

Pre-surgical Cranial Implant Design using the PARIS™ Prototype

Chris Scharver, Ray Evenhouse, Andrew Johnson, Jason Leigh
Electronic Visualization Laboratory, Virtual Reality in Medicine Laboratory
University of Illinois at Chicago
scharver@evl.uic.edu, raye@uic.edu, aej@evl.uic.edu, spiff@evl.uic.edu

Abstract

Repairing severe human skull injuries requires customized cranial implants, and current visualization research aims to develop a new approach to create these implants. Following pre-surgical design techniques pioneered at the University of Illinois at Chicago (UIC) in 1996, researchers have developed an immersive cranial implant application incorporating haptic force feedback and augmented reality. The application runs on the Personal Augmented Reality Immersive System (PARIS™), allowing the modeler to see clearly both his hands and the virtual workspace. The strengths of multiple software libraries are maximized to simplify development. This research lays the foundation to eventually replace the traditional modeling and evaluation processes.

1. Introduction

Today a medical sculptor utilizes anatomical modeling expertise to sculpt a prosthetic implant. However even with the aid of automated manufacturing techniques, the design process poses several problems. Techniques developed at the University of Illinois at Chicago (UIC) in 1996 have greatly improved the practice. Virtual reality research now aims to augment these tools and methods using a prototype display system.

Closing large cranial defects offers patients therapeutic benefits. These benefits include restoring the shape of the head, protecting vital brain tissue, minimizing pain, reducing operating and recovery times, and in some cases improving cognitive capabilities. Unfortunately, several factors limit cranial implant availability. Insurance companies often will not support this form of reconstruction due to the high labor and material costs. Because only neurosurgeons and medical modelers possess the specialized anatomical knowledge, assembling the necessary expertise is difficult. Travel expenses for both patients and specialists increase the overall cost. Currently acrylic polymer is the most commonly used implant material. When used intra-operatively, the material exudes extreme heat as it solidifies. Exposing the brain to these temperatures can cause tissue damage. For this reason, pre-surgical implant

design and fabrication is vital. Traditional cranial implant fabrication and surgical placement methods are heavily dependent on subjective skills and procedures. At times it has been necessary to produce multiple implants of various sizes to provide the surgeon the choice of the best fitting implant during surgery. The cranium's anatomical complexity impedes reconstruction without extensive planning. Pre-surgical cranial implant design and fabrication alleviates many of these shortcomings.

The 1996 UIC approach incorporated a series of expensive manual manufacturing steps to produce a custom-fitting implant. These techniques serve as a guide for implementing a digital approach combining augmented reality and haptics. Medical sculptors are trained using their hands, and their abilities are heavily dependent on that fact. The awkward control devices used by traditional VR systems are not conducive to intricate sculpting techniques. Introducing force feedback allows the user to feel the virtual defect and implant models, giving the user the sense of touch that is important for 3D modeling [2]. The design process also requires medical sculptors clearly see their hands while modeling an implant. Using augmented reality we can combine real and virtual information, allow real time interactivity, and manage 3D registration [1].

Stereovision, wide angle-of-view, interactivity and viewer-centered perspective help the participants understand the depth relationships. Rather than looking at improperly scaled models on a computer monitor, the user perceives objects at an absolute scale. Combined with the sense of touch supplied by a PHANToM haptic device [3], this creates a rich sensory environment for developing a medical implant design application. Connecting these systems across high-speed networks enables users in different locations to collaborate among shared, virtual, and simulated environment [4]. Geographically scattered participants may discuss implant design, surgical pre-planning, and postoperative evaluation. This also provides educational opportunities, enabling instructors to interactively present methods and techniques. Our current prototype system allows loading and manipulating data, marking areas of interest, extracting defect geometry, and feeling the bone surfaces.

In section 2 we will describe the current methodology for pre-surgical cranial implant design at UIC. Section 3

describes the virtual reality hardware. Section 4 illustrates the application capabilities, while section 5 explains the software techniques utilized for development. Related work is examined in section 6. We draw conclusions about the lessons we have learned in section 7, and finally, section 8 explores future research.

