Interactive Stereoscopic Visualization of Very Large Visible Men on CAVE

Insung Ihm[1], Bumdae Lee[1], Joong-Youn Lee[2], Minsu Joh[2], and Sanghun Park[3]

[1] Department Computer Science, Sogang University
Seoul, Korea
{ihm,bumdai}@sogang.ac.kr
http://grmanet.sogang.ac.kr
[2] Supercomputing Center, KISTI
Deajeon, Korea
{msjoh,jylee}@kisti.re.kr
http://www.hpcnet.ne.kr
[3] School of Comp. & Info. Comm. Engineering, Catholic University of Daegu
Kyungbuk, Korea
mshpark@cataegu.ac.kr
http://graphics.cataegu.ac.kr

Abstract. We have developed an interactive visualization software for an immersive 3D virtual environment system, which generates stereoscopic images from huge human volume datasets in real-time using an improved volume visualization technique. Our software utilizes a CAVE system to reconstruct such huge volume datasets as the Visible Human and the Visible Korean Human, produced by NLM and KISTI, respectively. In stead of using the isosurfacing and 3D texture-mapping based volume rendering methods that easily slow down as data sizes increase, our system exploits a new image-based rendering technique to guarantee the effective interactive visualization.

1 Introduction

Volume visualization is an important area of scientific visualization. It deals with various techniques that are used for effectively generating meaningful and visual information from abstract and complex volume datasets, defined in three- or higher-dimensional space. It has been increasingly important in a variety of science and engineering fields, including meteorology, medical science, and computational fluid dynamics, and so on. Virtual reality is a research field focusing on techniques that help acquiring experiences in virtual world with visual, auditory and tactile senses. Recently, there have been many studies going on about visualization in immersive virtual environments. The CAVE (CAVE Automatic Virtual Environment) is a projection-based immersive virtual reality system in which users can interactively manipulate objects in stereoscopic images using various interface devices [4, 3]. This innovative virtual environment system has been used in many application fields such as military, aerospace, automobile, and medical science.

These days very large volume datasets are being generated frequently. With significant advances in computation, measurement, and storage technologies, giga- or terascale datasets have become increasingly commonplace. Since the ability to visualize such very large datasets in real-time is beyond the current capabilities of a single processor, it is required to develop highly scalable visualization techniques that fully harness the computing power of massively parallel computers. In particular, the NLM (National Library of Medicine) created CT, MRI, and color cryosection images of male and female human cadavers, called Visible Human (VH), in an effort to provide a complete digital atlas of the human body [11]. Similarly, the KISTI (Korea Institute of Science and Technology Information) led a project to generate CT, MRI, and RGB datasets of a Korean male cadaver, called Visible Korean Human (VKH) [8].

Table 1. Statistics on Visible Human and Visible Korean Human

Name	Type	Resolution	Voxel Size (Bytes)	Total Size (MBytes)
VH	CT (male)	$512 \times 512 \times 1867$	2	933.5
VKH	CT (male)	$512 \times 512 \times 1737$	2	868.5

The goal of this paper is to interactively visualize these visible men simultaneously in an immersive virtual environment. As shown in Table 1, their CT datasets are too bulky to interactively manipulate using the traditional visualization techniques. We found it critical to develop a new rendering scheme which is suitable for effective stereoscopic visualization. In this paper, we describe our interactive stereoscopic volume visualization system which allows one to render the visible men effectively in a virtual environment, by exploiting the computing power of high performance workstations with multipipe graphics hardware, and a new effective multi-pass shading algorithm. For the implementation of our visualization technique, a CAVE system at the KISTI Supercomputing Center was utilized [7].

This paper is organized as follows: In Section 2, we discuss the limitation of the traditional volume rendering techniques for very large volume data, and also propose new hardware accelerated shading and image-based volume rendering methods. Then our system and its performances are described in Section 3 and 4. Finally, we conclude this paper with directions for the future research in Section 5.

2 Image-Based Multi-Pass Shading Technique for Very Large Volume Data

2.1 Isosurfacing

One of the most common techniques for generating isosurfaces from volume data is to create an explicit polygonal representation using such algorithms as Marching Cubes [9]. The polygonal surfaces are then rendered with graphics hardware that are optimized for polygonal data. The Marching Cubes method easily generates a huge number of polygons, which often takes too much time to effectively visualize. Table 2 shows the number of triangles, generated from the VKH CT dataset, and the average frame rates on an SGI InfiniteReality3 graphics hardware. A decimation technique was applied to the original isosurfaces to enhance the rendering speeds.

