The Props-Based Interface for Neurosurgical Visualization

Ken Hinckley, Randy Pausch, J. Hunter Downs,
Dennis Proffitt, and Neal F. Kassell

Neurovisualization Lab
University of Virginia, Department of Neurosurgery
Charlottesville, Virginia 22903

We describe a three-dimensional human-computer interface for neurosurgical visualization. The interface is based on the two-handed physical manipulation of hand-held tools, or "props", in free space. These user interface "props" facilitate transfer of the user's skills for manipulating tools with two hands to the operation of a user interface for visualizing 3D medical images, without need for training.

The interface allows neurosurgeons to explore a 3D MRI scan of a patient's brain during presurgical planning. From the surgeon's perspective, the interface is analogous to holding a miniature head in one hand which can be "sliced open" or "pointed to" using a cross-sectioning plane or a stylus tool, respectively, held in the other hand. Cross-sectioning a 3D volume, for example, simply requires the surgeon to hold a plastic plate (held in the preferred hand) up to the miniature head (held in the nonpreferred hand) to demonstrate the desired cross-section.

1. Introduction

In efforts to develop interaction techniques for virtual environments which are extremely flexible and versatile, manipulation in virtual reality has focused heavily on visual feedback techniques (such as highlighting objects when the selection cursor passes through them) and generic input devices (such as the glove). Such virtual manipulations lack many qualities of physical manipulation of objects in the real world which users might expect or which users might unconsciously depend upon. For example, in the case of selecting a virtual object using a glove, the user must visually attend to the object (watch for it to become highlighted) before selecting it. But what if the user's attention is needed elsewhere? We believe that designers of virtual environments can take better advantage of human motor, proprioceptive, and haptic capabilities without necessarily sacrificing versatility. In support of this statement, we present our experiences with the two-handed props interface for neurosurgical visualization.

If technological advances are to be well accepted by physicians, we will need corresponding advances in input devices and user interaction. The props-based interface achieves a high degree of usability by employing human motor, proprioceptive, and haptic capabilities to its advantage. The system has been developed and tested in the context of real surgeons doing real work, and provides a compelling demonstration that 3D interaction techniques based on hand-relative-to-hand manipulation of physical objects can allow users to focus attention on their tasks without becoming distracted by the interfacing technology.

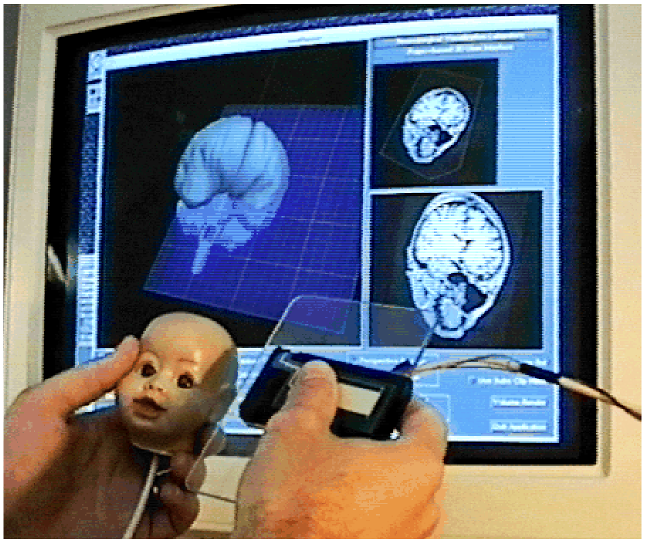

Figure 1: User selecting a cutting-plane with the props.

2. Neurosurgical Planning and Visualization

Neurosurgeons have recently become increasingly interested in computer-based surgical planning systems which allow them to quantify and visualize the three-dimensional information available from medical imaging studies. By making better use of this three-dimensional information, and by allowing the surgeon to quickly and intuitively access it, computer-based visualization and planning systems can positively impact both cost-of-care and patient outcome. Neurosurgery is inherently a three-dimensional activity; it deals with complex structures in the brain and spine which overlap and interact in complicated ways. To formulate the most effective surgical plan, the neurosurgeon must be able to visualize these structures and understand the consequences of a proposed surgical intervention, both to the intended surgical targets and to surrounding, viable tissues.

The user interface discussed in this paper focuses on the pre-surgical planning phase, which usually takes place on the morning of surgery. To develop a plan, the surgeon uses visualization, measurement, and other planning tools to select the surgical targets and to select a path to those targets that produces the least possible damage to viable tissue. To plan the best possible trajectory, the surgeon needs to understand the spatial relationships between the targets, surrounding structures, functional areas which must not be damaged, and feasible entry points. Visualizations of the cortical surface, proposed surgical trajectories, and volume cross-sections at both orthogonal and oblique angles can all help the neurosurgeon to make informed decisions.

