

Automating the Extraction of 3D Models from Medical Images for Virtual Reality and Haptic Simulations

Silvio H. Rizzi, P. Pat Banerjee, and Cristian J. Luciano

Abstract— The Sensimmer platform represents our ongoing research on simultaneous haptics and graphics rendering of 3D models. For simulation of medical and surgical procedures using Sensimmer, 3D models must be obtained from medical imaging data, such as Magnetic Resonance Imaging (MRI) or Computed Tomography (CT). Image segmentation techniques are used to determine the anatomies of interest from the images. 3D models are obtained from segmentation and their triangle reduction is required for graphics and haptics rendering. This paper focuses on creating an integrated interface between Sensimmer and medical imaging devices, using available software. Existing tools are evaluated, as well as aspects that require further development are identified. Solutions to overcome limitations and increase the degree of automation of the process are examined.

I. INTRODUCTION

SENSIMMER represents ongoing evolution of ImmersiveTouch [1], [2], the latest generation of augmented Virtual Reality (VR) technology, which integrates a haptic device with a head and hand tracking system, and a high-resolution and high-pixel-density stereoscopic display. A haptic device collocated with 3D graphics is the key factor to deliver extremely realistic simulations. Previously, ImmersiveTouch has been successfully applied to the simulation of neurosurgical procedures and training of neurosurgery residents [3]. It implements graphics and haptics rendering in a multi-threaded environment. In order to satisfy the minimum required graphic and haptic frame rates, it is essential to use efficient 3D models of the anatomical parts to be simulated.

In the context of on-demand high fidelity simulations [4], automatic or semi-automatic techniques to generate 3D models from medical images are highly desirable. Segmentation techniques are applied to Magnetic Resonance Imaging (MRI) or Computed Tomography (CT) images to obtain 3D models from medical data. These models have to

be reduced and converted to polygonal surfaces for simultaneous graphics and haptics rendering. Depending on the application, further processing may be needed (e.g. drilling of burr holes for simulation of a neurosurgery procedure known as ventriculostomy).

This paper describes ongoing research on automating the process of 3D model generation to be used in Sensimmer. For this purpose, a number of freely available open-source software tools are evaluated, namely, the Visualization Toolkit (VTK) [5], the Insight Segmentation and Registration Toolkit (ITK) [6], and ITK-SNAP [7]. Using these tools, a case of neurosurgery where ventriculostomy has to be performed on a patient with shifted ventricles is processed. From this semi-automated setup, aspects that require further development are identified and solutions to overcome limitations are examined.

II. BACKGROUND

A. Sensimmer

The Sensimmer platform, which represents ongoing evolution of ImmersiveTouch [1], [2], is capable of simultaneous haptics and graphics rendering of 3D models. As an important feature of its software design, several models representing different organs or objects can be loaded at the same time. This requires implementing a sophisticated collision detection mechanism and the assignment of different haptic properties to each model. For medical applications, the models are extracted from MRI or CT scans. Collections of segmented images are converted to 3D volumes, to be further transformed into polygonal surfaces. A frame rate of 60 Hz is used for stereoscopic graphics rendering, whereas a rate of 1000 Hz is used for haptics rendering. It is important not only to obtain accurate 3D models, but also to make them efficient for graphics and haptics rendering. Polygonal meshes obtained from segmentation must be carefully decimated; otherwise the number of polygons would be excessively high, making it impossible for the haptics library to perform adequately at the above framerate.

Figure 1 shows a block diagram of the system. This paper focuses on the automation of the pre-processing block.

B. Software Tools

The Visualization ToolKit (VTK) [5] is an open-source software package for visualization that supports a wide variety of advanced visualization and volume processing algorithms. In VTK, it is possible to construct visualization pipelines consisting of data and process objects [8].

Manuscript received April 30, 2007. This work was supported in part by the U.S. Department of Commerce NIST ATP Cooperative Agreement 70NANB1H3014 and by the Office of Naval Research through a subcontract from the National Center for Supercomputing Applications, University of Illinois TRECC Accelerator Program.

S. H. Rizzi is with the Department of Mechanical and Industrial Engineering, University of Illinois, Chicago, IL 60607 USA (e-mail: srizzi2@uic.edu).

P. P. Banerjee is with the Department of Mechanical and Industrial Engineering, Department of Computer Science, and Department of Bioengineering, University of Illinois, Chicago, IL 60607 USA (phone: 312-996-5599; fax: 312-413-0447; e-mail: banerjee@uic.edu).

