pp. 195-202.

[27] Smith G., Salzman T. and Stuerzlinger W.. “3D Scene Manipulation with 2D Devices and Constraints”. *In Graphics Interface ’01*, pp. 135-142.

[28] Szalavári Z., Gervautz M., “The Personal Interaction Panel, A Two-Handed Interface for Augmented Reality”, *Eurographics ’97*, pp. 335-346.

[29] Teather R., Stuerzlinger W., “Assessing the Effects of Orientation and Device on (Constrained) 3D Movement Techniques”, *IEEE Symposium on 3D User Interfaces 2008*, 43-50.

Design Considerations for Collaborative 3D User Interfaces

[30] Riege, K., Wesche, G., Holtkämper, T. and Froehlich, B., “The Bent Pick Ray: An Extended Pointing Technique for Multi-User Interaction,” *Proceedings of the IEEE Symposium on 3D User Interface 2006*, pp.63-66, 2006.
[31] Pinho, M., Bowman, D. and Freitas, C., “Cooperative Object Manipulation in Immersive Virtual Environments: Framework and Techniques,” *Proceedings of the ACM Virtual Reality Software and Technology 2002*, pp.171-178, 2002.
[32] Heldal, I., Schroeder, R., Steed, A., Axelsson, A. S., Spante, M. and Wideström, J., “Immersiveness and Symmetry in Copresent Scenarios,” *Proceedings of the IEEE Virtual Reality 2005*, pp.171-178, 2005.
[33] Kiyokawa, K., Billinghurst, M., Hayes, S. E., Gupta, A., Sannohe, Y. and Kato, H., “Communication Behaviors of Co-located Users in Collaborative AR Interfaces,*” Proceedings of the IEEE International Symposium on Mixed and Augmented Reality 2002*, pp.139-148, 2002.
[34] Kiyokawa, K., Billinghurst, M., Campbell, B., and Woods, E., “An Occlusion-Capable Optical See-through Head Mount Display for Supporting Co-located Collaboration,” *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality 2003*, pp.133-141, 2003.
[COLx] see http://

3D User Interfaces For VR and Beyond VR

[35] Bowman, D., Kruijff, E., LaViola, J., and Poupyrev, I. (2005). *3D User Interfaces: Theory and Practice*, Addison-Wesley, Boston.
[36] Brooks, F. (1999). "What's Real About Virtual Reality?" *IEEE Computer Graphics & Applications*, 19(6), 16-27.
[37] http://www.engineeringchallenges.org/cms/8996/9140.aspx.
[38] McMahan, R., Gorton, D., Gresock, J., McConnell, W., and Bowman, D. (2006). "Separating the Effects of Level of Immersion and 3D Interaction Techniques." *ACM Symposium on Virtual Reality Software and Technology*, Limmasol, Cyprus, 108-111.
[39] Ni, T., McMahan, R., and Bowman, D. (2008). "rapMenu: Remote Menu Selection Using Freehand Gestural Input." *IEEE Symposium on 3D User Interfaces*, Reno, Nevada, 55-58.

BIOGRAPHIES

Doug Bowman is an Associate Professor of Computer Science at Virginia Tech. His research interests include three-dimensional user interfaces and the benefits of immersion in virtual reality systems. He received a B.S. in Mathematics and Computer Science from Emory University and an M.S. and Ph.D. in Computer Science from the Georgia Institute of Technology.

Contact information Doug Bowman:
Dept. of Computer Science
Virginia Tech
2202 Kraft Drive
Blacksburg, Virginia 24060 USA
E-mail: dbowman@vt.edu

Sabine Coquillart is a Research Director at INRIA (French Research Institute in Computer Science and Automatic Control) and LIG (Laboratory of Informatics of Grenoble), France. Her research interests include virtual reality and three-dimensional user interfaces. Sabine received a PhD in Computer Science from Grenoble University and a research supervising Habilitation from the University of Paris XI. She has been a visiting researcher at University of Utah and GMD in Germany.