Software usability is crucial to get neurosurgeons to actually use advanced visualization software in the clinical routine. We have designed interaction techniques which facilitate use of the software by surgeons, without need for technical assistance. Rather than typing in commands or moving sliders with a mouse, the neurosurgeon thinks in terms of real objects in real space; a three-dimensional user interface should allow the neurosurgeon to work and think in these same terms. As one surgeon put it, “I want a skull I can hold in my hand.”

The user interface for a neurosurgical planning and visualization system must permit the surgeon to work quickly. The surgeon must cope with frequent distractions, and therefore must be able to quickly detach from the user interface, both physically and cognitively. Thus, the interface must not encumber the surgeon with devices such as gloves or head-mounted displays that will be difficult to remove, and it must not have explicit modes that are easily forgotten during a phone call or a discussion with a colleague.

Our laboratory has worked closely with the neurosurgeons at our University throughout the design process for our system. Our work has necessarily been heavily collaborative, relying on the advice and opinions of neurosurgeons to provide goals and specifications throughout the design process.

3. System Overview

Without the ability to render and manipulate images of the brain in real time, our approach to the interface would be infeasible. The system software has been designed to achieve high performance: typical interactive update rates are approximately 15-18 frames per second. During each frame, the system renders a simplified brain surface representation consisting of approximately 9,000 polygons and displays a volumetric cross-section from data which typically consists of 256 x 256 x 128 voxels. Detailed volume renderings (which require about 5 seconds to render) can also be generated once a view is selected. The software runs on a Hewlett Packard J210 workstation with the Visualize hardware polygonal and texture mapping acceleration. We use a six degree-of-freedom magnetic tracking system [12] to track the position and orientation of the input devices.

3.1 Input Devices

The surgeon uses a *head prop* (the doll's head seen in fig. 1) to manipulate the individual patient's MRI (Magnetic Resonance Imaging) head data. The prop is a small doll's head which can be held comfortably in one hand. The head prop is an absolute rotation controller: rotating the doll's head always causes a polygonal model of the patient's brain to rotate correspondingly on the screen. The user can control the image zoom factor by moving the prop towards or away from his or her body. Note, however, that the software does not in fact know where the user is sitting, so the zoom factor is actually based on the distance between the doll's head and the front of the screen.

The doll's head is used as a four degree-of-freedom controller: three degrees-of-freedom for rotation plus one degree-of-freedom for the zoom factor. In the context of surgical visualization, moving the object left-right or up-down is typically not useful, so it is helpful to constrain the polygonal brain to appear at the center of the screen. This simplifies the task and users find it natural.

Also note that the scale of the doll's head does not match the scale of the actual patient data. The doll's head acts only as an orientational reference and its outer surface conveys no information about scale: touching the outer surface of the doll's head does not correspond to touching the outer surface of the virtual patient data. Scaling down the virtual patient data to roughly match the size of the doll's head would result in a substantial loss of precision: one millimeter of real-world motion would then correspond to several millimeters of virtual motion.

3.1.1 Slicing the patient data with a cutting-plane prop

The surgeon can also employ a *cutting-plane prop,* to specify the position and orientation of an arbitrary slice through the patient's anatomy (*fig. 2, left*). Users can spread their fingers across the plate to get a direct haptic sense of how it is oriented in space. The appearance of the cutting-plane prop differentiates it from the head prop and makes its purpose immediately obvious.

Note that the cutting-plane prop is used in concert with the head prop rather than as a separate tool. The user holds the cutting-plane against the head to indicate a slice through the brain data. The computer shows a corresponding virtual tool intersecting the virtual head, along with a cross-section of the volumetric head data (*fig. 2, right*). The user can interactively sweep the cutting plane through the volume to quickly develop a sense of the objects embedded in the volume. Structures which are difficult to visualize when viewing orthogonal slices can now be easily found and inspected.