2. Current Pre-surgical Implant Design

Dr. Fady Charbel, Ray Evenhouse and their team at UIC pioneered a pre-surgical cranial implant design technique in 1996 [5]. The process produces custom-fitting cranial implants prior to surgery using the patient's computed tomography (CT) data. The medical modeler loads the patient's CT data into medical computer-aided design (CAD) software and produces a digital model of the defect area. This polygonal model is exported to a rapid prototyping stereolithography machine. This computer-aided manufacturing (CAM) process fabricates a physical defect model. The medical modeler sculpts, molds, and casts the implant based on this model. Shaping the clay on the defect model, the modeler progressively sculpts the form of the implant's mold. The modeler then casts the implant by filling the mold with a medical-grade polymer. After casting, the implant is sterilized and prepared for surgery. To date, nine patients have received implants using this method.

2.1. Data Acquisition

A digital representation of the patient's skull is the basis of the implant design process. Using a General Electric High-Speed Advantage scanner, technicians generate one-millimeter slice thickness computed tomography scans. These images are archived on optical disk using the Digital Imaging and Communications in Medicine (DICOM) protocol and also sent through the university's data network to a graphics workstation. Data can then be stored for later processing or imported directly into modeling software such as Mimics.

2.2. Defect Specification

Several processing steps must be completed to construct a file suitable for rapid prototyping. Pixel data from the CT slices undergoes segmentation to extract the skull information. Semiautomatic algorithms in commercial software initiate the segmentation scheme, but each slice is checked and manually corrected as necessary. The edges of the inner and outer tables of the cranium are identified and their contours converted from a pixel map representation to a vector map. The pixel to vector map conversion establishes the geometry for each skull outline from the CT slices.



Figure 1. The UIC pre-surgical implant process generates a defect model from the patient's CT data. This model serves as the basis for sculpting the implant with clay and wax. The final implant is cast using a polymer resin.

Pixels and their volumetric equivalent, voxels, are image representations only and do not explicitly establish the geometry of the objects they represent. Conversely, vector maps are mathematical curves explicitly defining geometry. Vector maps from each CT slice are then connected to generate a surface model of the skull. To create an appropriate stereolithography output file for use in rapid prototyping, the software positions the surface model in the optimal orientation for the "build cycle" and re-samples the data into appropriate slice thickness. The re-sampled data now represents a tool path guide for the stereolithography machine's model generation process.

2.3. Defect Model Creation

The defect geometry file is imported into a stereolithography machine. This system contains a laser positioned over a vat of liquid polymer. As the laser fires into the polymer, its energy converts the polymer to a solid mass. As each slice from the CAD file is traced and solidified, the platform on which the model is being built is lowered. The surface of the developing model is flooded, and polymerization by the laser beam is repeated. Slice-by-slice, a solid plastic form is gradually created. The finished model is removed from the vat, drained of excess polymer, and submitted to a final curing stage using ultraviolet radiation. The final model is a precise physical representation of the patient's skull and defect CT data. To save costs, only the portion of the skull immediately surrounding the defect is generated.

2.4. Implant Fabrication

The stereolithography model serves as a physical template for fabricating the final implant. Modeling clay is applied to the plastic model of the patient's defect. The clay is contoured to represent the inner prosthesis surface. A silicone rubber mold is made of this composite piece, and dental stone is poured into this mold. This produces a physical model of the defect that serves as the foundation upon which the implant is built.

A dental grade wax is used to fill the negative space representing the skull defect. Care is exercised in maintaining appropriate contour and thickness to assure the best cosmetic appearance following implantation.

When the wax pattern for the implant is completed, the remaining half of the mold can be poured. Registration notches are cut into the defect side of the mold so that the two mold halves can be perfectly realigned. Dental stone is poured to form the top half of the mold.

