3D User Interface Design for Virtual Environments

Doug A. Bowman
Three dimensions and virtual environments intuitively make sense for a wide range of applications, because of the characteristics of the tasks and their match with the characteristics of these environments. Immersion is the feeling of “being there” (replacing the physical environment with the virtual one), which makes sense for applications such as training and simulation. If a user is immersed and can interact using natural skills, then the application can take advantage of the fact that the user already has a great deal of knowledge about the world. The immediacy characteristic refers to the fact that there is a short “distance” between a user’s action and the system’s feedback that shows the result of that action. This can allow users to build up complex mental models of how a simulation works, for example.

Some applications in common use, each of which contain user interaction which is not very complex:
- design verifications, such as architectural walkthrough, and other passive visualizations
- phobia treatment (also very passive for the user)
- entertainment (e.g. point and shoot at enemies)
- training (usually only navigation is required)

Other types of applications (e.g. immersive design, education, complex scientific visualizations) are for the most part still stuck in the research lab, often because they have usability problems that limit their usefulness.

Better technology is not the only answer - 30 years of VE technology research have not ensured that today’s VEs are usable - we must also focus on the design of interaction for VEs. Therefore, we feel that 3D interaction is a vital topic for all 3D/VE developers, designers, and evaluators to understand.
Background: Human-computer interaction

- HCI studies *communication*
- Users and computers communicate via the interface
- Traditional UI design issues:
  - Input device
  - Interaction style
  - Feedback to the user
  - Gulf of execution / gulf of evaluation
- All these are equally relevant for 3D UIs
What makes 3D interaction difficult?

- Spatial input
- Lack of constraints
- Lack of standards
- Lack of tools
- Lack of precision
- Fatigue
- Layout more complex
- Perception
Isn’t the 3D interface obvious?

- Naturalism: make VE work exactly like real world
- Magic: give user new abilities
  - Perceptual
  - Physical
  - Cognitive
Goals of interaction design

- **Performance** (efficiency, accuracy, productivity)
- **Usability** (ease of use, ease of learning, user comfort)
- **Usefulness** (users focus on tasks, interaction helps users meet system goals)

We will try to keep in mind as we discuss ways to accomplish all of these tasks that we want to design for performance, usability, and usefulness.

Performance relates to quantitative measures indicating how well the task is being done by the user and the system in cooperation. This includes standard metrics like efficiency and accuracy.

Usability refers to the ease of communicating the user’s intentions to the system, and the qualitative experience of the user.

Usefulness implies that the system is actually helping the user perform work or meet his/her goals, without being hindered by the interface.

All three of these goals must be considered together, as all are essential. A system will not be used if users become frustrated after five minutes of usage (usability) even if it’s been shown to aid the user in getting work done in a new way. A business will not adopt a system that is incredibly easy to use but decreases productivity (performance).
Outline

- System control
- Symbolic input
- 2D interaction in VEs
- Constraints
- Passive haptic feedback
- Two-handed interaction
System control

- Issuing a *command* to
  - Change the system mode
  - Change the system state
- Often composed of other tasks
- Sometimes seen as a “catch-all” for 3D interaction techniques other than travel, selection, & manipulation
Most of the currently applied system control techniques are derived from a small number of basic metaphors. Influenced by the description of non-conventional control techniques by McMillan et al (in Salvendy, 1997), this categorisation describes four categories under which all commonly used system control techniques should fit. The categorisation is, in many ways, very device oriented. Several metaphors directly depend on a specific input or output device and can not be performed in a similar way with another device. Furthermore, degrees of freedom have also been taken into account, also with respect to input devices. The number of DOFs often coincides nicely with the usage of constraints, as will be shown in the description of several techniques.

The dependence on both devices and DOFs can also be found in the categorisation Lindeman (1999) introduced for manipulation techniques. In this categorisation, many system control techniques can also be found.
The first metaphor which will be described is the group of graphical menus. Graphical menus can be seen as the 3D equivalent of 2D menus. Placement influences the access of the menu (correct placement can give a strong spatial reference for retrieval), and the effects of possible occlusion of the field of attention. The paper by Feiner et al (1993) is an important source for placement issues. The authors divided placement into surround-fixed, world-fixed and display-fixed windows. The subdivision of placement can, however, be made more subtle. World-fixed and surround-fixed windows, the term Feiner et al use to describe menus, can be subdivided into menus which are either freely placed into the world, or connected to an object. Display-fixed windows can be renamed, and made more precise, by referring to their actual reference frame: the body. Body-centered menus, either head-referenced or body-referenced, can supply a strong spatial reference frame. Mine et al (1997) explored body-centered menus and found that the proprioceptive cues which are supplied by the reference frame can significantly enhance menu retrieval and usage. One particularly interesting possible effect of body-centered menus is “eyes-off” usage, in which users can perform system control without having to look at the menu itself. The last reference frame is the group of device-centered menus. These menus should not be mistaken with tools, which will be discussed later. Device-centered placement provides the user with a physical reference frame. A good example is the placement of menus on a responsive workbench, where menus are often placed at the border of the display device. Finally, take note of the effective screen resolution when placing an interface: as Darken (1994) showed, especially the borders of certain displays may cause visibility/readability problems.
Floating menus in VEWL

VIDEO...
Command and control cube (C³)
1 DOF menu

- Correct number of DOFs for the task
- Can be put away
- Only one menu level at a time