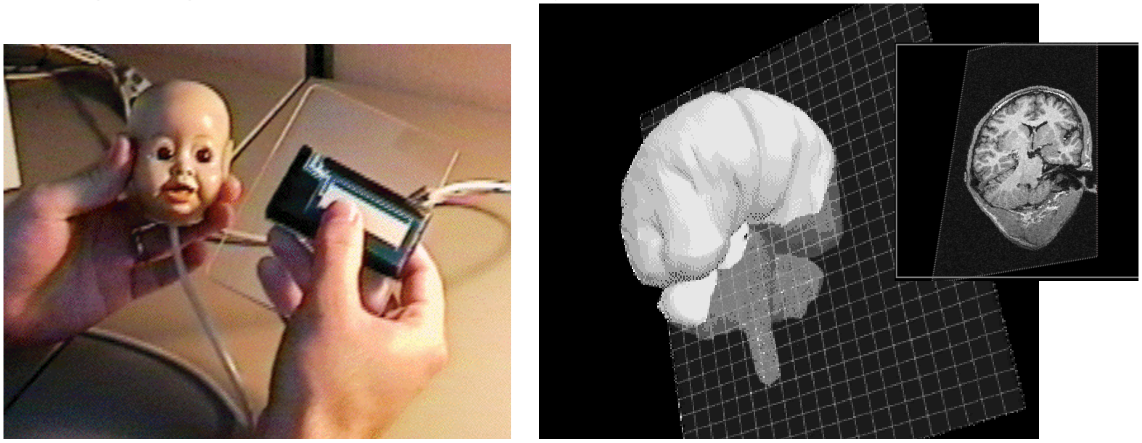

Figure 2: User indicating a cross-section.

To provide visual correspondence, the virtual representation of the cutting-plane prop mirrors all six degrees-of-freedom of the physical tool. But several of these degrees-of-freedom do not affect the cross-section of the object, because (mathematically) the resulting plane has only four degrees of freedom. Rotation about the axis normal to the cutting-plane does not affect the cross section, nor does left-to-right or front-to-back motion in the current plane. In this regard, the cutting-plane prop acts a bit like a flashlight: the virtual plane is much larger than the physical cutting-plane prop, so when one holds the input device to the side of the doll's head, on the screen the plane still virtually intersects the brain, even though the two input devices don't physically intersect.

3.1.2 Indicating surgical paths with a trajectory prop

The *trajectory selection prop* is a stylus-shaped tool (*fig. 3*) that allows the surgeon to specify 3D vectors and points. Moving the trajectory prop relative to the head prop specifies the position and orientation of a cylindrical virtual probe relative to the polygonal brain model. In previous work, Chung [3] implemented an interface for a similar task (radiotherapy treatment planning) using a head-mounted display, but Chung found that using a head-mounted display to select the trajectory of the radiotherapy beam did not have any task performance advantages over hand-guided rotation.

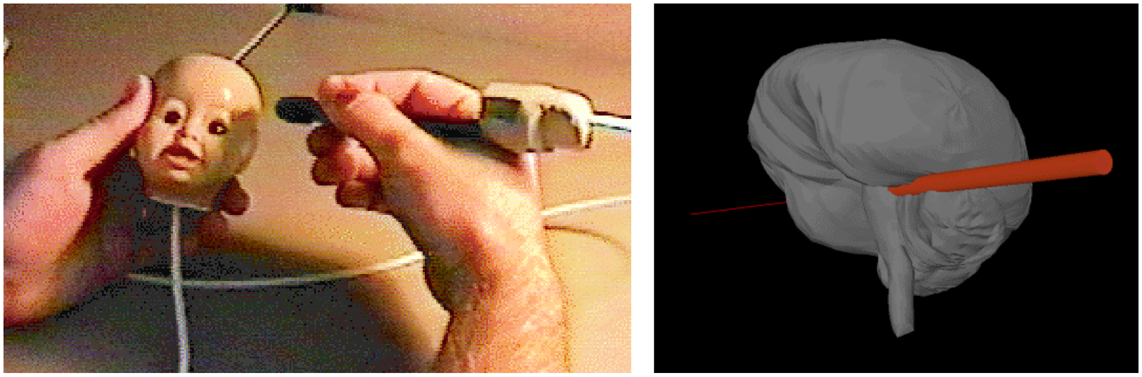

Figure 3: User selecting a trajectory.

A trajectory consists of a target point inside the brain and a vector to that point. The trajectory prop indicates the vector by its orientation relative to the head prop. The target of the trajectory is indicated by the intersection of a ray cast from the virtual probe and the brain model's surface. Points which lie on the interior of the brain model can be selected by first bisecting the volume with the cutting plane to expose the contents of the volume, and then selecting a point on the exposed surface.

