A Block-Based Volume Rendering Scheme for the Spine Biopsy Simulator

Sungmin Kwon, Jin Kook Kim, and Jong Beom Ra
Dept. of Electrical Engineering, Korea Advanced Institute of Science and Technology
373-1, Kusongdong, Yusonggu, Taejon, Korea
jbra@ee.kaist.ac.kr

Abstract

Volume rendering is a powerful tool for visualizing sampled scalar values from 3D data without fitting geometric primitives to the data. However, the size of volume data is usually too big to handle in real time. Recently, various volume rendering algorithms have been proposed in order to reduce the rendering time. However, most of the proposed algorithms are not proper for large volume data, which is required in some applications such as the spine biopsy simulator. In this paper, we propose a block-based fast volume rendering algorithm using shear-warp factorization. The algorithm performs volume rendering by using information of organ segmentation as well as 3D volume data, which is a common environment in the spine biopsy simulator. The proposed block-based algorithm can reduce storage requirement and increase the rendering speed by treating the 3D data on a block-by-block basis. The proposed algorithm is evaluated by rendering 3D body images obtained from X-ray CT (computed tomograph) system.

1. Introduction

Medical engineering technology has kept a pace with remarkable development of electronic engineering. Such splendid development of medical engineering technology has introduced medical machinery such as XCT (X-ray computerized tomography) scanner, MRI (magnetic resonance imaging) scanner, and ultrasonic diagnosis apparatus. Those apparatuses can extract 2D medical images from the inner parts of the body without a surgical operation. However, 2D images are too flat to show observers the inner parts of the body intuitively. Therefore, many researchers engage themselves in making 3D images from a set of 2D images.

In order to achieve interactive rendering speed, most visualization techniques extract iso-surface data from 3D volume data and use specified hardware. But those surface rendering methods need enormous pre-processing time for fitting geometric primitives to the data and may lose the shape of volume data in the fuzzy boundary [1]. Furthermore, surface rendering methods must use an expensive 3D texture map device for optical effects like transparent boundary and for reality of surface.

In order to achieve accuracy and reality without specified hardware, volume rendering methods are being used [2-5]. Volume rendering is a technique for visualizing sampled scalar values from 3D data. However, the size of volume data is usually too big to handle in real time. Therefore, various volume rendering algorithms have been proposed in order to reduce the rendering time. One of them is the shear-warp factorization method [3]. In this method, by shearing 3D volume data and compositing voxels, an intermediate projected image is obtained. And then, the final projected image is obtained by 2D warping of the intermediate projected image. This method can provide interactive rendering speed without any specified hardware. However, as the volume data size increases, the larger storage device is needed. Also, rendering speed becomes slower.

In this paper, we propose a block-based shear-warp factorization scheme by using information of organ segmentation that is to be prepared in the spine biopsy simulator. The proposed method provides fast volume rendering, and requires a small size of storage. Also, the method is comparable with the previous fast shear-warp factorization method in terms of rendering speed.

2. The proposed scheme

2.1. Previous volume rendering method using shear-warp factorization

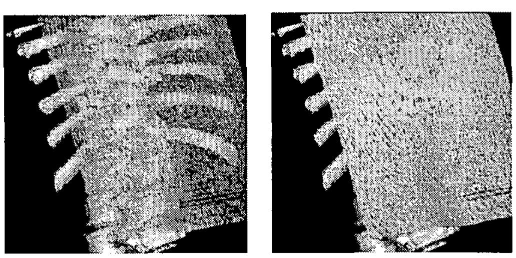
Usually, volume rendering has two bottlenecks: Finding interpolation points in the object space and interpolating using contiguous voxels. In the fast shear-warp factorization method, the space between rays corresponding to the final projected image pixels is adjusted in order that intersection points between a slice and rays are regularly located on the slice. Thereby, the complexity of interpolation can be reduced. This adjustment is called shearing. Then shearing and compositing produce an intermediate projected image.

Since the intermediate projected image is different from the final projected image, the intermediate projected image should be warped into the final projected image. Warping is considered a sort of the 2D interpolation techniques. Therefore, shear-warp factorization can be performed with two steps; adjusting space between rays for fast compositing and correcting the distorted projected image.