1DOF menus are menus which use a circular object on which several items are placed. After initialisation, the user can rotate his/her hand along one axis until the desired item on the circular object falls within a selection basket. The performance is highly depending on hand and wrist physical movement and the primary rotation axis should be carefully chosen. 1DOF menus have been made in several forms, including the ring menu (see picture), sundials, spiral menus (a spiral formed ring menu), and a rotary tool chooser. The second group of hand-oriented menus are hand-held-widgets, in which menus are stored at a body-relative position. Hand-held-widgets function use relative hand positions for selection of items. Please refer to (Liang and Green 1992), (Shaw and Green 1995), and (Mine 1997) for more details.
TULIP menus

- Goal: display all options but retain efficiency and comfort
- Pinch Gloves: Three-up, Labels In Palm (TULIP)
  - only three items available for selection
  - other items appear in sets of three on the palm
  - "more" item linked to next set

VIDEO...
Gestural commands

- Can be “natural”
- Limited vocabulary
- Lack of affordances
- Fuzzy recognition issues
- Gesture as command - doesn’t mimic our use of gestures in the real world
- Pen-based gestures can be powerful
- Perhaps more appropriate in multimodal interfaces
System control design guidelines

- Don’t disturb flow of action
- Use consistent spatial reference
- Allow multimodal input
- Structure available functions
- Prevent mode errors by giving feedback

Extracted from the descriptions of system control techniques, several important design guidelines can be stated. Due to the relative lack of formal evaluations, these guidelines are primarily based on tendencies described by researchers and personal experience.

System control is often integrated within another universal interaction task. Due to this integration, we should avoid disturbing the flow of action of an interaction task. The user should stay focused on the task. “Modeless” interaction (where the mode changes are very natural) is ideal. One way of supporting the user to easily access a system control interface is by using a correct spatial reference. This guideline is of course 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, or hybrid, system control interface. Multimodal interfaces can increase the performance of issuing a command, and may allow multiple channels to access the system control interface. However, keep in mind that multimodal system control is not always suitable or applicable.

After the user has accessed a system control interface, he/she has to select an item from a set: when this set is large, i.e. when a large number of functions are available, one needs to structure the items. As stated in the guidelines on graphical menus, this might be achieved by methods like using context-sensitivity, or by clearly communicating the hierarchy of items and (sub)menus.

Finally, always try to prevent mode errors by providing the user with appropriate feedback during and after selection of a command. Mode errors can be highly disturbing and they interrupt the flow of action in an application.
Symbolic input

- Communication of symbols (text, numbers, and other symbols/marks) to the system
- Is this an important task for 3D UIs?
- Why is symbolic input in 3D UIs different from symbolic input elsewhere?
Symbolic input techniques

- Keyboard-based
  - Miniature keyboards
  - Low key-count keyboards
  - Chord keyboards
  - Pinch Keyboard
  - Soft keyboards
- Pen-based
  - Pen-stroke gesture recognition
  - Digital Ink
Symbolic input techniques 2

● Gesture-based
  ○ Sign language gestures
  ○ Numeric gestures
  ○ Instantaneous gestures

● Speech-based
  ○ Single-character speech recognition
  ○ Whole-word speech recognition
  ○ Unrecognized speech input
Pinch keyboard video
Symbolic input scenarios

Which technique(s) would you choose when...

- … architects will be making annotations to various parts of a design?
- … engineers need to enter file names to save their work?
- … an AR user needs to add hundreds of labels to an environment?
2D interaction in 3D VEs

- Quite useful for appropriate tasks (match task and input DOFs)
- Can integrate seamlessly with 3D
- If presence is important, the 2D interface should be *embedded*, not *overlaid*
- Examples:
  - Interaction on the projection surface or viewplane
  - Using a PDA for VE input
Constraints

- Artificial limitations designed to help users interact more precisely or efficiently
- Examples:
  - Snap-to grid
  - Intelligent objects
  - Single DOF controls
Passive haptic feedback

- Props or “near-field” haptics
- Examples:
  - Flight simulator controls
  - Pirates’ steering wheel, cannons
  - Elevator railing
- Increase presence, improve interaction
Two-handed interaction

- Symmetric vs. Asymmetric
- Dominant vs. non-dominant hand
- Guiard’s principles
  - ND hand provides frame of reference
  - ND hand used for coarse tasks, D hand for fine-grained tasks
  - Manipulation initiated by ND hand
Two-handed interaction examples
Pen & tablet interaction

- Involves 2D interaction, two-handed interaction, constraints, and props
Pen & tablet interaction

- Can put away
- Constrained surface for input
- Combine 2D/3D interaction

- Handwriting input?
- Use any type of 2D interface, not just menus
Virtual habitat video
Conclusions

- Usability one of the most crucial issues facing VE applications
- Implementation details critical to ensure usability
- Ease of coding not equal to ease of use
- Simply adapting 2D interfaces is not sufficient
More work needed on…

- System control performance
- Symbolic input and markup
- Mapping interaction techniques to devices
- Integrating interaction techniques into complete UIs
- Development tools for 3D UIs
Resources

- International 3DUI group
  - Mailing list
  - Annotated bibliography
  - www.3dui.org
- 3DI research at VT
  - research.cs.vt.edu/3di/
Acknowledgments

- Joe LaViola, University of Central Florida
- Ernst Kruijff, Graz Univ. of Technology
- Ivan Poupyrev, Sony CSL