Interactive Volume Rendering for Virtual Colonoscopy

Suya You*, Lichan Hong*, Ming Wan*, Kittiboon Junyaprasert*, Arie Kaufman*,
Shigeru Muraki*, Yong Zhou*, Mark Wax† and Zhengrong Liang‡

Center for Visual Computing
and Departments of Computer Science and Radiology
State University of New York at Stony Brook
Stony Brook, NY 11794-4400

Abstract

3D virtual colonoscopy has recently been proposed as a non-invasive alternative procedure for the visualization of the human colon. Surface rendering is sufficient for implementing such a procedure to obtain an overview of the interior surface of the colon at interactive rendering speeds. Unfortunately, physicians can not use it to explore tissues beneath the surface to differentiate between benign and malignant structures. In this paper, we present a direct volume rendering approach based on perspective ray casting, as a supplement to the surface navigation. To accelerate the rendering speed, surface-assistant techniques are used to adapt the resampling rates by skipping the empty space inside the colon. In addition, a parallel version of the algorithm has been implemented on a shared-memory multiprocessing architecture. Experiments have been conducted on both simulation and patient data sets.

CR Categories: I.3.3 [Picture/Image Generation]: Display Algorithms; I.3.5 [Computational Geometry and Object Modeling]: Physically Based Modeling; I.3.6 [Methodologies and Techniques]: Interaction Techniques; I.3.7 [Three-Dimensional Graphics and Realism]: Hidden Line/Surface Removal; I.3.8 [Applications]; I.4.10 [Image Processing]: Image Representation—Hierarchical, Volumetric;

Keywords: Virtual Colonoscopy, Endoscopy, Visibility, Interactive Navigation, Volume Rendering, Surface Rendering, Parallel Processing, Virtual Environment

1 Introduction

Optical colonoscopy is a commonly used effective diagnostic and surgical tool in medical clinics, to assist the physician in viewing and examining the inner mucosal surface of the human colon. However, it suffers from problems such as patient discomfort, high cost, sedation, risk of perforation, and a limited range of exploration.

The *virtual colonoscopy* technique [1, 2] has recently been proposed as an alternative procedure for imaging and examining the interior structures of colon. The system takes a 3D radiological volumetric image of the patient's abdomen, acquired with a helical CT (computed tomography) scanner as input. After the volume reconstruction and colon region extraction, the 3D volumetric image serves as a virtual environment of the interior colon, whereupon the user can navigate and probe interactively the virtual colon.

Substantial progress has been made at SUNY Stony Brook towards such a system. In addition to the automatic plan navigation technique of our early work [1], we have implemented a prototype system which can achieve interactive navigation and visualization by using surface rendering techniques. A hardware-assisted visibility algorithm has been proposed to achieve adequate interactive rendering speed, and a physically-based mechanism is designed to accomplish guided navigation inside of the virtual colon world while avoiding collision with the surface [2].

Using such a system, the physician can perform a conceivable exploration of the colon interior structures just as in the optical colonoscopy, yet avoiding most of the problems mentioned above. However, since only the interior surface of colon is visible, the diagnostic capability of our virtual colonoscopy is limited. In order to remove the guesswork involved in diagnosing cancerous polyps, we have developed a novel feature of direct volume rendering which can visualize the detailed structure of the possible abnormality, and even the tissues beneath the colon surface.

In the following, Section 2 gives an overview of our interactive virtual colonoscopy prototype system, and then in Section 3 we focus on the direct volume rendering technique for the system. In Section 4, some experimental results of our approach on both simulation and patient data sets are presented.

2 System Overview

Our interactive virtual colonoscopy takes 3D volumetric data acquired with commonly available medical imaging modalities, such as CT, and regard it as a virtual environment which

*Email:{suyay|lichan|mwan|junyapra|ari|muraki|yzhou}@cs.sunysb.edu
†Email:mwax@radiol.som.sunysb.edu
‡Email:jzl@clio.rad.sunysb.edu

is a clone of the real colon. The user can then interactively navigate inside the virtual colon world in a manner similar to — yet more flexible than — the optical colonoscopy procedure. To perform this procedure, the following three stages are involved:

- Data acquisition: a set of 2D slice images which covers the entire range of the colon is captured by a helical CT scanner. In order to produce better density contrast, the patient's colon must be cleansed and inflated with air before the scan, as traditionally done in optical colonoscopy.