A run-length coding technique has been applied to volume data for fast rendering in the shear-warp factorization method [3]. In pre-processing, it is assumed that a user decides objects to be rendered and their opacity mapping functions. All voxels in volume data are scanned to select voxels having non-zero opacity, and produce a run-length code using them. In rendering processing, compositing and warping are carried out after run-length decoding. Therefore, when the user changes the opacity mapping function, the rendering time increases because non-zero opacity voxels should be encoded again by using a new opacity mapping function. However, it should be mentioned that its rendering speed is very fast for a fixed opacity mapping function.

2.2. Requirements in volume rendering for the spine biopsy simulator

Spine biopsy is to take tissue samples from a lesion inside the spine, by using a needle without surgical operation, in order to carry out pathological inspection. In performing spine biopsy, the needle should not touch other important organs near the spine, such as lung, kidney, nerves, vessels. Therefore, the spine and organs around the spine need to be rendered separately. And the opacity of the organ must be changed as the needle proceeds, because the region of interest varies depending on the location of the needle tip. Figure 1 shows rendering examples with the two different opacities of skin, where the opacity of spine is unchanged. Here, the rendering speed has to be fast enough so that the viewing angle and needle location can be changed interactively.

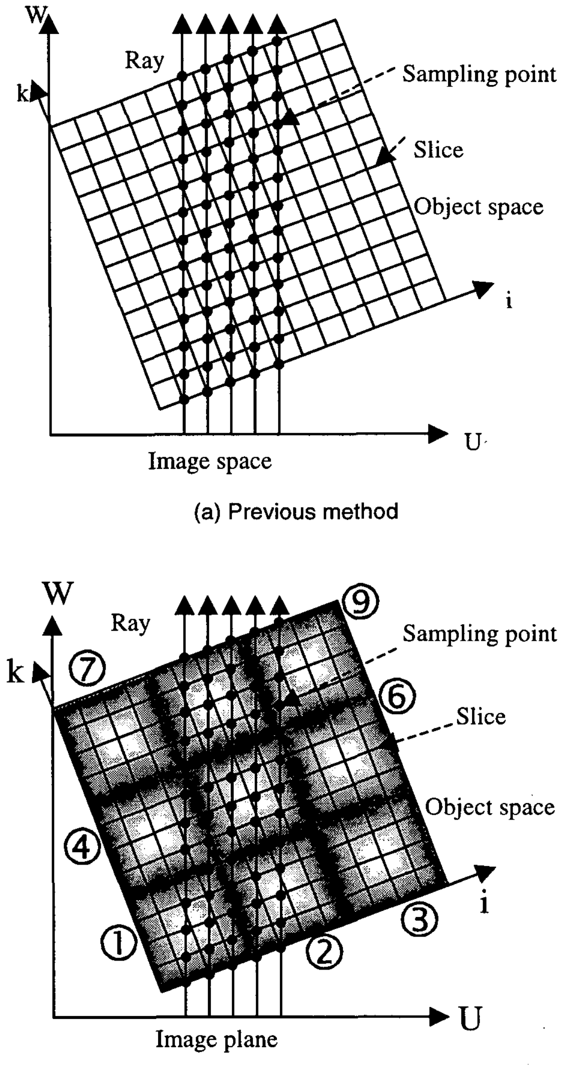


(a) Smaller opacity (b) Larger opacity

Figure 1. Opacity change in the skin.

2.3. The proposed algorithm

In this paper, we propose a block-based volume rendering method that uses shear-warp factorization and segmentation information. Unlike the previous non-block-based volume rendering method using shear-warp factorization, which suffers from heavy computational burden in case of the change of opacity mapping function, the proposed algorithm can change opacity easily by dividing volume data into small blocks without encoding. In addition, it can reduce rendering time by ignoring blocks not of interest and using information of organ segmentation. The proposed method is different from the



(a) Previous method

(b) Proposed method

Figure 2. Simplified 2D diagram to show the rendering process.

previous methods [3-5] in that it renders volume data in block-by-block order using segmentation information and the shear-warp factorization method. The segmented data is used for selecting region of interest, and a volume data is stored and rendered in block-by-block basis. The rendering order of blocks should be arranged according to the distance from the projection plane. The circled numbers of Figure 2(b) denote the rendering order of blocks.

Figure 3 shows the overall procedure of the proposed rendering scheme for a 3D volume data. The proposed algorithm is composed of two stages. One is pre-processing stage and the other is rendering processing stage.