Note that a pair of images, left and right, must be displayed simultaneously for a stereoscopic view. Hence, the actual frame rates reduce to half the figures in the table when the polygonal rendering technique is applied to our visualization system. A close examination of displayed images in the CAVE system reveals that, for the quality which is appropriate for effective immersive visualization, we need to use the polygonal models, generated using decimation ratio of at least 15%. However, the 1,311K triangles of the models are too many to visualize stereoscopically on the available graphics hardware (frame rates = 1.2fps). While it is possible to visualize the 0.3% models stereoscopically in real-time in the CAVE (frame rates = 44.6fps), the rendering quality is too poor to use (See Fig. 5(e)).

Table 2. Statistics on produced isosurfaces of the VKH dataset (SKIN)

	Decimation Ratios (%)									
	100	15	10	5	4	3	2	1	0.5	0.3
# of Tri.'s (*thousands*)	8,740	1,311	874	437	350	262	175	87	44	26
Frame Rates (*fps*)	-	1.2	1.8	3.5	4.1	5.6	7.5	15.0	22.5	44.6

2.2 Texture-Based Volume Rendering

Direct volume rendering is an excellent technique in scientific visualization. It has been used frequently in visualizing a wide range of volumetric datasets. When data is larger than memory space, software approaches are usually limited, and far from interactive due to the tremendous requirements of computation and bandwidth. As an alternative, the use of 3D texture mapping hardware has been recognized as an efficient acceleration technique to achieve real-time frame rates for reasonably sized datasets [13, 12, 5, 10, 6]. Since the amount of available texture memory is usually small compared to the size of volumes to be visualized, a clever strategy must be designed to develop an effective hardware-assisted volume visualization.

Multi-Pass Shading Technique In this work, we have designed a new multi-pass volume shading method which utilizes the optimized 2D texture-mapping and color-blending hardware. Unlike such recent commodity graphics processors as nVIDIA GeForce4 and ATI Radeon 8500 which offer user-programmable shader technologies, the SGI InfiniteReality3 hardware is based on a fixed-function pipeline. Hence, we had to develop a multi-pass shading technique for effective visualization in the CAVE system. Basically, the method evaluates the Phong's illumination model that has been used for realistic image synthesis. Our method was designed to fully utilize the SGI's graphics hardware for the fast computation of the Phong's model for the volumetric data. It consists of two steps: For a given set of viewing parameters, 2D normal vector images N are created in the pre-processing step. Then the final images are rendered quickly as described below. The calculations for given rendering parameters, such as ambient γ, diffuse δ, specular ε components, specular exponent n_s, light source direction L, and halfway vector H, are performed via the fast 2D texture mapping and color matrix operations. The Phong's illumination model implemented in our technique is given by:

$$C = M_{ad} \cdot N + \varepsilon (M_s \cdot N)^{n_s}$$

The above equation can be represented in the following matrix form:

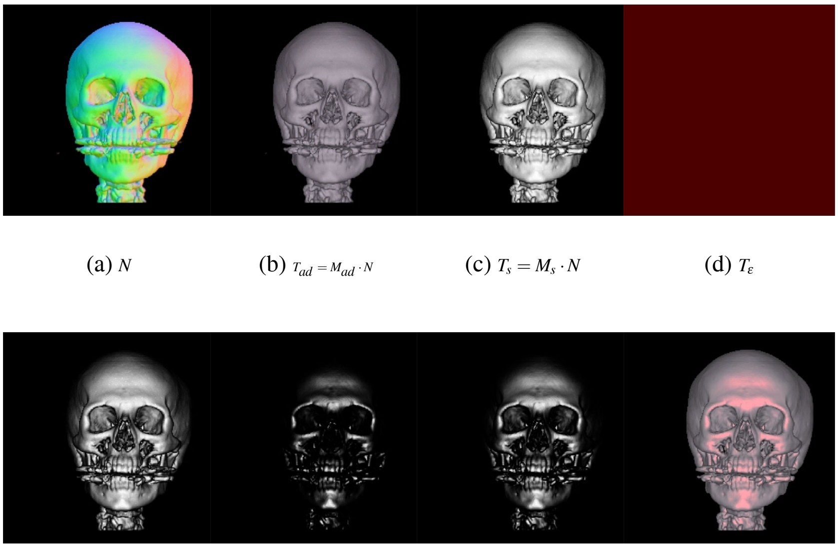
$$\begin{pmatrix} C_r \\ C_g \\ C_b \\ 1 \end{pmatrix} = \begin{pmatrix} \delta_r L_x & \delta_r L_y & \delta_r L_z & \gamma_r \\ \delta_g L_x & \delta_g L_y & \delta_g L_z & \gamma_g \\ \delta_b L_x & \delta_b L_y & \delta_b L_z & \gamma_b \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} N_x \\ N_y \\ N_z \\ 1 \end{pmatrix} + \begin{pmatrix} \varepsilon_r \\ \varepsilon_g \\ \varepsilon_b \\ 1 \end{pmatrix} \left[\begin{pmatrix} H_x & H_y & H_z & 0 \\ H_x & H_y & H_z & 0 \\ H_x & H_y & H_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} N_x \\ N_y \\ N_z \\ 1 \end{pmatrix} \right]^{n_s}$$

The multi-pass algorithm for real-time volume shading is summarized as follows:

1. Step I (Pre-processing)
   - 1-a. Create 2D normal images N by projecting and blending normal vector data stored as 3D volume textures.
2. Step II
   - 2-a. Create ambient and diffuse reflection texture T_{ad} by computing $M_{ad} \cdot N$ using color matrix operations.
   - 2-b. Create specular reflection texture T_s by computing $M_s \cdot N$ using color matrix operations.
   - 2-c. Set specular component texture T_ε using ε.
   - 2-d. Draw T_s into frame buffer repeatedly n_s times.
   - 2-e. Composite T_ε with frame buffer image.
   - 2-f. Generate the final image by blending T_{ad} with frame buffer image.

Fig. 1 illustrates the generated images through each step of the multi-pass shading algorithm. The second step can be computed very quickly on the fly since it is based on the simple and fast 2D texture mapping and color matrix operations. On the other hand, the first step takes most of the computing time since the combination of 3D texture mapping and color composition is relatively expensive. We found that the first step in the shading technique can not be computed fast enough when the datasets become large as in our application.

Image-Based Volume Rendering The 3D texture-mapping based volume rendering techniques are usually much faster than the traditional software-based direct volume rendering algorithms if the entire volume data can be loaded in texture memory. However, when we attempt to visualize volume data larger than the amount of available texture memory, inevitable texture swapping hinders real-time performances. In particular, it is quite difficult to visualize such large data as VH and VKH in real-time on the



(a) N (b) $T_{ad} = M_{ad} \cdot N$ (c) $T_s = M_s \cdot N$ (d) T_ε

(e) $T_s^{n_s} (n_s = 2)$ (f) $T_s^{n_s} (n_s = 10)$ (g) $T_\varepsilon \cdot T_s^{n_s}$ (h) $T_{ad} + T_\varepsilon \cdot T_s^{n_s}$

Fig. 1. Intermediate and final images generated by the multi-pass shading technique

SGI InfiniteReality3 using a 3D texture-based rendering method (See Table 3). Also, 3D texturing hardware with limited precisions makes it difficult to create normal images of high quality comparable to those produced in ray-casting. As we mentioned, the first step of the multi-pass shading method includes expensive 3D texture mapping and composition operations to generate 2D normal images.

In this work, we propose to employ an image-based rendering technique for real-time stereoscopic display. In contrast to the traditional 3D model-based rendering, image-based rendering techniques rely primarily on the acquired/renderd input images to produce new, virtual views. In our visualization system, the high resolution 2D normal image textures are prepared for all possible viewing directions using a modified ray-casting software in the preprocessing stage. Then, real-time visualization is achieved using both image-based rendering and multi-pass shading techniques. Our image-based volume rendering scheme is different from traditional image-based rendering in that the prepared images through pre-processing are not final rendered images but 2D normal textures. Hence, we do not have to generate new image sets every time rendering parameters such as the positions and colors of light sources, and the material colors of objects, are changed. This method is not only able to visualize huge volume datasets very fast but also to produce high quality images since ray-casting, known to generate the highest quality images, is used to create 2D normal textures.