- Pre-processing: the acquired 2D slices are reconstructed into a 3D volumetric data set and the colonic surface is extracted from the volume data using the Marching Cubes algorithm [3]. At the same time, the *center-line* (or *skeleton*) and the *potential field* (as a part of the physically-based mechanism for interactive navigation) of colon are determined for the navigation stage.

- Navigation: Our system permits two kinds of dynamic visualization modes: planned navigation, in which the view moves automatically along the *center-line* without user interaction; and the interactive navigation mode that allows the user to manipulate the camera to interactively explore structures as desired. While planned navigation can provide a quick general overview of the colon, it is rather limited because no user interaction is possible. A novel physically-based mechanism has been developed to interactively control the moving view, without colliding with the surface of the colon. An efficient hardware-assisted visibility algorithm is also employed to cull invisible surfaces, thus supporting interactive rendering [2].

The system has been implemented and tested on several simulation and patient data sets including a plastic pipe, the Visible Male Data, and patients' scanned data sets obtained from Stony Brook University Hospital. Figure 1 shows a navigation frame from our system for a patient's data set. The green line represents the pre-defined navigation path. As we can see, a clear view of the inner surface can be obtained to depict the interior structures. Currently, the system has achieved an adequate interactive speed for image size of 512×512 on a single processor. The encouraging results have shown that such a non-invasive procedure can potentially provide an alternative technique to assist the physician in examining the inner structures of the human colon, which may improve the diagnostic sensitivity and specificity with fewer complications, while serving as a mass screening procedure for certain populations.

3 Direct Volume Rendering

3.1 Motivation

Several existing systems [2, 4] have employed a surface rendering technique for image generation. That is, the acquired volume data must first be pre-segmented to extract the colonic surface, and then pass the extracted surfaces (triangles) to the graphic pipeline to render the isosurface triangles during the navigation. The navigation is then used to get an overview of the inner surface of the colon to find possible abnormalities on the surface, during which a comfortable and natural speed of image generation on-the-fly is particularly important. Hence, the surface rendering approach is an appropriate technique in this intended application.

Once possible abnormalities are found in the navigation procedure, a detailed study and analysis of the tissues under the surface are necessary for the physician, but impossible to obtain in the surface navigation system. Fortunately, the direct volume rendering technique can meet such demand by directly mapping certain ranges of sample values of the original volume data to different colors and opacities, wherein the intermediate geometric representation is omitted. Specifically, perspective volume ray casting algorithm is employed in our system. It is composed of three steps [5]: traversing and resampling along the ray cast from each pixel of the image; assigning specified color and opacity to each sampling point; and compositing along the ray to obtain the pixel color. Since this procedure is repeated for every pixel in the image, it is very expensive in terms of both computational and memory costs, and is far from being real-time or interactive rendering on existing graphics workstations [6], especially for large data sets, such as $512 \times 512 \times 256$ CT data of the patient's abdomen in our case. Therefore, several strategies, such as *surface-assistant ray casting* and *parallel processing* are exploited to accelerate the volume rendering rate.

3.2 Surface-Assisted Ray Casting

In our system, volume rendering works along with surface rendering, and thus, the information provided by the surface rendering can be used to speed up the volume ray casting process. Recall the fact that the colon has a cavity structure with a thin surface, and there is almost nothing inside it. Thus, instead of sampling the whole sight ray for the ray accumulation, we can skip over those empty spaces and simply perform sampling in the neighbor of the colon surface.

To determine the bounds of the ray integral, we utilize the depth information produced by the surface navigation. In [2], an efficient visibility determination algorithm — hardware-assisted visibility — has been developed to support interactive surface rendering. This actually gives us a conservative estimation of the depth information at every given navigation site, which can be used to estimate the *hither* (front) bound of the ray integral. The *yon* (back) bound could be determined in two ways. The first one is that we can *a priori* define a length of the integral that enables the sampling to sufficiently cover the region of interest. This method is suitable for visualizing the information hidden just behind the surface, such as exploring the abnormal tissues underneath the surface of the colon. The other way is to simply terminate the sampling process as soon as the accumulated opacity reaches unity or a user-defined threshold for opacity, as we perform composition in a front-to-back order along each ray.