In the pre-processing, volume data is stored in the form of small blocks, and each block has header bits indicating which organ is contained in the block from the segmentation information. In this step, all organs of interest are actually defined by a user, and the organs should be segmented previously. Figure 4(a) shows a slice out of 256^3-CT volume data, and Figure 4(b) shows the organ segmentation information. If a block has at least one organ, the block is stored and processed. Otherwise, the block is discarded and no operation is required. In Figure 3, the discarded block is shown as the empty block in the pre-processing stage.

In the rendering processing, blocks to be rendered are picked up using the block headers that have object (or organ) information. The rendering processing is composed of three steps. First, we choose objects to be rendered more specifically among the objects defined at the pre-processing stage. The blocks related to the objects to be rendered is re-determined and a queue of rendering blocks (QRB) is prepared. Second, the blocks in QRB are shaded and composited to make the intermediate projected image. And then, the intermediate projected image is warped into the final projected image. If we render other objects or re-allocate opacity mapping function for a certain object, the rendering processing should be carried out from the first step. But if we change only viewpoint for the rendering, the first step of the rendering processing can be avoided and faster rendering speed is achieved.

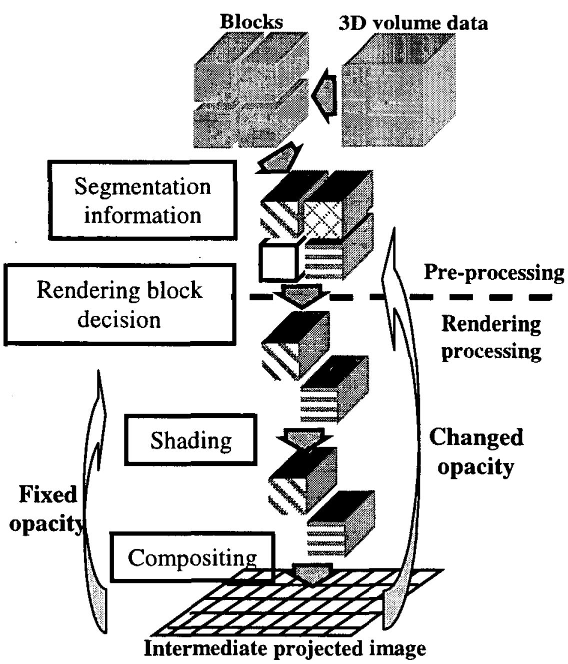
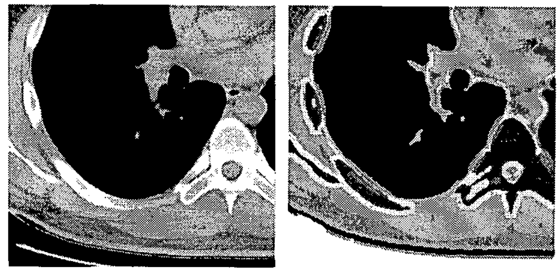


Figure 3. Overall block-diagram of the proposed algorithm.



(a) CT volume data (b) Segmentation information

Figure 4. Segmentation information for a CT image.

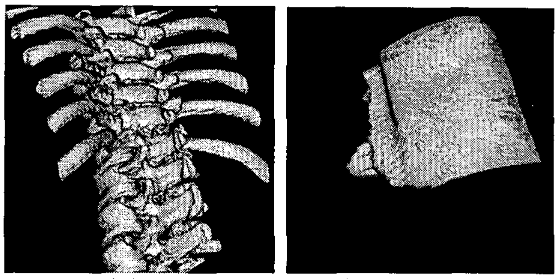
3. Simulation results

Simulation has been performed using a Silicon Graphics Onyx2 reality workstation, which has the R10000 250MHz CPU, primary cache of 32KByte, secondary cache of 1MByte, and 256MByte RAM.

3.1. Decision of block size

The block size for the proposed method was determined experimentally with an 8bit 256^3 skull CT data and 12bit 256^3 body CT data. The block size of $(N+1)^3$ is adopted instead of $(N)^3$ so that each block has overlapped pixels to remove blocking artifacts. Therefore, it may spend slightly more storage than the original volume data. Since the block data corresponding to the regions not of interest spends no storage, however, total storage requirement is reduced actually.

Table 1 shows the number of clocks required for compositing when all 256^3 voxels are rendered without voxel skipping. As the block becomes larger, the cache miss-rate increases because data in a block are usually not localized [3]. On the contrary, as the block becomes smaller, the number of blocks becomes dominant because



(a) Spine (b) Right lung

Figure 5. Rendering results.