The mold halves are separated and the wax is removed, leaving a negative space to receive the implant material. Medical-grade acrylic monomer liquid and polymer powder are mixed to form a soft, pliable mass and carefully packed into the mold to avoid air entrapment. The mold is then closed, tightly clamped, and placed in a hot-water bath to slow-cure the resin. Following the prescribed cure time, the solid implant is removed from the mold. Additional curing in an oven releases any residual free monomer trapped in the implant. After trimming and polishing, the finished implant is sent to the operating suite for sterilization.

2.5. Surgery

Depending on the bone defect location, the surgeon positions the patient in the appropriate position. After

shaving and cleansing the surgical site, the surgeon drapes the patient according to the cranial defect. The implant is fixed in place with a minimum of three titanium plates and screws. The surgeon seals the incision using sutures and closes the skin with staples.

2.6. Limitations

Even with this approach, several steps remain labor-intensive and expensive. Although trained specialists create the implants, the process remains heavily dependent on subjective skills and procedures. Stereolithography is a relatively slow process, and the price increases directly with the volume and overall dimensions of the model being built. For example, an entire skull takes 16 hours and costs approximately \$1700. A separation of responsibilities exists between the modeler designing the implant and the doctor performing surgery. Assembling these specialists for consultation and evaluation can be difficult.

3. PARIS™

The proposed solution involves enhancing virtual reality so that the modeler may work efficiently. A user must see his hands, and a new prototype display systems fills this need. Interface tools have been designed considering the 1996 UIC process. The description of this system reveals areas for future development.

3.1. Key Features

One of the first VR systems designed within VR, the Personal Augmented Reality Immersive System (PARIS™) incorporates significant improvements over previous projection-based virtual reality displays [6]. Previous systems such as the CAVE and the ImmersaDesk support 3D vision well, managing separate stereo images for each eye and tracking head motion. There remain at least two important depth perception cues, occlusion and accommodation, that are not supported correctly in previous displays. For instance, in a conventional projection-based virtual reality display, an object in front of the hand is obstructed by the hand itself. This occlusion causes a visual conflict because the hand, which should be behind the object, appears in front of the object. The object should be visible in front of the hand. The half-silvered mirror in the PARIS™ display superimposes the displayed image over the hands. One does not occlude the other. The second depth cue, accommodation, refers to muscles controlling the eye to adjust sharpness. In a conventional VR display the eye will always focus on the display screen, which is typically significantly farther than arm's reach. The PARIS™ display is designed so that the hands and the virtual object

are the same distance as the image of the screen. Unique to the PARIS™, these projection improvements are particularly important for sculpting. The modeler's hands, the constructed implant, and the patient's data are all visible in the same working volume.

3.2. Hardware Configuration

Configuring PARIS™ involves aligning the coordinate systems, thus enabling graphics and haptic software packages to maintain coincident model representations. Adding haptics to an augmented reality environment creates problems because these multiple coordinate systems must be measured and aligned [7]. Once aligned, the union of graphics and haptic representations presents a very compelling illusion to the user.

Screen projection corner measurements establish the location of the window into the virtual environment. A world origin is chosen, and all measurements must be taken relative to that origin position.

4. Application Capabilities



Figure 2. A medical modeler "touches" a cranial defect while sitting at PARIS™. The stylus corresponds to a tool held in the right hand; a virtual hand tracks the position of the left hand. A real model remains visible on the table.

Sitting at PARIS™ as shown in Figure 2 and Color Plate 1, the modeler interacts with the system in a manner similar to the methods pioneered at UIC in 1996. Any patient or family photographs are scanned and saved to the computer. The modeler then converts the patient's skull CT data into polygon geometry. The application loads all models and photographs into default positions. After adjusting positions as needed, the modeler may manipulate the skull data to obtain the best view of the defect. A pencil tool creates 3D annotations and highlights the edge of the defect. This defect outline serves as input for calculating a defect model separate from the entire skull. The user traces the defect edge interactively, so even irregularly shaped defect geometry

may be extracted. Viewing the defect, the sculptor makes annotations indicating where to attach bolts during the surgery. Feeling the surface of the defect allows the modeler to determine how to sculpt the patient's implant. Referring to the digitized photographs, the sculptor builds material into the hole, slowly shaping the implant. When the work is completed, the application saves the model file to disk for evaluation and eventual fabrication

4.1. Reference Images and Models

Traditionally, the medical modeler utilizes patient and family photographs for reference. These digital photographs provide visual cues for guiding the implant design. Rather than cluttering the workspace with numerous photographs, the modeler may instead load digital images directly into the virtual environment. These images may be moved freely throughout the environment. Transparent images may be positioned directly in relation to the CT data. Because it is not important to feel the image planes, these images are manipulated using the tracked pointer rather than the haptic device.