C. J. Luciano is with the Department of Mechanical and Industrial Engineering, University of Illinois, Chicago, IL 60607 USA (e-mail: clucia1@uic.edu).

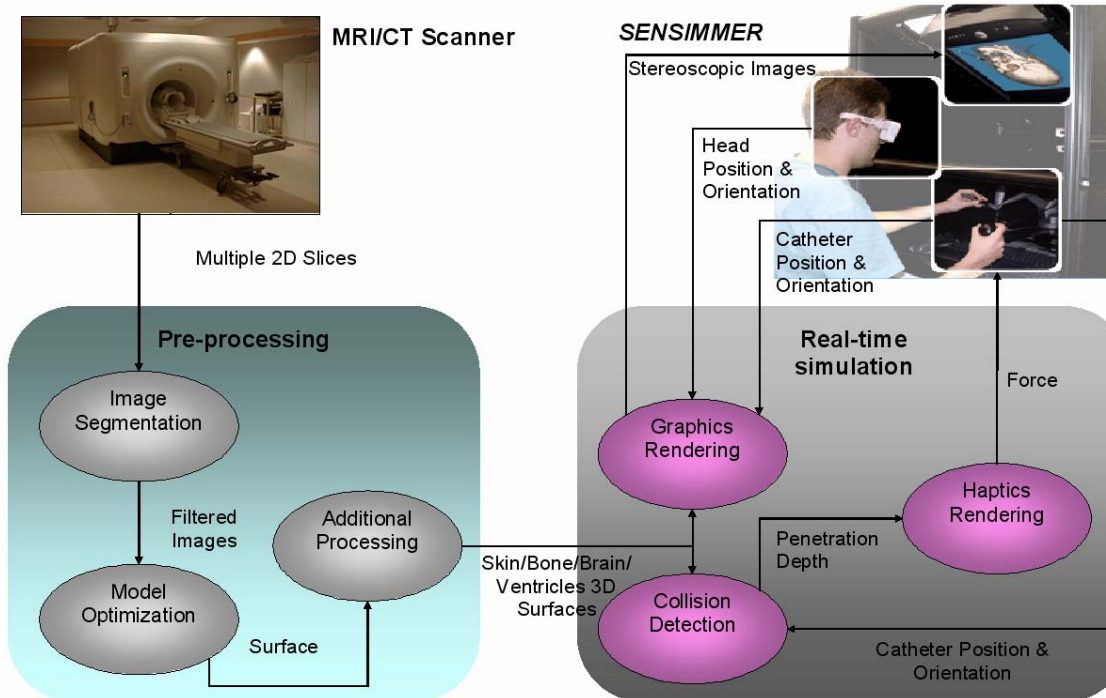


Fig. 1 Interface between Sensimmer and medical imaging devices.

The Insight Segmentation and Registration Toolkit (ITK) [6] is an application framework initially developed to support the U. S. National Library of Medicine's Visible Human Project [9]. ITK-SNAP [7] is an open-source software package, built on top of ITK, oriented to the segmentation of 3D anatomical structures from medical images. Using ITK-SNAP, it is possible to perform segmentation as a semi-automated procedure. Though referred as snakes [10] within the software, ITK-SNAP uses two 3D active contour segmentation methods [11], namely Geodesic Active Contours [12] - driven by intensity edges - and Region Competition [13] - driven by intensity regions. ITK-SNAP has been validated as a highly reliable tool in the context of a child autism neuroimaging study [14].

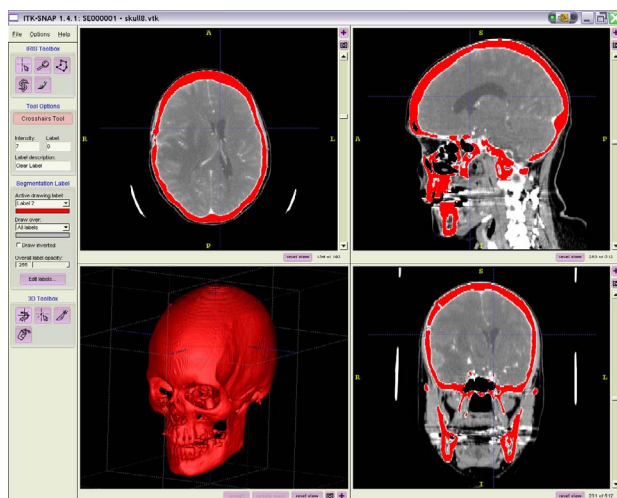


Fig. 2. Segmentation of a skull in ITK-SNAP.