4. Two-handed Interaction

Guiard [7] has proposed that humans use the preferred and nonpreferred hands to control frames of reference which are organized in a hierarchy. For right handers, the left hand specifies a *base frame of reference* relative to which the right hand expresses a second *active frame of reference*. The props interface assigns the base frame of reference to the doll's head and the active frame of reference to the cutting plane. Since the neurosurgeon's task is to specify a cutting plane relative to a particular desired view of the brain, the interface's frames-of-reference assignment matches the surgeon's mental model of the task, resulting in an easily understood two-handed interface. Understanding the structure of bimanual manipulation (a hierarchy with the preferred hand moving relative to nonpreferred hand) is essential to define appropriate mappings for the input devices [10].

During the early stages of the interface design, we felt some concern that users might not be able to effectively control the four degrees-of-freedom provided by the doll's head using only their “weak” hand. In practice, however, informal evaluations have confirmed that the non-dominant hand is well suited to this task. The nonpreferred hand is not merely a poor approximation of the preferred hand, but can bring skilled manipulative capabilities to a task [12], especially when it acts in concert with the preferred hand.

In related experimental work, we have demonstrated that using two hands can provide more than just a time savings over one-handed manipulation [10]. Two hands together provide the user with information which one hand alone cannot. Using two hands can impact performance at the cognitive level by changing how users think about a task: using the props with both hands helps users to reason about their tasks.

5. Interactive volume cross-sectioning

In the original concept of the interface, the surgeons wanted to interactively clip away a portion of the object and paste the volumetric cross-section data on to the resulting capping polygon, so that the cross-section would always be seen directly in the context of the polygonal brain model. Without texture mapping hardware, the implementations we attempted could only render the volume slice in context at a maximum of about 2-3 frames per second, compared to about 15-18 frames per second for a display in a separate window. Thus we pursued a separate, out-of-context display.

The separate display can take the slice data that would have been superimposed on the polygonal model and draw it in a separate window, giving a *perspective view* of the slice data, or alternatively a *map view* of the slice from directly above could be used. Figure 4 (*see next page*) compares these approaches by showing how the separate cross-section display behaves for the perspective view and map view techniques as the user changes the view of the polygonal brain over time.

As seen in the figure, when the user tilts or moves the plane, the perspective view changes accordingly. Thus, if the user is primarily visually attending to the cross-section display (and not to the 3D view of the polygonal objects), the perspective view technique provides useful visual feedback of the motion of the plane. However, if the user holds the plane so that it is seen roughly edge-on in the 3D view of the polygonal objects, the perspective view conveys essentially no information. This imposes certain restrictions on how one can hold the cutting plane prop: it has to be held roughly vertical so that the cross-section data can be seen during interactive manipulation. Some users initially find this requirement to be confusing, though it is easy to adapt to.