2.3 Parallel Volume Ray-Casting

The parallel volume ray-casting technique is used in our system for two purposes. The first usage is to generate 2D normal textures of high quality in the pre-processing step. When a normal image is rendered with only one sample per pixel using the limited precision offered by graphics hardware, various aliasing artifacts may occur. To cope with such aliases, our system computes normal images in full precisions using the jittering technique.

The parallel ray-casting is also used for interactively generating high resolution snapshot images during stereoscopic visualization in the CAVE. For this application, we have implemented an efficient parallel volume ray-casting method optimized for the SGI Onyx multi-processor system. It is combined with the image-based volume visualization scheme to allow users to render high quality images which are difficult to produce with the image-based rendering technique. Fig. 2 shows some example images with different viewing parameters. Currently, our parallel ray-casting module does not offer real-time frame rates if the data sizes are large. However, the timing performance is expected to improve when the parallel computing power of all the available 20 processors of our system are fully exploited.

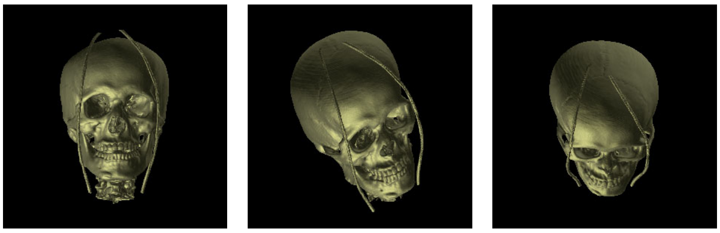


Fig. 2. Parallel ray-casting images of the VH head

3 Immersive Stereoscopic Visualization

3.1 Stereoscopic Display

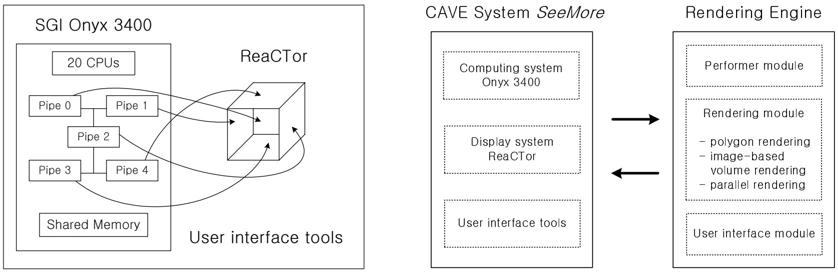
The stereoscopic images are produced by rendering two images for the left and right eye positions of viewpoint, and displaying them simultaneously. There are two frequently used methods for generating stereoscopic images: toe-in and off-axis [1]. In the toe-in method, each virtual camera located at the left and right eye positions points to a single focal point. Toe-in is usually identical to the methods that involve a rotation of the scene. This method can generate stereoscopic images but it often causes discomfort because of the vertical parallax. On the other hand, the off-axis method is a technique in which the directions of camera are made parallel. This method is more accurate than the toe-in method. The eyes may feel more comfortable since vertical parallax is never generated. It is, however, more expensive because the viewing frustum need to be modified for each of the eye points.

In our stereoscopic display system, both toe-in and off-axis methods were implemented. In the pre-processing stage, the virtual camera positions of ray-caster are moved around a circumference of object in volume space. The off-axis is used to set up view frustums for interactive stereoscopic display. We experimentally found the proper angles between left and right eyes at which human visual system perceives virtual depth comfortably. It turned out that the best angle for our visualization environment are three degrees. Since the texture images are prepared with very dense intervals of angles, the problem of the toe-in method are negligible.

3.2 Our CAVE System

It is known that there are more than 100 CAVE systems over the world, where volume visualization systems are being actively developed. The visualization systems directly related to ours are as follows: Zhang et al. presented a virtual reality environment using the CAVE to visualize tensor-valued volumetric datasets acquired from DT-MRI (Diffusion Tensor Magnetic Resonance Imaging) [14]. To visualize neural structures contained in DT-MRI data, they generate geometric models of stream tubes and surfaces from input data. A subset of the possible geometric modes are selected and decimated for interactive display. Boyles and Fang implemented an immersive environment, called 3DIVE, for interactive volume visualization and exploration inside the CAVE [2]. They use a 3D texture-mapping based rendering technique, and obtained about 6-12 frames per second.