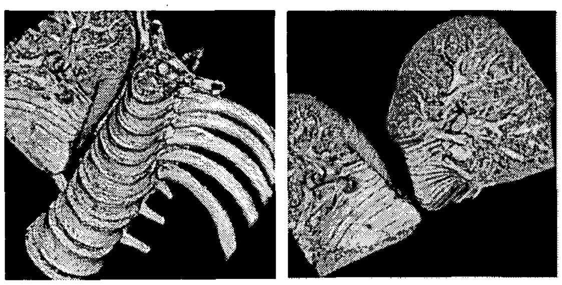


Figure 6. Rendering results for 2 objects.

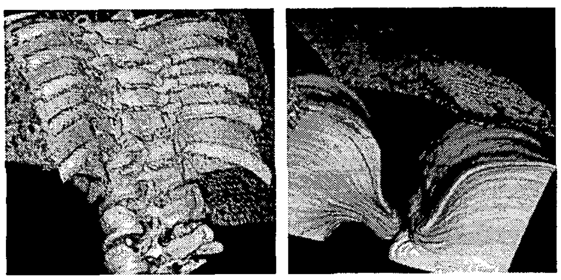


Figure 7. Rendering results for more than 3 objects.

many blocks must be handled for scanning the same number of voxels. Therefore, the block size should be determined within the range where both cache miss-rate and number of blocks are not dominant.

Table 2 shows more reliable results by introducing block skipping, early ray-termination, and image run-length coding. It is noted in the table that 9^3 is proper for the block size for compositing 256^3 volume data.

3.2. Performance of the proposed algorithm

For the performance test, 8 bit 256^3 skull CT data is used with the block size of 9^3. Performance is evaluated for two rendering methods; the existing method using run-length coding and the proposed method. As shown in Table 3, when the opacity mapping function changes, the proposed method is far faster than the previous method. When the opacity mapping function is fixed, however, the rendering time is similar. Therefore, the proposed method is especially proper for the spine biopsy simulator where the opacity mapping function is to be changed frequently.

Table 4 shows the rendering speed of the proposed method for different object selections. Figures 5-7 show several rendered images corresponding to Table 4.

4. Conclusions

We have introduced a block-based concept into the shear-warp factorization method. The proposed method can change opacity easily by dividing volume data into small blocks without encoding. In addition, it can reduce rendering time by ignoring blocks not of interest and using information of organ segmentation. The proposed rendering algorithm is especially proper for the spine biopsy simulator in which the viewpoint and opacity mapping function can be changed simultaneously, and information of organ segmentation is available.

References

[1] W. E. Loresen, "Marching cubes: A high resolution 3D surface construction algorithm," *Computer Graphics*, vol. 21, no. 4, pp. 163-169, July 1987.

[2] M. Levoy, "Display of surfaces from volume data," *IEEE Computer Graphics and Applications*, vol. 8, no. 3, pp. 29-37, May 1988.

[3] P. Lacroute, "Fast volume rendering using a shear-warp factorization of the viewing transformation," Ph.D. dissertation, Stanford Univ., 1995.

[4] D. Meagher, "Geometric modeling using octree encoding," *Computer Graphics and Image Processing*, vol. 19, pp. 129-147, 1982.

[5] M. Levoy, "Efficient ray tracing of volume data," *ACM Transactions on Graphics*, vol. 9, no. 3, pp. 245-261, July 1990.

Table 1. Block size versus the number of clocks for compositing total voxels.

Block size	Number of clocks for compositing
5^3	4402708
9^3	3709829
17^3	3551682
33^3	3565304
65^3	3589609
129^3	6516379

Table 2. Block size versus the number of clocks for compositing voxels to be rendered.

Block size	Block skipping	Block skipping + Early ray termination + Image run-length coding
	Number of clocks for compositing	Number of clocks for compositing
5^3	285440	114337
9^3	267785	94930
17^3	315889	122593
33^3	438222	221376
65^3	679213	
129^3	1745354	

Table 3. Proposed scheme versus shear-warp run-length coding scheme.

Time	Run-length coding	Proposed
Opacity change	51.84 sec	0.92 sec
Compositing	96 msec	116 msec

Table 4. Object rendering speed in the proposed method.

Objects	Rendering speed
Spine	5.0 frames/sec
Spine + Right lung	3.5 frames/sec
Spine + Left/right lungs	2.3 frames/sec
Spine + Left/right lungs + Skin	1.9 frames/sec