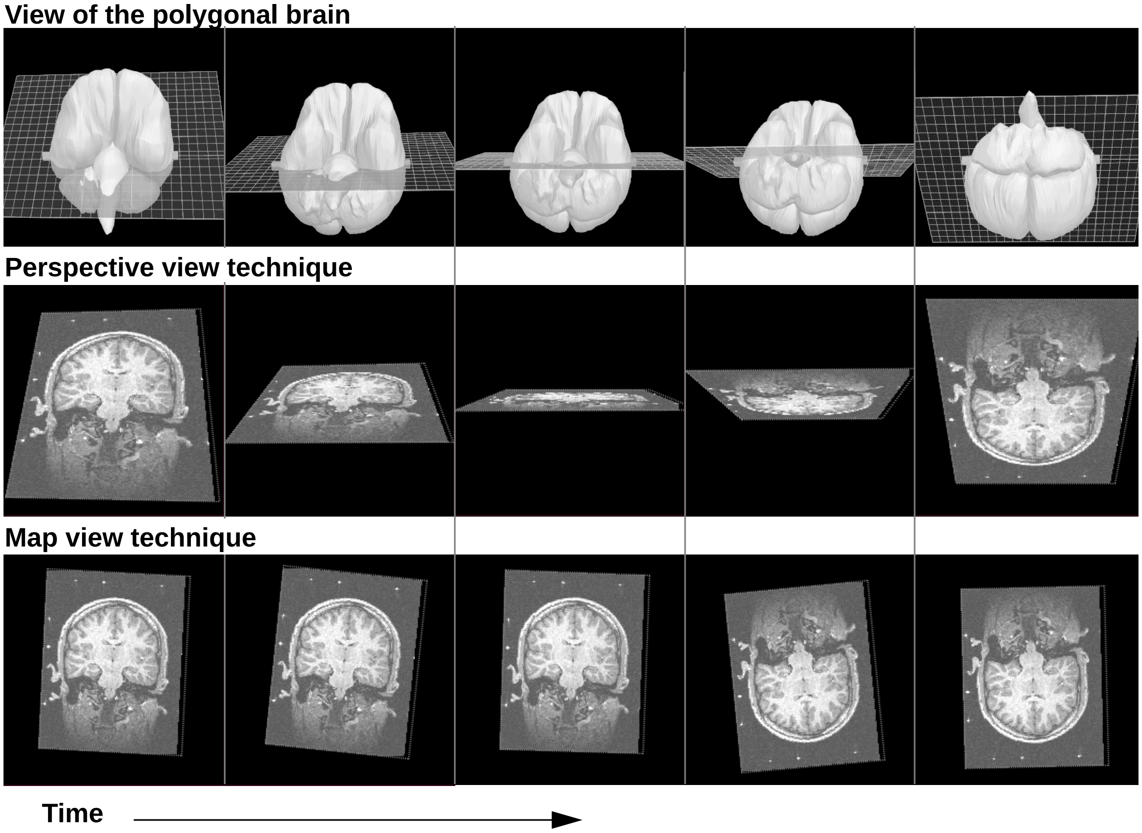

Figure 4: Comparison of perspective and map views of the cross-section data.

The map view technique (*fig. 4, bottom row*) does not impose any restrictions on the orientation of the cutting plane prop relative to the view-- even when the user holds the plane edge-on, he or she can still see the data resulting from the cross-section. But the map view technique lacks the motion cues provided by the perspective view; as seen in figure 4, the map view essentially does not change at all over the first three frames of the time sequence, despite a large angle of rotation. Thus, each technique can be considered to have its own strengths and weaknesses. For the dynamic view of the cross-section, the current version of the props interface defaults to the perspective view and has the map view available as an option. When the user selects a plane (by releasing a clutch button on the cutting-plane prop), a separate static view of the cross-section is saved for subsequent 2D manipulation, and this static view is displayed using the map view technique.

5.1 Texture mapping hardware

Texture mapping hardware allows for the possibility of in-context viewing of the cross-section data. Standard texture mapping hardware does not help to calculate the cross section itself; this still must be calculated in software to compute the initial texture map. Unfortunately, selecting an arbitrary plane through a volume is probably a worst case for traditional texture mapping. Loading a texture map to texture memory is an expensive operation. Texture mapping hardware typically assumes that most textures are known when an application begins, and that they will be used repeatedly, so the system performs pre-computations and creates data structures when a texture is first loaded. But when users move the cutting plane prop to interactively select a volume cross-section, the texture map changes every frame and the exact same plane is not usually accessed more than once.

Thus, commonly available texture mapping hardware still does not allow an in-context presentation of the cross-section while the cross-section itself is changing (some high-end SGI machines do support “voxel maps” which allow volume cross-sectioning in hardware). However, once the user selects a cross-section, this cross-section can be turned into a traditional texture map and integrated with the view of the polygonal objects. With the cross-section available in context, users find it much easier to select a subsequent plane which passes through a target revealed by a previous cut.

6. Touchscreen Interface

The 3D interface props excel for 3D manipulation, but when 2D tasks such as panning and zooming an image arise, there is an awkward pause in the human-computer dialog while the user must put down the props to move the mouse or to use the keyboard. The problem lies in a 3D input versus 2D input dichotomy: some tasks are best done in 3D, others are better suited to 2D, and users need an intuitive and consistent mechanism for switching between the different styles of input. Users are distracted from the focus of their work because they must decide which device to acquire for a given input task.