C. Literature survey

Several systems and methods to attack similar problems have been reported in the literature. Cebral and Löhner [15] propose a method for construction of arterial surface models from medical images. A region-growing algorithm is used to segment the arteries. After isosurfacing, smoothing, and mesh optimization, a finite element mesh suitable for computational fluid dynamics calculations is generated. User interaction is required during the segmentation and geometry modeling stages. Examples where the method succeeds are shown. Algorithms and methods are discussed but specific software implementations are not presented. The time required to perform the entire process is reported to be in the order of few hours for the examples shown.

Du, Yang, and Du [16] present an integrated system based on ITK and VTK. Their focus is on creating finite elements models from CT or MRI images, but specific applications are not shown. Segmentation is done either manually or automatically. However, it is not clear how the system determines the regions to be segmented when working in automatic mode. Besides that, it is neither specified how much user interaction is required nor the time needed to perform the entire process for common cases.

Ito, Shum, Shih, Soni, and Nakahashi [17] propose a method for unstructured mesh generation for high-fidelity numerical simulations. Their main goal is to obtain high quality surface meshes. For that, two different approaches are used: direct advancing front method and modified decimation method. ITK and VTK are used along with custom code. Segmentation is done using threshold filters and transfer functions. The authors remark the importance of

working closely with medical experts to validate the results of segmentation.

Young et al. [18] present examples where 3D image data is automatically converted into polygonal meshes. Since the approach used is outlined but not detailed, it is difficult to evaluate the degree of automation achieved. Also, the degree of human intervention required is not discussed.

Melonakos, Al-Hakim, Fallon, and Tannenbaum [19] present an implementation in ITK of a segmentation algorithm based on incorporating prior knowledge through Bayes' rule. The intensity value of each voxel is considered a random variable. Additional assumptions on intensity distributions and prior likelihoods are made. According to the authors, this knowledge-based segmentation algorithm requires minimal user interaction. Examples of two different applications where the algorithm succeeds are shown.

Wolf et al. [20] describe The Medical Imaging Interaction Toolkit (MITK), an object-oriented, cross-platform library extending VTK and ITK. According to the authors, their goal is not to reinvent anything already existing, but to add previously non available features. It is stressed the fact that software for clinical use in image-guided procedures and image analysis requires a high degree of interaction to verify and correct results from automatic or semi-automatic algorithms.

III. METHODOLOGY

In this section we present a challenging case of ventriculostomy to be performed on a patient whose ventricles are notoriously shifted. For this experiment, the input data consists of 192 CT images in DICOM format. Each image contains intensity levels of 512 by 512 pixels. Intensity levels are given as 16-bit signed integers.

For the existing ventriculostomy simulator [3], there are four models to be extracted from patient data: ventricles, brain, skull, and skin (figure 1).

The process for obtaining 3D models from medical images consists of three main stages: (i) Segmentation, (ii) Model Optimization, and (iii) Additional Processing

A. Segmentation

ITK-SNAP provides a friendly user interface by which the user guides the segmentation process in a semi-automatic manner. Its source code is freely available, it is being actively developed, and it has a growing community of users. Version 1.4.1 is used in this work.

Series of CT images in DICOM format are read with ITK-SNAP. Orthogonal axial, coronal, and sagittal planes are displayed. The user can adjust the image histogram to enhance the visualization and contrast of the anatomical part under study. As the first step in segmentation, the volumetric region on which to perform the segmentation is selected. Input images have to be preprocessed before being fed into the segmentation algorithm. ITK-SNAP provides two methods for image preprocessing: Intensity Regions and Image Edges. In our case, Intensity Regions is the method that gives best results. Essentially, the method consists of applying a thresholding function to the input images. Two

probability fields are estimated: the probability that a pixel in the image belongs to the structure, and the probability that the pixel belongs to the background. The active contour is attracted to the points where both probabilities are equal [11].

Table I shows the type of thresholding function and the values used to obtain our models, whereas Figure 3 shows the resulting intensity regions for ventricles. There is a substantial amount of noise artifacts on the left side (top left view). Initially, the user must place 3D spheres of variable radius - called "bubbles" - as starting values for the algorithm. Afterwards, the active contour evolves continuously in each iteration, eliminating great part of the original noise.

