A method for interactive manipulation and animation of volumetric data

Yves Jean, Larry F. Hodges

Graphics, Visualization and Usability Center
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332

Roderic Pettigrew

Department of Radiology
Emory University School of Medicine
Atlanta, GA 30322

ABSTRACT

We outline an efficient method for visualizing and manipulating volumetric data, in particular, cardiac MRI data sets. The approach is designed to allow interactive manipulation and real-time animation of volumetric data sets. The underlying model provides an efficient graphical representation for interactive rendering while not eliminating data from the volume of interest. We believe this model to be a valuable medical imaging tool that is applicable to other volume rendering problems.

1. INTRODUCTION

The advent of medical scanning technologies such as Computer Tomography (CT) and Magnetic Resonance Imaging (MRI) provides clinicians with volumetric image data capturing physiologic and anatomic information. Currently, these images are either transferred to film or displayed on a monitor for viewing by a physician. When the tissue of interest encompasses a volume the number of tomographic images produced makes a concise integrated display difficult. Computer graphics and medical imaging researchers have addressed this problem by combining their respective backgrounds to develop specialized volume visualization techniques. In this paper we outline an approach to visualizing and manipulating volume data, in particular, four-dimensional cardiac MRI data sets. Motivated by the need to animate and interact with volume data we have created a volume rendering model that preserves and renders the volume data, displaying the surfaces and volumes of interest while maintaining the original grayscale and contrast. The model provides an effective solution to visualization problems involving interactivity and data integrity.

1.1 Previous work

Most previous volume rendering techniques can be classified as either surface or volume methods.[1,2] Surface techniques fit a geometric model to the volume data and render the model. Volume methods attempt to generate more faithful reproductions of the volume by simulating some form of chromatic light ray transformation through the data. The two techniques have different advantages and disadvantages. In general, surface modelling creates a smaller rendering load for the graphics pipeline, potentially leaving residual graphics rendering power for interactivity and animation, but at the risk of introducing artifacts and suppressing salient information.[3] In addition, surface-based techniques require that the surface of interest be detectable and mappable to the geometric model. Volume rendering algorithms model the data as opacities or densities by using ray casting techniques to simulate the spectral transformation of the ray through the volume.[4,5] While volume techniques are more powerful, they are also computationally expensive.

To be published in *The Proceedings of the VBC '92*

Interestingly, some of the volume techniques are similar to surface techniques in that they incorporate some type of surface extraction component. Levoy detects surfaces and composes the total color from evenly spaced samples along a ray through the volume.[5] Other researchers have produced similar as well as more robust work.[6] Drebin uses a material mixture model to classify voxels of multiple substance composition.[7] More sophisticated models and fitting techniques have been developed. They use more complex geometric models and/or constraints to dynamically deform the model to fit the surface data .[8]

There are three general categories of attributes that we wish to represent in medical image visualization: anatomical, physiological, and temporal. Anatomical information refers to the gross structures of the human body. Physiological attributes are represented by the actual grayscale texture data. In clinical settings such texture is the basis of diagnostic medicine. Texture captures both macro and micro features in the data. The extent of a texture can represent the size of a tissue region (macro) while the spectral quality captures the state (micro). Temporal attributes record the time-varying conditions of the data. Such attributes capture the change of various features over time, but more interestingly, capture the change of structure and texture over time (a requirement when analyzing 4D cardiac MRI data).