The design process requires an anatomically correct model of the patient's skull. Preprocessing converts the patient's skull CT data into Open Inventor geometry. Decimation reduces the detail so that the model will not impede interactivity within the virtual environment. This model is subsequently converted into the haptic library's scene graph for haptic rendering and tool interaction.

4.2. Pencil Tool

The first basic tool is a 3D pencil with which the user may mark points of interest. The conversion of CT data into geometry can result in threshold errors, and the resulting geometry may not accurately reflect the structure of the skull. The pencil tool allows users to draw and write freely, indicating those scanning errors or where to place clamps when attaching the implant to the bone during surgery.

4.3. Defect Specification

As described in section 2.2, defect specification is another important part of the planning process. The modeler must indicate the boundary of the cranial defect. Using the haptic feedback as a guide, the modeler can trace along that edge of the bone. As shown in Figure 3, a semi-transparent sphere indicates the volume included as part of the defect. The defect geometry must be extracted, and this process calculates the resulting data based on where the lines were drawn. Depending on the complexity of the defect, this process could take only a few minutes. Contrasted with the stereolithography fabrication, extracting the defect in the virtual

environment takes less time. The modeler may then begin designing the implant.

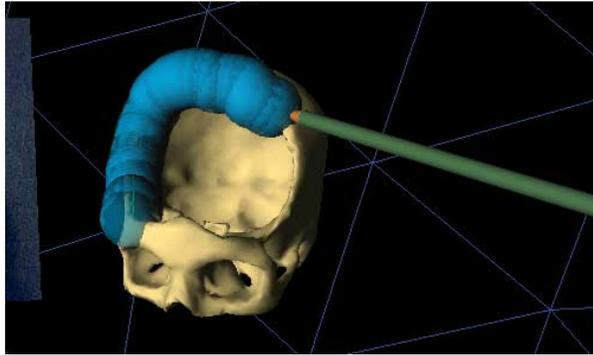


Figure 3. The defect tool traces a path for extracting the defect geometry.

4.4. Implant Modeling

Volume modeling is used for implant sculpting as shown in Color Plate 2. After the user drags the PHANToM™ through space, the application calculates a volumetric mesh based on the tool's path.

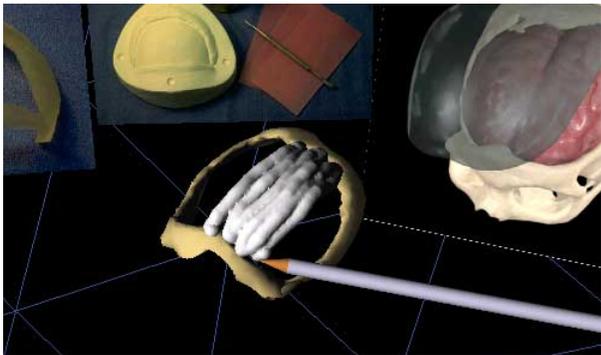


Figure 4. As the user draws lines to indicate the implant, VTK generates surface geometry.

Figure 4 depicts the user creating the volumetric implant. Notice the similarity of the defect with the physical model in Figure 1. Current research is focused strongly on refining the implant volume sculpting. The interaction should be fluid for the medical sculptor, but this is only one of the many areas of future development.

5. Software Implementation

The software architecture directs information between multiple libraries. The application receives tracking data from the wand in a read-only manner. Interaction with the PHANToM requires tighter integration. It is also important to mediate between the different data formats used by the rendering and haptic libraries.