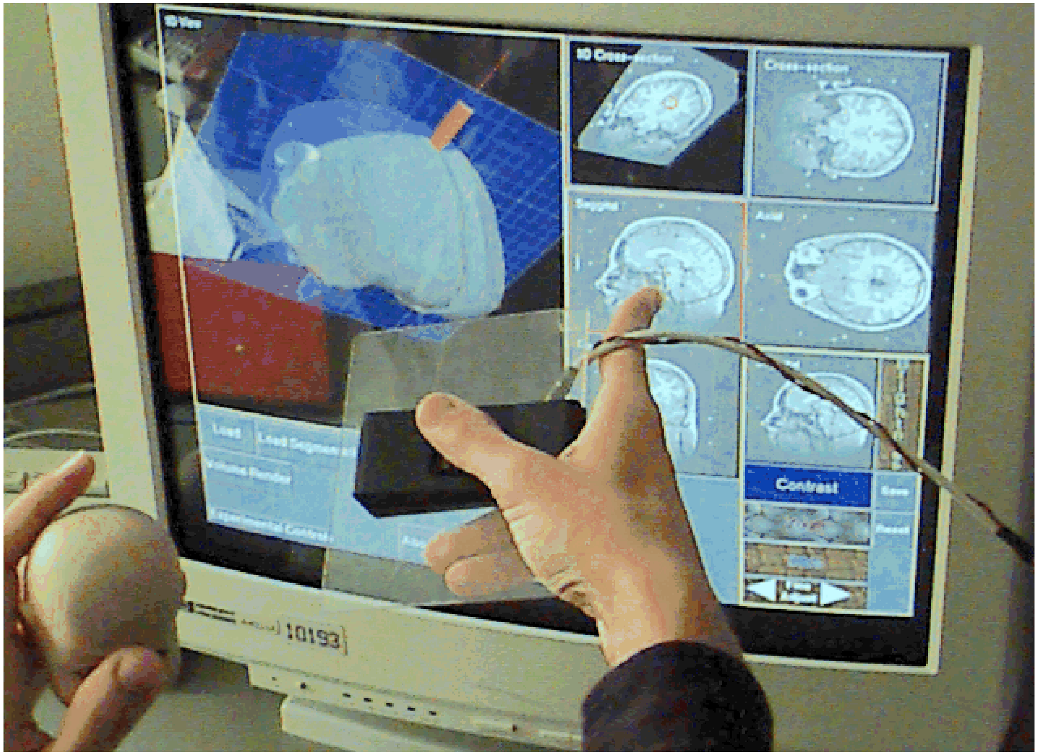

Figure 5: User employing the touchscreen in combination with the props.

To address this shortcoming, we added a touchscreen sensor to the monitor used with the interface props. This hybrid interface combines 3D input with more traditional 2D input in the same user interface. Note the ergonomic facility with which a touchscreen can be used: the surgeon can move in 3D using the props; then, without having to put the props down, the surgeon can reach out and touch the screen to perform 2D tasks, since the hand is sufficiently free to extend a finger or knuckle (*fig 6*). This provides a consistent input medium for both 2D and 3D tasks, since the user always interacts gesturally with objects in the real environment: one interacts gesturally with the props to perform 3D operations; one interacts gesturally with the touchscreen to perform 2D operations.

The touchscreen graphical user interface (GUI) divides the screen into a set of tiles (*fig 7*) which contain different views of the same volumetric data set. These tiles are interchangeable; for example, to increase the screen real estate for the sagittal view, the user can drag it with his finger into the large area on left side of the screen. The region in the lower right hand corner of the screen acts as a general purpose control panel for all tiles. When the user touches a tile, it becomes selected and a miniature copy of the tile appears in the control panel. The control panel widgets can then be used to interactively manipulate the miniature copy, and after a brief pause, the changes are propagated to the original tile. The control panel includes controls for image contrast and brightness, zooming and panning, browsing or precisely stepping through parallel slices, saving the current image, and resetting the default viewing parameters.