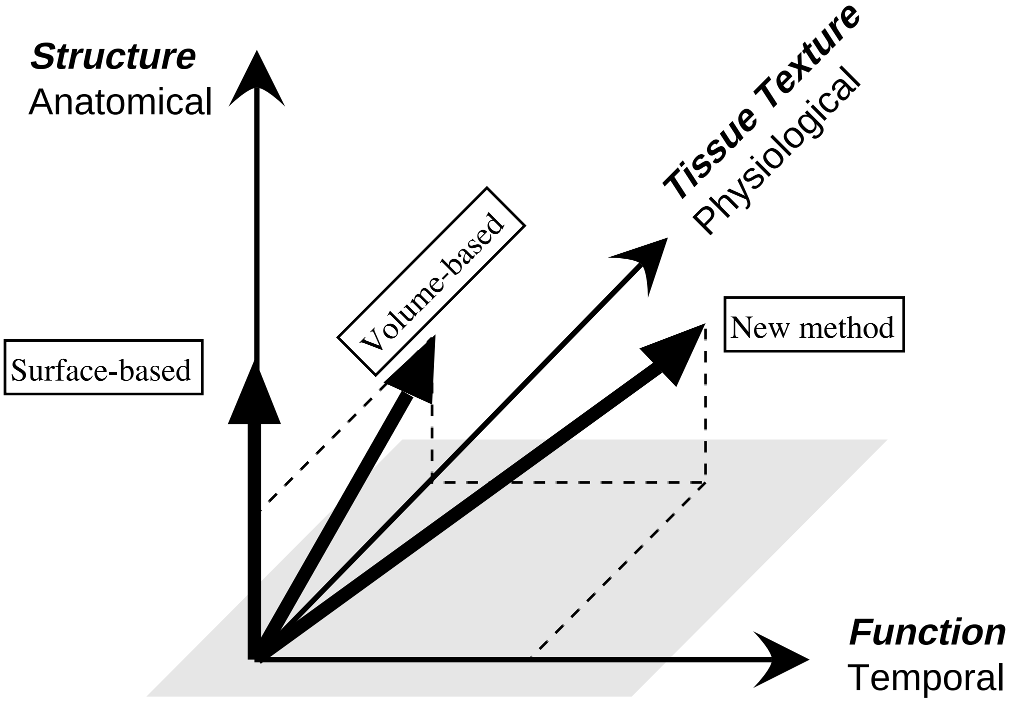


Fig. 1. Medical Data Volume Rendering Space. Superimposed are vectors representing the feature space character of volume and surface-based techniques and our new rendering method.

Fig. 1 illustrates current modelling approaches with respect to their temporal, physiological, and anatomical characteristics. Surface-based techniques and the geometric models they render capture the anatomical structures in medical volume data. Since volume techniques also use surface information, they can become functionally equivalent to surface-based ones with the difference being the absence of a geometric model. A good example of this is the Marching Cubes algorithm.[6] Therefore, it would be inaccurate to restrict their domain to tissue texture, although representation of physiological data is the dominant characteristic of volume techniques. Unlike surface-based techniques, volume techniques may represent both anatomical and physiological attributes.

Neither current surface nor volume rendering techniques are adequate for real-time interactive manipulation of volumetric data and the reproduction of it's intrinsic structural, textural, and temporal characteristics.

2. MATERIALS

Our model is rendered on a Silicon Graphics Iris 120 GTX graphics workstation. In addition to traditional flat screen images, we also provide stereoscopic images to aid visualization of the 3-D relationships of the slices. Stereoscopic images are displayed with a Crystal Eyes™ stereoscopic shutter in conjunction with a 120 Hz. monitor. The shutter's design allows multiple shutters to be used with one monitor, thereby encouraging consultation among clinicians and researchers. A SpaceBall™ serves as the 3-D direct manipulation mechanism.

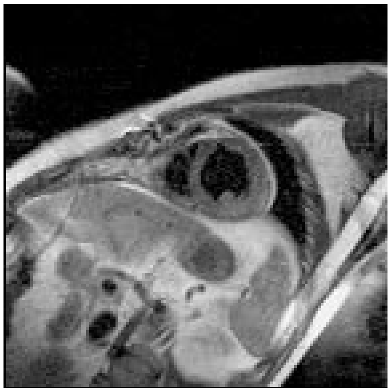


Fig. 2. Original cross-sectional MR image

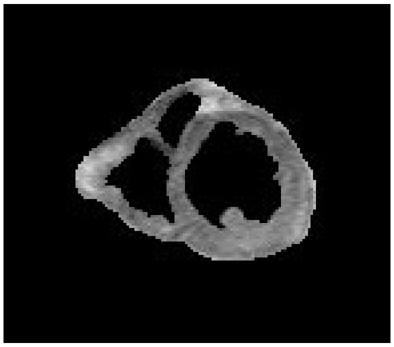


Fig. 3. Masked region of MR image

3. METHODS

Three inputs are needed to generate the model: the volume data, the volume mask, and the sampling geometry parameters

3.1 Volume mask

A volume mask is required for defining the chamber contours as well as the external wall boundaries of the cardiac MRI data. A physician "paints" mask images over the cardiac MRI slices to identify the heart muscle areas (see Fig. 2 and 3). The mask image is then processed with a contour extraction algorithm. The resultant multiple contours (chamber definitions and organ outline) are used in defining the graphical model of the heart as well as opaque image areas to be rendered.

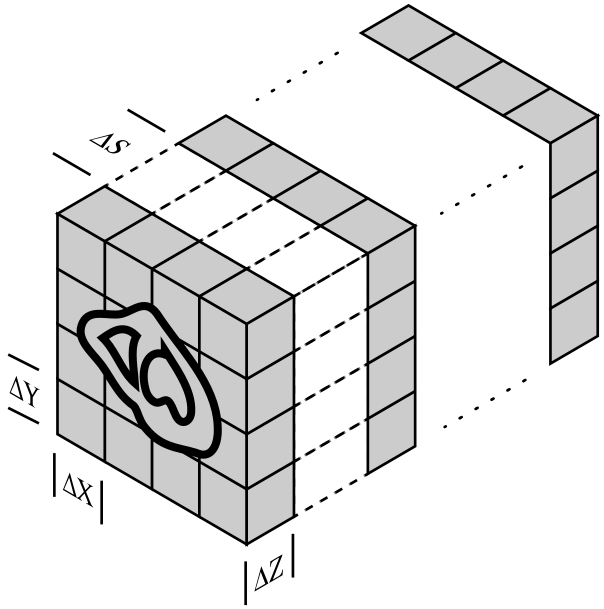


Fig. 4. Sampling geometry parameterized by ΔX, ΔY, ΔZ, and ΔS (slice separation)