5.1. Communication Between Software Libraries

Leveraging the use of existing libraries allows development to focus on the environment rather than low-level implementation details. This system has been developed using the CAVE Library (CAVELib), TGS Open Inventor, Kitware's Visualization Toolkit (VTK), and SensAble Technologies' General Haptics Open Software Toolkit (GHOST). CAVELib handles the model and view transformations for head tracked stereo rendering. It generates display threads that synchronize active stereo rendering. Open Inventor is an OpenGL scene graph library optimized for interaction and easy extensibility. TGS Open Inventor is built upon SGI Open Inventor 2.1, enhancing that version's capabilities with multi-threading and immersive interactivity. The Open Inventor model format is human-readable and supported by many 3D modeling packages. VTK provides a comprehensive collection of visualization algorithms to store raw data and construct visual representations of that data. Finally, GHOST provides the haptic rendering by interfacing with the PHANToM Desktop device. It spawns a haptic servo thread for handling force feedback processing. GHOST has its own scene graph differing from Open Inventor and VTK, but similar concepts are used.

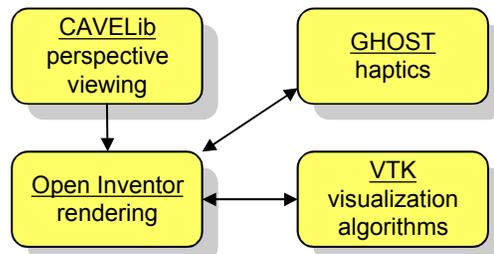


Figure 5. Software toolkit relationships

Generating usable models requires converting between the toolkits' data representations. Employing separate toolkits maximizes the strengths of each. Figure 5 illustrates the responsibilities and relationships among the three toolkits. CAVELib and GHOST provide input by sampling user positions from tracking and haptic hardware, respectively. VTK generates geometry using volume modeling methods. VTK data structures may be easily converted to Open Inventor scene graph nodes. Once VTK finishes calculations, it converts the results to an Open Inventor model. This scene graph is loaded into the main application and used for visualization and graphical interaction. The application also converts the resulting triangle geometry to GHOST's scene graph. GHOST communicates with the PHANToM™ hardware and generates force feedback that coincides with the environment's graphics.

5.2. Resolving Coordinate Systems

Accurately resolving the multiple hardware coordinate systems is essential for providing a convincing unity of real and virtual objects. The PHANToM offset measurement is placed into a transform within the scene graph. Measured from the world origin, this offset ensures that the PHANToM force feedback is coincident with the modeler’s view of the virtual environment.

5.3. Open Inventor-GHOST Integration

Open Inventor and GHOST both store data in scene graphs containing varying node types, and making the two libraries interact demands careful management. Open Inventor includes textures, shapes, transformations, manipulators, and cameras. Although Open Inventor allows lateral state inheritance during scene traversal, GHOST only allows top-down inheritance. Open Inventor also allows the same node to appear multiple times in a scene graph. GHOST prohibits this repetition.

These structural differences are most important for transformations. Open Inventor stores transformations as matrices within nodes. Multiple SoTransform nodes result in accumulated transformation matrices. GHOST’s `gstSeparator` nodes contain a single transform matrix. Storing multiple transform matrices requires chaining `gstSeparators`. Software must manage changes in either scene graph and reflect those changes in the other representations.

5.4. Open Inventor-VTK Integration

The Visualization Toolkit executes a pipeline that performs a series of filters on data sources. VTK refers to its visualization result objects as actors. As VTK does not provide a scene graph, Open Inventor integration requires implementing collaboration classes. Two classes, `vtkOIVSource` and `vtkActorToIV`, serve as the bridges between the VTK pipeline and Open Inventor. Conversion from VTK to Open Inventor is based on Paul Rajlich’s VTK to Performer implementation. [8].

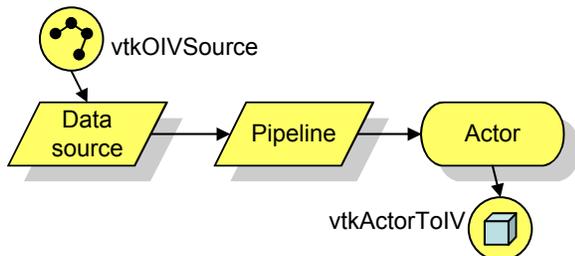


Figure 6. `vtkOIVSource` and `vtkActorToIV` bridge Open Inventor and VTK.