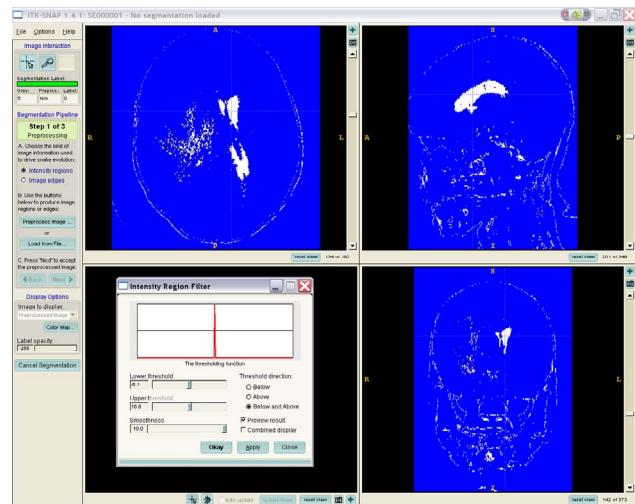


Fig. 3. Intensity regions for ventricles.

After a number of iterations the user must stop the process. The decision of when to stop iterations is solely based on user's expertise - there is not a stop criterion in the current ITK-SNAP implementation. Figure 4 shows the final result of the shifted ventricles segmentation as well as a 3D view of its isosurface.

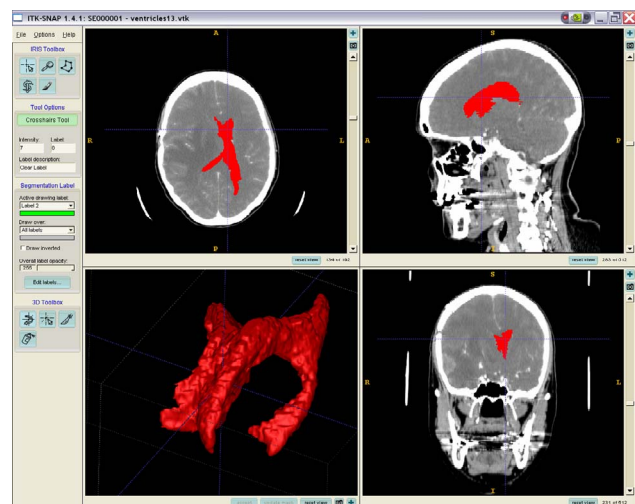


Fig. 4. Segmentation of ventricles.

TABLE I

Model	Type of filter	Threshold value(s)
<i>Skin</i>	Below	-362.8
<i>Skull</i>	Below	100.4
<i>Ventricles</i>	Below and Above	-6.1 ; 16.8
<i>Brain</i>	Below and Above	-15.6 ; 150.2

B. Generation and optimization of 3D models

At this stage, results of the segmentation process are available as a set of monochrome images. VTK is used to construct an optimal volumetric representation of the data. Figure 5 describes the VTK pipeline used. The first stage consists of a `vtkDataSetReader` filter, which reads the data segmented in ITK-SNAP and outputs volumetric data. The dimensionality of the input data is the same for each of the four models considered (512 x 512 x 192 voxels). Hence, at this stage in the pipeline we have 49874111 cells and 50331648 points, requiring 50.332 MBytes of memory.

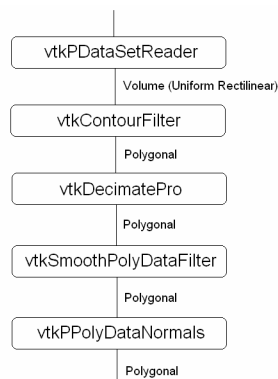


Fig. 5. VTK pipeline for model optimization

The next filter is `vtkContourFilter`, which generates an isosurface from its input data. This isosurface must be drastically decimated to reduce the number of polygons to be rendered. A `vtkDecimatePro` filter is used with the option "preserve topology" activated. Table II summarizes the number of polygons, vertices, and required memory for each model before and after decimation.