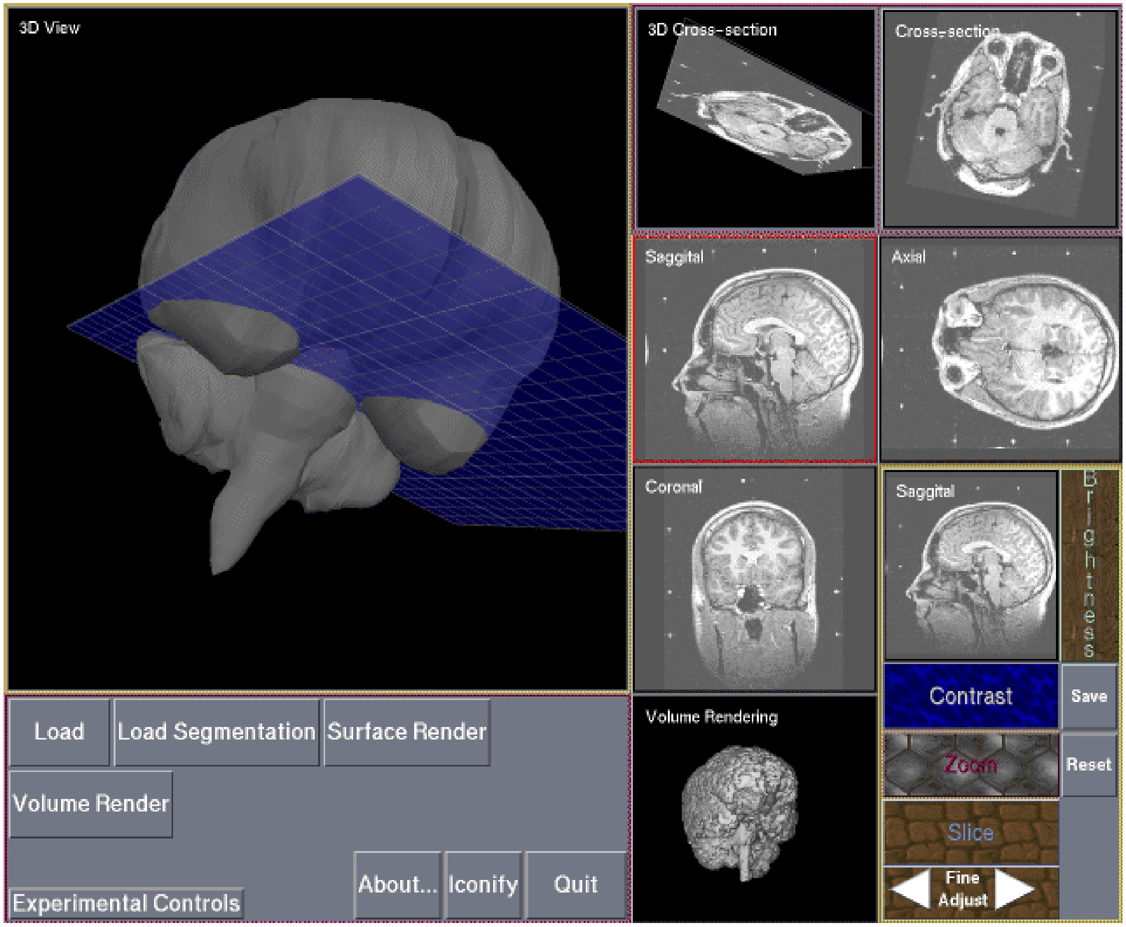

Figure 6: Touchscreen graphical user interface for use with the 3D props.

The brightness, contrast, zoom, and slice navigation touchscreen controls in the control panel (*fig. 6*) were suggested by the physical thumb-wheels which are found on many 2D medical image viewing consoles. When the user touches and drags the touchscreen thumb-wheels, the background textures slide up and down, giving immediate visual feedback. The touchscreen thumb-wheels were designed to be used without looking directly at them, because the user is typically focusing on the image being modified, and not the widget itself. The moving background textures can be seen even with peripheral vision, and so meet this “eyes-free” requirement effectively. Using a standard scrollbar on a touchscreen completely fails in this regard, since the user’s finger occludes the thumb of the scrollbar. An experimental implementation of the touchscreen thumb-wheels with nonspeech audio feedback has suggested that the technique would be even more effective with appropriate audio feedback.

Rather than having constraint modes for the 3D devices, we found that users were more comfortable expressing constraints using the naturally constrained dialog afforded by the touchscreen. Thus, the tiles which show the standard sagittal, axial, and coronal slice orientations act as subtle constraint modes; all operations on these tiles are naturally constrained to the appropriate axis of the volume. Similarly, once an oblique slice has been selected with the props, this becomes a tile (seen in the upper right of fig. 6) which expresses constraint along the normal of the currently selected oblique cutting-plane.

Informal observation of users of the touchscreen and props hybrid interface suggest that touchscreens offer not only intuitive 2D input which is well accepted by physicians, but that touchscreens also offer fast and accurate input which blends well with 3D interaction techniques.

While promising, the implementation of the hybrid interface concept has some shortcomings related to the surface acoustic wave (SAW) touchscreen technology which we use, including parallax errors and a limitation on the type of materials which can be used to touch the screen. The SAW touchscreen can only detect the presence of soft materials such as a finger or an eraser tip. This means that the interface props cannot be used to directly touch the screen, something which users naturally try to do. We believe that a resistive membrane touchscreen may provide a more suitable technology for this application, and in future work we would like to experiment with this.

7. Informal Evaluation with Neurosurgeons

The most valuable tool for evaluating and improving the interface design has been informal observation of test users. We have tested the interface with neurosurgeons as well as other physicians who work with volumetric data, such as neurologists, cardiologists, and radiologists. We have also performed many informal user observations during demonstrations to the general public.

The methodology for testing with surgeons was simple but effective. We almost always began by briefly showing the system to the visiting neurosurgeon. Many surgeons were eager to try it and would jump in themselves without needing an invitation. Then we would stand back and just watch the neurosurgeon operating the interface. If the surgeon seemed to be having some troubles, rather than immediately intervening with advice or suggestions, we would wait for the surgeon to ask a question or make a comment. In this way, we could understand a problem in the terms the surgeon was using to think about it. we found that a good question for stimulating discussion of the interface was: “Can you show me what you did during your most recent surgical procedure?”