The first class, `vtkOIVSource`, converts Open Inventor geometry to a data source at the beginning of VTK’s pipeline. Software plugs vertex-based Open Inventor geometry directly into the VTK pipeline. After VTK executes the pipeline and generates visualization geometry, `vtkActorToIV` converts the resulting triangles back into an Open Inventor scene graph. This processing makes available VTK’s comprehensive algorithms while retaining Open Inventor scene graph organization.

5.5. Controller Abstraction

Force feedback is a key feature for using PARIS™, but remote users may not have the required hardware. Interaction must still be available with only traditional tracking devices. The implementation abstracts the CAVELib and GHOST control devices. A combination of the Observer and Strategy patterns [9] separates controllers and user interface tools.

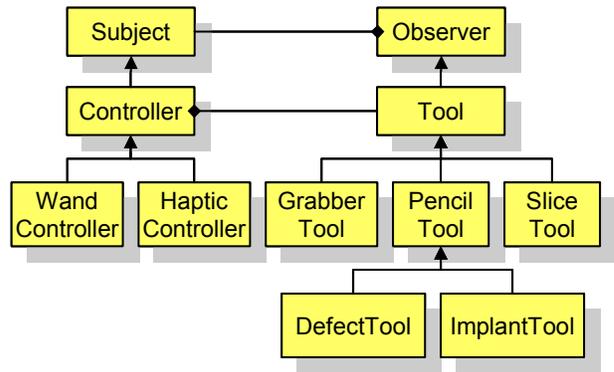


Figure 7. Controller-Tool Abstraction

Figure 7 illustrates the relationships between the Controller and Tool classes. The `WandController` and `HapticController` classes communicate with the CAVELib and GHOST toolkits, respectively. As devices’ states change, Controllers notify their Observers. Every Tool is a separate Strategy, and each responds uniquely to Controller events. Force feedback responses may be customized within a Tool, but such behaviors only apply when observing a `HapticController`.

6. Related Work

Previous work explores both computed-aided cranial implant design and immersive modeling, but the challenges of unifying these concepts into an immersive cranial implant design application have not been examined. Any immersive modeling tools must be appropriate for the tasks, as the medical modeler requires specific tools and capabilities.

Key related work leverages computer-aided design during the planning process, and several methods successfully created cranial implants. One approach combines rapid prototyping and commercial modeling software [10]. Another method utilizes skull symmetry and numerical analysis, providing solutions in certain cases [11]. While UIC uses medical modeling software to build the defect, these projects construct the implant entirely in software. In either case, such tools require complex interaction techniques. Additionally, users operate only within 2D windows. These 2D interactions are difficult to bring into a virtual environment [12]. If anatomical training involves sculpting with clay and wax, as at UIC, the virtual environment demands interaction with natural techniques rather than flat 2D windows.

Modeling within an immersive environment presents unique challenges. Programming tools often provide easy creation of simple geometric shapes such as boxes, spheres, cylinders, and lines. HoloSketch [13] presented a non-immersive stereo display to the user. The head-mounted display modeler 3DM [14] followed CAD techniques like requiring specific axes and providing extensive 2D menus and windows. More effort is required to support clay-like 3D volume sculpting. Recent experiments indicated that both experienced and inexperienced VR users preferred volume modeling to geometric modeling [15].

Spatial input creates problems since there are often no physical constraints on user input movements [16]. The introduction of force feedback addresses this weakness. Although commercial products such as SensAble's FreeForm have been used for medical modeling [17], these tools are closed to custom development. Researchers cannot add targeted features supporting tele-immersion to the existing software.

7. Lessons Learned

Developing this application on PARIS™ revealed several issues with both the hardware and software. Lessons learned from these experiences are already being incorporated into future research plans.