TABLE II

	Model	Before decim.	After decim.	% reduc.
<i>Brain</i>	Number of polygons	495364	22742	95.4
	Number of vertices	247748	11437	95.4
	Memory (MBytes)	13.872	0.822	94.0
<i>Ventricles</i>	Number of polygons	93528	23176	75.2
	Number of vertices	46782	11606	75.2
	Memory (MBytes)	2.621	0.837	68.0
<i>Skin</i>	Number of polygons	916608	83346	90.9
	Number of vertices	458306	41675	90.9
	Memory (MBytes)	25.666	3.003	88.3
<i>Skull</i>	Number of polygons	1495636	79810	94.7
	Number of vertices	747604	39691	94.7
	Memory (MBytes)	41.875	2.87	93.1

The degree of decimation is determined by visually inspecting the density of polygons in the resulting polygonal

meshes. Figure 7 exemplifies a polygonal mesh obtained from a brain model which could be further decimated without significant loss of information.

Following decimation, a smoothing filter is applied using `vtkSmoothPolyDataFilter`. The number of iterations required for each model was determined observing the smoothness of the model. After that, a `vtkPPolyDataNormals` filter is applied to generate normal vectors for each polygon. This step is essential for a correct rendering of the resulting models. Finally, the polygonal mesh is saved in VTK file format to be further converted into VRML or a different 3D representation (e.g. Stereo Lithography - STL) should additional processing is needed.

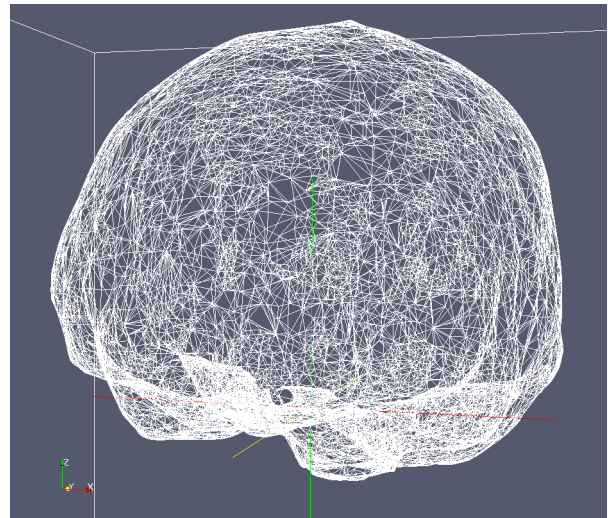


Fig. 7. Polygonal mesh representation of the brain

The VTK pipeline to obtain VRML models is shown in Figure 8. The `vtkVRMLExporter` filter converts the polygonal mesh to VRML. If additional processing is to be performed with another tool, different file formats are better and more efficient options. STL was chosen, since we use 3DS Max to drill the burr holes in our models. In this case, `vtkSTLExporter` replaces `vtkVRMLExporter`.

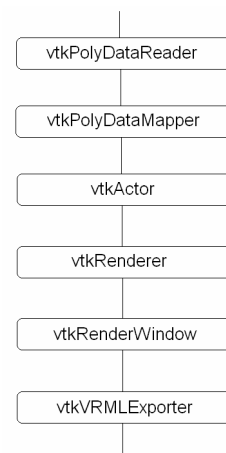


Fig. 8. VTK pipeline for VRML generation

C. Additional processing

In ventriculostomy, burr holes must be drilled according to anthropometric measures [22]. We simulate the process creating cylinders and ellipsoid-like shapes that are combined with the original model using Boolean operations in 3DS Max. Using this tool, not only can the user easily find an optimum view of the model, but also the proper placement and orientation of the cylinders and ellipsoid-like shapes are facilitated (Figure 9).

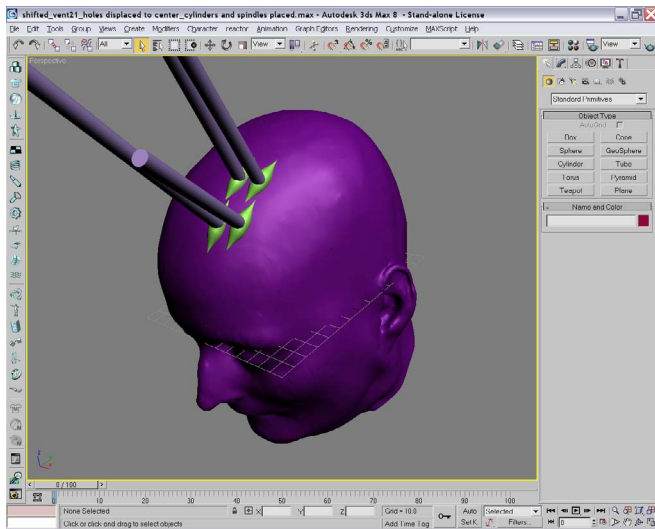


Fig. 9. Drilling burr holes

A boolean(-) operation is performed between the skin model and the ellipsoid-like shapes. Similarly, a boolean(-) operation is also performed between the skull model and the cylinders. Before exporting the final models in VRML, 3DS Max is also used to edit their color, lighting and texture mapping. Different views of the final models are shown in Figure 10.