3.3 Parallelization

A commonly adopted strategy to speed up volume rendering is to employ parallel architectures and approaches [7, 8, 9, 10, 11]. Our surface-assisted ray casting algorithm has been parallelized on the Silicon Graphics Power Challenge, a bus-based shared-memory MIMD (Multiple Instruction, Multiple Data) machine. The shared-memory architecture permits straightforward implementation of our algorithm.

In general, there are two main issues that should be considered for designing a parallel algorithm for the shared-memory architecture: allocating the total computation into tasks for each processing element (PE), and choosing appropriate synchronization mechanisms for these tasks. Object-based partition and image-based partition are two commonly used strategies of task allocations for parallel volume rendering. In the object-based approaches, the volume data is partitioned into appropriate subvolumes and each PE is responsible for a subvolume for volume resample and composition. Since in this case, the image space is shared by all PEs and the partial results from each PE must be composited together to form the final image, explicit synchronization of the PEs is necessary to avoid erroneous composition. However, in the image-based algorithms the partition is performed directly in the image space. Each PE is responsible for compositing a specific portion of the image, and the shared data structure is the volume space. Thus, less synchronization is required for this type of partition. We, therefore, use the image-based partition strategy in our algorithm.

An important consideration for designing a parallel algorithm is how to choose the work units assigned to PEs, while minimizing load imbalances. In our ray casting algorithm, instead of sampling the whole ray, we skip over the empty space inside the colon. The sampling segment is almost the same for every ray, so the rays in a local block have very close approximation of the amount of work one has to perform for ray traversal. We, therefore, choose to use the uniform static partition technique, that is, the image is divided into equal sized rectangular blocks (4×4 size, for example for 16 PEs). Each pixel of a block is allocated to a PE for raycasting traversal and composition.

4 Experimental Results

The direct volume rendering algorithm has been implemented and integrated into our interactive virtual colonoscopy system, providing a superior supplement to the interactive surface navigation. The experiments here are conducted on a Silicon Graphics Power Challenge equipped with sixteen R10000 processors in a bus-based symmetric shared-memory configuration. The parallel code of the ray casting algorithm is implemented using the Silicon Graphics multithread support library based on the Sequent Computer System parallel primitives.

We have conducted experiments on simulation data, the Visible Male Data, and patients' colon data sets. Figure 2 shows the volume rendering result for a pipe simulation data. This simulation is based upon a CT scan of a plastic pipe of 20mm radius forming a volume of $512 \times 512 \times 107$. To simulate colonic polyps, we also attached three small rubber objects of size 7mm, 5mm, and 3mm to the inner surface of the pipe, which have been clearly depicted in the figure. Table 1 presents the measured rendering times of the simulation data with different numbers of processors and image sizes, showing linear scalability.

In the second experiment, we used a colon data set obtained from a patient at Stony Brook University Hospital. The objective of this study has been to employ the procedure of interactive virtual colonoscopy in order to detect the presence and characteristics of mucosal lesions measuring at least one centimeter in diameter, which are considered clinically significant since they have a high probability of being malignant. Prior to the interactive virtual colonoscopy, the patient went through a colonic cleansing routine similar to that required for a barium enema or colonoscopy, so that retained stool could be differentiated from polyps or tumors. After the patient arrived at the CT scanning suite, air was pumped into the colon to distend the colon, and then 358 slices of high-resolution (512×512) abdomen images were produced by a GE HighSpeed CT in the helical mode.

To visualize the tissues underneath the colonic surface, we have chosen different opacities and colors for different tissues. This feature of volume rendering is superior to the surface-based techniques, allowing the physician to confirm possible abnormalities without physical biopsy and surgery. Figure 3a shows a volume rendering image inside the patient's colon. The measured rendering times are presented in Table 2 with different image sizes and different numbers of processors, showing linear scalability.

For comparison, Figure 3b presents the corresponding surface view generated by the surface rendering technique. As we can see, the volume rendering view provides more information than that of the surface rendering. To illustrate, a small gully has been found in the volume rendering view (Figure 3a, enclosed with a box), but not in the surface rendering view.

In this patient's data set, there is an abnormality in the colonic surface, successfully detected by our surface navigation procedure and confirmed with the volume rendering view. Figure 4 presents the close views of the abnormality with both volume rendering and surface rendering.