The Supercomputing Center at KISTI is equipped with a virtual system, called SeeMore, which is made of a CAVE system with five projectors and an SGI Onyx3400 system with five InfiniteReality3 graphics pipes and 20 MIPS R12000 processors. (See Fig. 3(a) for the SeeMore system overview) The computing system has 6 GBytes of shared memory and 256 MBytes of texture memory. The rendering for each wall of the CAVE is done by a separate process on a dedicated graphics pipe. When a stereoscopic display is desired, each process calls the application's display function twice per frame. Users can interact in the virtual world using the provided tracker and controllers.



(a) SeeMore system of KISTI (b) Visualization System Structure

Fig. 3. Hardware and software system overview

4 Rendering Engine of Visualization System

The rendering engine of our visualization system is made of three core parts: Rendering module, Performer module, and User interface module (See Fig. 3(b)). Of course, the most important part is the rendering module to which our image-based multi-pass volume shading technique is applied. The rendering module must produce a stereoscopic pair for each new viewpoint, detected by the tracker, which requires twice the frame rate as a monocular viewpoint. In order to achieve target frame rates, we have optimized the rendering module so that it can fully exploit the resources of the Onyx system.

For the experimentation, we have generated 2D normal images of resolution 256×1024 for coronal view of the visible human datasets. The entire size of normal images amounts to 360Mbytes per dataset when normals are taken for every degree of viewpoint. Note that each graphics pipe of our Onyx2400 InfiniteReality3 hardware is equipped with 256 Mbytes of texture memory. Texture swapping may deteriorate the rendering performance as more sets of normal images are loaded.

Table 3 indicates the timing performance of the rendering module, in which a frame includes a pair of left and right images. As shown in the table, our rendering technique turns out to be very appropriate for the stereoscopic display. When only VKH is visualized (Image-based volume rendering VKH), we achieve about 23 (stereo) frames per second. Even when VKH and VH are loaded into texture memory (720Mbytes in total) simultaneously, it still visualizes the visible men at the interactive frame rates (about 15fps). Note that the interactivity is greatly affected by the size of data loaded into texture memory. When only normal images, for instance, corresponding to every other viewpoints are loaded, the texture memory requirement becomes relieved. In this case, the frame rates increase although some flickering may occur due to the missing images.

Table 3. Comparison of rendering speeds in stereo mode

Rendering Methods	Data	Frame Rates (*fps*)
Polygonal rendering	Decimated VKH (15%)	0.61
	Decimated VKH (0.3%)	22.31
3D Texture Mapping	VKH	0.88
Image-based volume rendering	VKH	22.52
	VKH + VH	14.99

On the other hand, we find out that the other two rendering methods are not suitable for producing stereoscopic images for the visible human datasets. In measuring the timing performance of 3D texture-mapping based technique, only the upper half of the VKH dataset ($512 \times 512 \times 700$, 700Mbytes) was loaded as texture data due to the limitation of the available texture memory. As indicated in the timing result (3D Texture Mapping), it is too slow for interactive stereoscopic visualization. We also tested with isosurfaces produced with various decimation parameters (Polygonal rendering). We could achieve the target frame rate when the original isosurfaces of VHK was decimated with rate 0.003. However, the quality of rendered images was not appropriate for visualization as shown in Fig. 5(e). The image quality becomes comparable to that of our image-based multi-pass shading technique when 10 to 15% of polygons were used. Unfortunately, they were too many for a single pipe to render interactively (Currently, a single pipe is dedicated to each wall in our CAVE system.) Fig. 4 demonstrates some snapshots of volume visualization of the two visible men in the immersive environment of our SeeMore CAVE system.