Neurosurgeons have been very enthusiastic about the props-based interface. All of the neurosurgeons who have tried to use the interface were able to “get the hang of it” within about one minute of touching the props; many users required considerably less time than this. This clearly demonstrates that with a cursory introduction, neurosurgeons who have never before seen the interface can rapidly apply their existing skills for manipulating physical objects with two hands, and can understand and use the interface without training.

The overall response of physicians in other specialties has been similar to that of neurosurgeons, particularly from specialists who commonly deal with volumetric MRI data, such as data representing the knees, shoulders, or heart. From talking with these physicians, it is clear that oblique plane visualization problems commonly occur in these fields as well; for example, when visualizing the knee, oblique cuts along the ligaments are clinically valuable to assess the severity of an injury.

8. Conclusion

Informal evaluation of over fifty neurosurgeons, and of hundreds of non-physicians, has demonstrated that our interface facilitates transfer of the user’s skills for manipulating tools with two hands to the operation of an interface for visualizing volume data, without training.

Our interface design takes a decidedly minimalist approach: it is perhaps more notable for the technologies it does *not* use than for those it does use. It does not use standard virtual reality equipment and techniques such as immersive viewing goggles, head tracking, stereoscopic projection, or gloves. Nonetheless, the interface allows the user to see and explore the interior of a virtual object in interesting new ways. The point we are trying to make is that most applications of interactive 3D graphics do not need a full scale virtual reality implementation, but often some rather simple technology, along with an understanding of the human operator, can make a little technology go a long way. The interface uses six degree-of-freedom input devices and two-handed manipulation because these techniques are well suited to some of the data interaction and visualization problems that neurosurgeons are trying to solve.

9. References

1. Chung, J., “A comparison of Head-tracked and Non-head-tracked Steering Modes in the Targeting of Radiotherapy Treatment Beams,” Proc. 1992 Symp. on Interactive 3D Graphics, 193-196.
2. Goble, J., Hinckley, K., Snell, J., Pausch, R., Kassell, N., “Two-handed Spatial Interface Tools for Neurosurgical Planning,” *IEEE Computer*, July 1995, 20-26.
3. Guiard, Y., “Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model,” J. Motor Behavior, 19 (4), 1987, pp. 486-517.
4. Hinckley, K., R. Pausch, J. Goble, N. Kassell, “Passive real-world interface props for neurosurgical visualization,” ACM CHI’94 Conference on Human Factors in Computing Systems, 1994, pp. 452-458.
5. Hinckley, K., Pausch, R., Proffitt, D., “Attention and Visual Feedback: The Bimanual Frame-of-Reference,” ACM 1997 Symposium on Interactive 3D Graphics, to appear.
6. Hinckley, K., Goble, J., Pausch, R., Kassell, N., “New Applications for the Touchscreen in 2D and 3D Medical Imaging Workstations,” Proc. 1995 SPIE Conference on Medical Imaging.
7. Kabbash, P., MacKenzie, I. S., Buxton, W., “Human Performance Using Computer Input Devices in the Preferred and Non-Preferred Hands,” INTERCHI’93, pp. 474-481.
8. Kelly, P. J., B. A. Kall, eds., ***Computers in Stereotactic Neurosurgery***, Blackwell Scientific Publications Inc., Cambridge, MA, 1992.
9. Leksell, L. “A stereotactic apparatus for intracerebral surgery,” Acta Chir Scand 99, pp. 229-233, 1949.
10. Polhemus Navigation Sciences, Inc., (802) 655-3159.
11. Poston, T., Serra, L., “Dextrous Virtual Work,” Communications of the ACM, 39 (5), pp. 37-45, 1996.
12. Sachs, E., Roberts, A., Stoops, D., “3-Draw: A Tool for Designing 3D Shapes,” IEEE Computer Graphics & Applications, Nov. 1991, pp. 18-26.
13. Sellers, M., “Designing for Demanding Users,” ACM Interactions, July 1994, pp. 55-64.
14. Snell, J. W., Merickel, M. B., Ortega, J. M., Goble, J. C., Brookeman, J. R., Kassell, N. F., “Model-Based Boundary Estimation of Complex Objects Using Hierarchical Active Surface Templates,” Pattern Recognition, 1995.
15. Snell, J. W., Jackson, T. R., Katz, W. T., Hinckley, K., Goble, J. C., Kassell N. F., “A Three-Dimensional Stereotactic Neurosurgical Planner/Simulator”, Proc. 1995 SPIE Conference on Medical Imaging.