7.1. Hardware Issues

The PARIS™ demonstration at Supercomputing 2002 intended to utilize acoustic tracking. The confined area under the mirror created significant feedback to the audio hardware. Software driver limitations prevented solving these complications. Magnetic tracking has been used as a fallback method while development addresses the acoustic tracking problems.

Magnetic tracking requires a base sensor unit, and that unit is currently mounted under the table at which the modeler sits. The head and hand are both tracked using

magnetic trackers. However, if these trackers move too close to the sensor unit, then interference negates tracking. As a result, the wand sensor cannot be used directly in front of the modeler. The PHANTOM is available in this area, but this limitation remains something to address. Another drawback to this sensor position is that any table movement will offset the measurements for the entire environment.

By definition, augmented reality depends on registration of real and virtual objects [1]. Any table or mirror movement compromises calibration. Future hardware designs must minimize any interference and maximize stability.

7.2. Software Techniques

Identifying the common capabilities of the multiple libraries facilitated application development. Managing separately the haptic and graphic representations reduced development by relying on existing libraries.

These software techniques are being applied by the Rehabilitation Institute of Chicago to design stroke rehabilitation software using the second manufactured PARIS™ [18]. Although that configuration does not use tracking, it does include the larger PHANTOM 3.0 for a tremendously increased haptic work area.

7.3. Defect and Implant Specification

Using the traditional process, manufacturing a stereolithography defect model can take several hours and cost almost \$2000. Using PARIS™, the user may rapidly specify the defect interactively. Depending on the complexity of the injury, the medical modeler can outline a defect in less than one minute.

Implant modeling requires further development. Using VTK is convenient, but it is slow and does not fully integrate with the notion of a scene graph

8. Future Research

Full effectiveness cannot be determined without testing with real patient data. Volume sculpting requires further attention now that developers have obtained familiarity with the hardware. As part of a three-year research grant, evaluation will compare digital models against traditionally constructed models. The digital tools will be refined continuously. Over the next two years, we will implement and profile the networking between PARIS™ and other immersive display systems. A new PARIS™ system with tracking and hardware improvements will also be built as part of this research.

Separating computational resources, we will explore how best to take advantage of clustered computing. Only Open Inventor and the CAVELib are tied directly to the

graphics hardware. VTK and the GHOST SDK, both of which primarily perform calculations, do not require graphics processing. These libraries may run on high-performance processors. Clusters may allow partitioning the computational tasks across multiple machines. Intelligently managing computer resources could enable multiple users to securely share medical data, analysis material, evaluation sessions, and more actively steer implant construction.

The consultation components can be applied to any area where medical professionals would benefit from a rich visualization environment combined with superior teleconferencing tools. We will explore these other areas in the third year of the grant. Color Plate 3 depicts the environment enhanced with potential patient information and medical CT slices. These capabilities were demonstrated to radiologists at the Radiological Society of North America's 2003 conference. There is interest in utilizing this system for the pre-operative consultation and planning related to reconstructive surgery. As network capabilities improve, the application can transmit more data to more clients. Future research hopes to determine definitively that utilizing a networked virtual reality system will minimize the consultation time and cost to the patient, surgeon, and manufacturer.

9. Acknowledgements

The Virtual Reality in Medicine Laboratory at the University of Illinois at Chicago (UIC) is funded, in part, by the National Library of Medicine/National Institutes of Health contract N01-LM-3-3507. The Electronic Visualization Laboratory (EVL) at UIC receives major funding for virtual reality equipment acquisition and development and advanced networking research from the National Science Foundation (NSF), awards EIA-9871058, EIA-9802090, EIA-0224306 and ANI-0129527, as well as the NSF Partnerships for Advanced Computational Infrastructure (PACI) cooperative agreement (ACI-9619019) to the National Computational Science Alliance. Previously, EVL received funding for PARIS from the US Department of Energy (DOE) ASCI VIEWS program. PARIS is a trademark of the Board of Trustees of the University of Illinois.