Contact information Sabine Coquillart :
I3D – Cendre de Recherche INRIA Grenoble Rhône-Alpes
Montbonnot, 655 avenue de l’Europe
38334 Saint Ismier Cedex, France
Tel: +33 4 76 61 52 65 / Fax: +33 4 76 61 54 54
E-mail: Sabine.Coquillart@inria.fr

Bernd Froehlich is currently a full professor with the Media Faculty at Bauhaus-Universität Weimar, Germany. Bernd received his M.S. and Ph.D. degrees in Computer Science from the Technical University of Braunschweig in 1988 and 1992, respectively. His research interests include real-time rendering, visualization, 2D and 3D input devices, 3D interaction techniques, display technology and support for collaboration in co-located and distributed virtual environments.

Contact information Bernd Froehlich:

Virtual Reality Systems Group
Faculty of Media , Bauhaus-Universität Weimar
Bauhausstraße 11
99423 Weimar, Germany
Tel: +49 3643 - 58 3732 / Fax : +49 3643 - 58 3709
E-mail: bernd.froehlich@medien.uni-weimar.de
web : www.uni-weimar.de/medien/vr

Michitaka Hirose is a professor at Graduate School of Information Science and Technology, the University of Tokyo.

His research interests include human interface, interactive computer graphics, wearable computer and virtual reality. He was a project leader of Multimedia Virtual Laboratory project and Scalable VR Contents project sponsored by ministry of general affairs. Currently, he is a project leader of Digital Public Art project and the head of Cyber Interface group of IRT project sponsored by ministry of education.

Prof. Hirose received BE, ME and Ph.D. degree from the University of Tokyo in 1977, 1979 and 1982 respectively. He is a member of the ACM, IEEE and vice president of VRSJ (Virtual Reality Society of Japan).

Contact information Michitaka Hirose:

Graduate School of Information Science and Technology, the University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
E-mail : hirose@cyber.rcast.u-tokyo.ac.jp

Yoshifumi Kitamura received his PhD degree from Osaka University in 1996. After working for Canon Inc. from 1987 to 1992, he participated in virtual space teleconference project at ATR from 1992 to 1996. From 1997, he was an Associate Professor at Graduate School of Engineering, and is currently at Graduate School of Information Science and Technology, Osaka University. His current research interests include user interfaces and display systems.

Contact information Yoshifumi Kitamura:

Graduate School of Information Science and Technology, Osaka University
2-1 Yamadaoka, Suita,
Osaka 565-0871, Japan.
Tel: +81-6-6879-7752 / Fax: +81-6-6879-7743
E-mail: kitamura@ist.osaka-u.ac.jp

Kiyoshi Kiyokawa received his Ph.D. degree from NAIST, Japan in 1998. He worked for CRL, Japan from 1999 to 2002. He was a visiting researcher at HIT Lab of the University of Washington from 2001 to 2002. He is an Associate Professor at Cybermedia Center, Osaka University. His research interests include virtual reality, augmented reality, 3D user interface, and CSCW.

Contact information Kiyoshi Kiyokawa:

Cybermedia Center, Osaka University
Toyonaka Educational Research Center 517
1-32 Machikaneyama, Toyonaka, Osaka 560-0043, Japan
Tel: +81-6-6850-6821 / Fax: +81-6-6850-6829
E-mail: kiyo@ime.cmc.osaka-u.ac.jp

Wolfgang Stuerzlinger is an Associate Professor and a member of the Centre of Vision Research at York University. He got his Doctorate from the Vienna University of Technology. Dr. Stuerzlinger is a leader in research on three-dimensional user interfaces and virtual reality systems, but has also strong research interests in human-computer interaction, display technologies, and collaborative systems.

Contact information Wolfgang Stuerzlinger:
Department of Computer Science and Engineering
Faculty of Science and Engineering , York University, CSEB 1003
4700 Keele Street
Toronto, Ontario, M3J 1P3 , Canada
Tel: +1 (416) 736-2100 ext. 33947 / Fax: +1 (416) 736-5872
E-mail: wolfgang@cse.yorku.ca