5 Conclusions

Computerized virtual colonoscopy can provide an effective alternative for clinical diagnostics, to assist the physician in

Table 1: *Volume rendering time (in seconds) for the pipe data on SGI Power Challenge (NP: Number of processors; RS: Resolution of image)*

Data	*Pipe*		
NP\RS	512×512	256×256	128×128
1	12.39	3.16	0.79
4	3.14	0.79	0.20
9	1.40	0.36	0.09
16	0.78	0.19	0.05

Table 2: *Volume rendering time (in seconds) for the patient's data set SGI Power Challenge (NP: Number of processors; RS: Resolution of image)*

Data	*Patient*		
NP\RS	512×512	256×256	128×128
1	16.05	4.04	1.01
4	4.05	1.01	0.26
9	1.84	0.48	0.12
16	1.01	0.25	0.06

performing a conceivable exploration of human organs. It has the multiple advantages of being non-invasive and cost-effective, while considering patient-comfort. In order to remove guesswork involved in diagnosing cancerous polyps, the volume rendering algorithm is adopted as a supplement to the interactive surface navigation. To speed up the volume rendering process, we have presented surface-assistant techniques to accelerate rendering rate by skipping the empty space inside the colon. We have also explored the hardware acceleration technique by using the shared-memory based parallelization, resulting in linear speedup.

We have achieved what can be expected from direct volume rendering. Currently, we are exploring additional volume acceleration techniques, such as the multi-resolution method and visual temporal coherence, expecting to achieve further speed-up of the volume rendering. Furthermore, the system is being extended to more complex applications, including exploration of branching structures. At that time, direct volume rendering will become not only a supplementary tool, but also a vital component of this system.

6 Acknowledgements

This work has been supported by grants from the Center for Biotechnology and E-Z-EM Inc. The Visible Male data set is courtesy of the National Library of Medicine and the Visible Human Project. The pipe and patient data sets were provided by Stony Brook University Hospital. We would like to acknowledge the collaboration and help of Dr. Ajay Viswambharan from the Radiology Department.

References

[1] L. Hong, A. Kaufman, Y. Wei, A. Viswambharn, M. Wax, and Z. Liang. "3D Virtual Colonoscopy". *Proc. 1995 Symposium on Biomedical Visualization*, 1995, 26-32.

[2] L. Hong, S. Muraki, A. Kaufman, D. Bartz, and T. He. "Virtual Voyage: Interactive Navigation in the Human Colon". To appear in *Proc. SIGGRAPH '97*.

[3] W. Lorensen and H. Cline. "Marching Cubes: A High Resolution 3D Surface Construction Algorithm". *Computer Graphics (Proc. SIGGRAPH '87)*, 163-169.

[4] W. Lorensen, F. Jolesz, and R. Kikinis. "The Exploration of Cross-Sectional Data with a Virtual Endoscope". In R. Satava and K. Morgan (eds.), *Interactive Technology and the New Medical Paradigm for Health Care*, IOS Press, Washington D. C., 1995, 221-230.

[5] M. Levoy. "Display of Surface from Volume Data". *IEEE Computer Graphics and Applications*, Vol. 8, No. 5, May 1988, 29-37.

[6] A. Kaufman. *Volume Visualization*. IEEE Computer Society Process, Los Alamitos, CA, 1991.

[7] L. S. Haynes, R. L. Lau, D. P. Siewiorek, and D. W. Mizell. "A survey of highly parallel computing". *Computer*, Vol. 5, No. 1, Jan 1982, 9-27.

[8] U. Neumann. "Communication Costs for Parallel Volume-Rendering Algorithms". *IEEE Computer Graphics and Applications*, Vol. 14, No. 4, July 1994, 49-58.

[9] U. Neumann. "Interactive Volume Rendering on a Multicomputer". *Proc. 1992 Symposium on Interactive 3D Graphics*, Cambridge Mass., March 1992, 87-93.

[10] W. Hsu. "Segmented ray casting for data parallel volume rendering", *Proc. 1993 Parallel Symposium*, ACM Press, Oct. 1993, 7-14.

[11] J. Nieh, and M. Levoy. "Volume rendering on scalable shared-memory MIMD architecture". *Workshop on Volume Visualization*, ACM Press, Oct. 1992, 17-24.