10. References

[1] R. Azuma, "A Survey of Augmented Reality," in *Presence: Teleoperators and Virtual Environments*, vol. 6, 1997, pp. 355-385.
 [2] T. Massie, "A Tangible Goal for 3D Modeling," in *IEEE Computer Graphics and Applications*, vol. 18, 1998, pp. 62-65.
 [3] J. Salisbury, Srinivasan, M., "Phantom-Based Haptic Interaction with Virtual Graphics," in *IEEE Computer Graphics and Applications*, vol. 17, 1997, pp. 6-10.

[4] K. Park, Cho, Y., Krishnaprasad, N., Scharver, C., Lewis, M., Leigh, J., "CAVERNsoft G2: A Toolkit for High Performance Tele-Immersive Collaboration," presented at ACM Symposium on Virtual Reality Software and Technology, Seoul, Korea, 2000.
 [5] M. Dujovney, Evenhouse, R., Agner, C., Charbel, LFT., Sadler, L., McConathy, D., "Performed Prosthesis from Computer Tomography Data: Repair of Large Calvarial Defects," in *Calvarial and Dural Reconstruction*, S. Rengachary, Benzel, E.C., Ed. Park Ridge, Illinois: American Association of Neurological Surgeons, 1999, pp. 77-88.
 [6] A. Johnson, Sandin, D., Dawe, G., DeFanti, T., Pape, D., Qiu, Z., Thongrong, S., Plepys, D., "Developing the PARIS: Using the CAVE to Prototype a New VR Display," presented at ACM Symposium on Immersive Projection Technology, Ames, Iowa, 2000.
 [7] J. Vallino, Brown, C., "Haptics in Augmented Reality," presented at IEEE International Conference on Multimedia Computing and Systems, Florence, Italy, 1999.
 [8] P. Rajlich, "vtkActorToPF," <http://brighton.ncsa.uiuc.edu/~prajlich/vtkActorToPF/>, 2003.
 [9] E. Gamma, Helm, R., Johnson, R., Vlissides, J., *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading, Massachusetts: Addison-Wesley, 1995.
 [10] F. Taha, Testelin, S., Lengele, B., Boscherini, D., "Modeling and design of a custom made cranium implant for large skull reconstruction before a tumor removal," http://www.materialise.com/medical-rpmodels/case11_ENG.html, 2001.
 [11] O. Burgert, Seifert, S., Salb, T., Gockel, T., Dillmann, R., Hassfeld, S., Mühling, J., "A VR-System Supporting Symmetry Related Cranio-Maxillofacial Surgery," presented at Medicine Meets Virtual Reality, Newport Beach, California, 2003.
 [12] M. Mine, "Working in a Virtual World: Interaction Techniques Used in the Chapel Hill Immersive Modeling Program," University of North Carolina at Chapel Hill 1996.
 [13] M. Deering, "HoloSketch a virtual reality sketching/animation tool," *ACM Transactions on Computer-Human Interaction*, vol. 2, pp. 220-238, 1995.
 [14] J. Butterworth, Davidson, A., Hench, S., Olano, T., "3DM: A Three Dimensional Modeler Using a Head-Mounted Display," presented at ACM Symposium on Interactive 3D Graphics, Cambridge, MA, 1992.
 [15] J. Deisinger, Blach, R., Wesche, G., Breining, R., Simon, A., "Towards Immersive Modeling - Challenges and Recommendations: A Workshop Analyzing the Needs of Designers," presented at EuroGraphics Conference of Virtual Environments, Amsterdam, the Netherlands., 2000.
 [16] K. Hinkley, Pausch, R., Goble, J., Kassell, N., "A Survey of Design Issues in Spatial Input," presented at ACM Symposium on User Interface Software and Technology, Marina del Rey, California, 1994.
 [17] T. Kling-Petersen, and Rydmark, M., "Modeling and Modification of Medical 3D Objects. The Benefits of Using a Haptic Modeling Tool," presented at Medicine Meets Virtual Reality, Newport Beach, California, 2000.
 [18] R. Kenyon, "Virtual Reality and Real Habilitation," http://www.nichd.nih.gov/about/ncmrr/symposium/Kenyon_slides.htm, 2001.