Fig. 4. Snapshots of stereoscopic visualization in the SeeMore system

5 Conclusion and Future Work

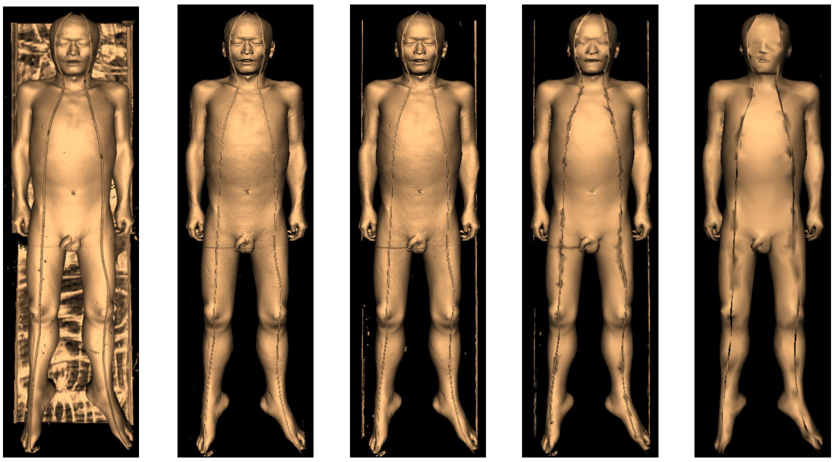
In this paper, we have developed an interactive visualization technique which allows one to visualize large volume datasets stereoscopically in an immersive virtual environment. Through the proposed image-based multi-pass volume shading technique, we were able to interactively visualize both Visible Human and Visible Korean Human in the CAVE system. Although the SGI Onyx system rendered the volume datasets successfully, we are currently investigating PC clusters as a replacement for more effective visualization. We expect that the programmable shading technologies of the commodity graphics hardware will improve the visualization quality and performance drastically.

Acknowledgement

This work was supported by grant No. R01-2002-000-00512-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

References

1. P. Bourke. Calculating stereo pairs. In *URL*. http://astronomy.swin.edu.au/~pbourke/stereographics/stereorender, 2002.
2. M. Boyles and S. Fang. 3DIVE: An immersive environment for interative volume data exploration. *Proceedings of CAD and Graphics 2001*, pages 573–580, China, August 2001.
3. C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. *Proceedings of ACM SIGGRAPH '93*, pages 135–142, August 1993.



(a) Ours (b) 15% (c) 8% (d) 1% (e) 0.3%

Fig. 5. Comparison of image qualities for VKH: (a) image-based volume rendering, (b) to (e) polygonal rendering of decimated isosurfaces

4. C. Cruz-Neira, D. J. Sandin, T. A. DeFanti, R. Kenyon, and J. Hart. The CAVE: Audio visual experience automatic virtual environment. *Proceedings of ACM SIGGRAPH '92*, pages 65–72, June 1992.
5. F. Dachille, K. Kreeger, B. Chen, I. Bitter, and A. Kaufman. High-quality volume rendering using texture mapping hardware. *Proceedings of Eurographics/SIGGRAPH workshop on graphics hardware '98*, pages 69–76, Lisbon, Portugal, 1998.
6. K. Engle, M. Kraus, and T. Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. *Proceedings of Eurographics/SIGGRAPH Workshop on Graphics Hardware 2001*, pages 109–116, 2001.
7. KISTI. SeeMore. In *URL*. http://www.hpcnet.ne.kr/vis/hw.htm, 2002.
8. KISTI. Visible Korean Human. In *URL*. http://vkh3.kisti.re.kr, 2002.
9. W. Lorensen. Marching through the visible man. *Proceedings of Visualization '95*, pages 368–373, Atlanta, October 1995.
10. M. Meißner, U. Hoffmann, and W. Straßer. Enabling classification and shading for 3D texture mapping based volume rendering using OpenGL and extensions. *Proceedings of IEEE Visualization '99*, pages 207–214, San Francisco, October 1999.
11. NLM. Visible Human Project. In *URL*. http://www.nlm.nih.gov/research/visible, 2001.
12. R. Westermann and T. Ertl. Efficiently using graphics hardware in volume rendering applications. *Proceedings of SIGGRAPH '98*, pages 169–177, Orlando, August 1998.
13. O. Wilson, A. Van Gelder, and J. Wilhelms. Direct volume rendering via 3D textures. Technical Report UCSC-CRL-9419, University of California, Santa Cruz, 1994.
14. S. Zhang, C. Demiralp, D. F. Keefe, M. DaSilva, D. H. Laidlaw, B. D. Greenberg, P. J. Basser, C. Pierpaoli, E. A. Chiocca, and T. S. Deisboeck. An immersive virtual environment for DT-MRI volume visualization applications: a case study. *Proceedings of IEEE Visualization 2001*, pages 437–440, San Diego, October 2001.