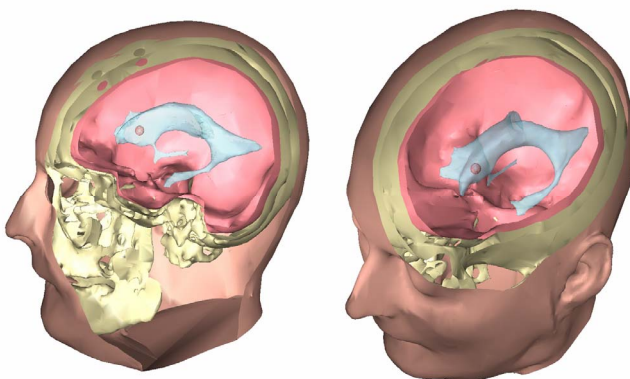


Fig. 10. Final models. The small sphere inside the ventricles indicates the target region for the ventriculostomy simulation

IV. DISCUSSION

In this section we analyze the results obtained in the previous section and elaborate on the improvements we plan to develop in order to increase the degree of automation of the process. All the modifications proposed can be neatly

integrated on a single user interface that will be part of the Sensimmer software, resulting in a compact and highly integrated product.

A. Segmentation with ITK-SNAP

One of the limitations found in ITK-SNAP is that there is not a stop criterion for the iterative process. In addition, during its use we repeatedly found crashes, especially when using a big number of bubbles or when the region of interest is too large. This does not prevent its effective use, since a segmentation process can be split into multiple regions to be individually processed. Furthermore, since the package is being actively developed, these bugs will eventually be corrected.

To segment a model, the region of interest must be defined, values for the preprocessing stage must be supplied to the algorithm, and finally the bubbles must be placed and its radii selected. Once the iterative process is running the user must stop it when considered appropriate. Manual segmentation is allowed, resulting useful to correct deficiencies in the automatic process. Based on these considerations, we found the degree of user intervention needed is moderate.

Since the source code for ITK-SNAP is freely available we plan to add the following modifications: (i) initial estimation of the region of interest based on statistical data, (ii) estimation of preprocessing parameters based on statistical data, (iii) initial placement of a number of bubbles inside the region of interest, taking advantage of the fact that the bubbles either shrink or grow attracted to the boundaries, and (iv) implementation of a stop criterion either based on a maximum number of iterations or on the fraction of voxels segmented with respect to the total volume of the region of interest.

B. Model optimization with VTK

There are two problems to be solved in model optimization. The first is the degree of decimation applied to the isosurface obtained from the segmented data. The second problem is the number of smoothing iterations applied.

The degree of decimation applied can be determined from previous knowledge of the maximum number of polygons either allowed or desired for a given model. The number of smoothing iterations is a parameter that must be determined by the user based on the visual evaluation of the resulting models. However, the system can suggest representative values based on each model.

C. Additional processing

This step is highly dependent on the specific application. In the case of the ventriculostomy simulator, the burr holes can be performed using Boolean operations in VTK instead of 3DS Max. Once the models have been obtained in the previous step, they will be displayed in 3D in Sensimmer. In this way, the user can determine both the entry point of the catheter and its direction within a familiar environment. From this data, the burr holes can be drilled as previously described and the final models obtained.

V. CONCLUSION

Extraction of accurate and efficient 3D models from medical images is a fundamental part of our Virtual Reality and Haptic simulator. Due to the large variability in different patients, this component is difficult to be completely automated. However, a reasonable degree of automation can be achieved using available software tools. In this work we have evaluated these tools, proved their suitability to solve specific problems, and outlined the steps to be followed in order to increase the degree of automation of the whole process.

ACKNOWLEDGMENT

The authors would like to thank Dr. Michael Lemole, Dr. Mateo Calderon, and Dr. Ali Alaraj from the Department of Neurosurgery, University of Illinois-Chicago. Not only did they provide the data used in this work, but also gave us invaluable help and assistance.

REFERENCES

- [1] C. Luciano, P. Banerjee, L. Florea, G. Dawe, "Design of the ImmersiveTouch™: A High-Performance Haptic Augmented VR System," Proceedings of Human-Computer Interaction (HCI) International Conf. Las Vegas, 2005.
- [2] P. Banerjee, C. Luciano, L. Florea, G. Dawe, et. al, Compact haptic and augmented virtual reality device. U.S. Provisional Patent Application No. 60/646,837, March 2005, U.S. Patent Application, January 2006.
- [3] C. Luciano, P. Banerjee, G.M. Lemole, F. Charbel, "Second Generation Haptic Ventriculostomy Simulator Using the ImmersiveTouch™ System," Proceedings of 14th Medicine Meets Virtual Reality, J.D. Westwood et al. (Eds.), IOSPress, pp. 343-348, 2006.
- [4] P. Banerjee, F. Charbel, "On-Demand High Fidelity Neurosurgical Procedure Simulator Prototype at University of Illinois using Virtual Reality and Haptics," Accreditation Council for Graduate Medical Education (ACGME) Bulletin, September 2006; p. 20-21.
- [5] The Visualization Toolkit. Available: <http://www.kitware.com>
- [6] The Insight Segmentation and Registration Toolkit. Available: <http://www.itk.org/>
- [7] ITK-SNAP. Available: <http://www.itksnap.org/>
- [8] W. Schroeder, K. Martin, B. Lorensen, *The Visualization Toolkit An Object-Oriented Approach to 3D Graphics*, Kitware, Inc. , 2006.
- [9] The Visible Human Project. Available: http://www.nlm.nih.gov/research/visible/visible_human.html
- [10] M. Kass, A. Witkin, D. Terzopoulos, "Snakes: Active contour models," International Journal of Computer Vision, 1(4), 1987.
- [11] P. Yushkevich, J. Piven, H. Cody, S. Ho, G. Gerig, "Geodesic Snakes for User-Guided Segmentation of 3-D Anatomical Objects: Significantly Improved Efficiency and Reliability," Jan. 2005. Available: <http://www.itksnap.org/~paul/files/docs/yushkevich05snap.pdf>
- [12] V. Caselles, R. Kimmel, G. Sapiro, "Geodesic Active Contours," Int. J. Comput. Vis. 22, 61 - 79. 1997.
- [13] S. Zhu, A. Yuille, "Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation," Pattern Analysis and Machine Intelligence, IEEE Transactions on, Vol.18, Iss.9, Sep 1996, 884-900
- [14] P. Yushkevich, J. Piven, H. Hazlett, R. Smith, S. Ho, J. Gee, and G. Gerig, "User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability," Neuroimage 31 (2006) 1116-1128
- [15] J. Cebral, R. Löhner, "From medical images to anatomically accurate finite element grids," Int. J. Numer. Meth. in Engng 2001; 51:985-1008.
- [16] J. Du, X. Yang, Y. Du, "From Medical Images to Finite Grids System," Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai, China, Sept. 2005.
- [17] Y. Ito, P. Shum, A. Shih, B. Soni, and K. Nakahashi, "Robust generation of high-quality unstructured meshes on realistic biomedical geometry," Int. J. Numer. Meth. in Engng 2006; 65:943-973.
- [18] P. Young, G. Tabor, T. Collins, J. Richterova, E. Dejuniat, T. Beresford-West, "Automating the generation of 3D finite element models based on medical imaging data," Digital Human Modeling for Design and Engineering Conference, Lyon. July 2006.
- [19] J. Melonakos, R. Al-Hakim, J. Fallon, and A. Tannenbaum, "Knowledge-Based Segmentation of Brain MRI Scans Using the Insight Toolkit," The Insight Journal - ISC/NA-MIC/MICCAI Workshop on Open-Source Software , 2005
- [20] I. Wolf, M. Vetter, I. Wegner, M. Nolden, T. Böttger, M. Hastenteufel, M. Schöbinger, T. Kunert, H. Meinzer, "The Medical Imaging Interaction Toolkit (MITK) - a toolkit facilitating the creation of interactive software by extending VTK and ITK," Proc. SPIE Int. Soc. Opt. Eng. 5367, 16 (2004)
- [21] S. Prabhu, A. Zauner, M. Bullock, *Surgical Management of Traumatic Brain Injury*, Youmans Neurological Surgery, 5th. edition, WB Saunders, Philadelphia, 2004, pp. 5145-5180.