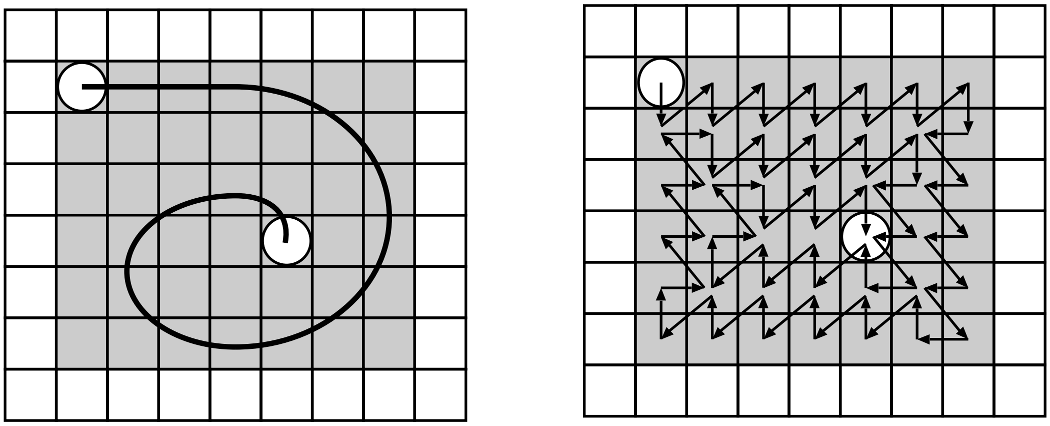
3.2 Geometry

Four geometry parameters are required. The sampling geometry defines the physical positioning of the sampling planes when the patient is scanned. Sampling geometries can be defined and parameterized by dx, dy , dz, and ds . The dx, dy, and ds variables specify the planar resolution and relative positioning of slice data, respectively (see Fig.4).

Since a volumetric data set is composed of infinitesimally small points, a slice from this data set is similarly thin. In actuality, the sampling geometry is not so ideal. Almost all scanners sample over a volume for every data point produced. Therefore the geometric model representing the volume must account for this reality. Our model solves this problem by rendering "thick" planar contours for each slice. This representation gives each component of the model an extruded quality reflecting the actual spatial conversion, and enhances the physical realism.

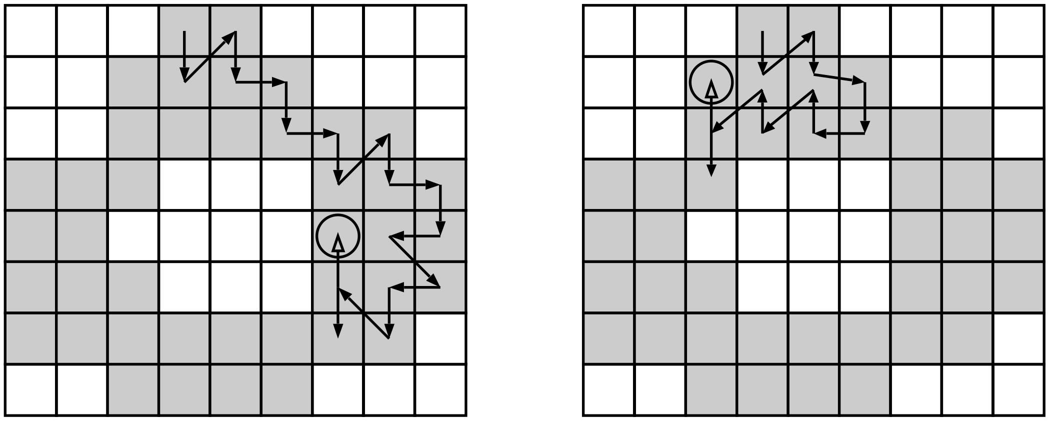
The scalar values in the volume data, or some processed version of this data, are mapped onto the geometric model (derived from the mask volume). The mapping process is the most critical aspect of the rendering since it determines the complexity of the graphical model. Texture mapping is not practical in this situation since the contours have non-rectangular, arbitrary areas, and holes.

The most efficient means of rendering the model is a triangular mesh. Most high-performance graphics workstations implement triangular meshes with a constraint on vertex ordering. In order to avoid redundant vertex calls these systems require a sequential ordering of the mesh triangle vertices. This constraint may, in general, speed up the rendering computations but it adversely affects our needs, as will be shown.



(a) (b)

Fig. 5. (a) & (b) represent a square image contour without holes and it's resultant mesh ordering. □ All data points from (a) are captured in (b) without redundancy from the start pixel to the end pixel. □ (a) shows the general mesh order for circumnavigation. (b) is the resultant mesh vertex ordering.



(a) (b)

Fig. 6. (a) & (b) illustrate the data loss problem when the image contour has holes. (a) is a □ circumnavigated mesh construction . Note the missed data point. (b) is an area or top down □ ordering of the same image contour with holes. Again, note the missed data point.

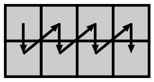
Legend

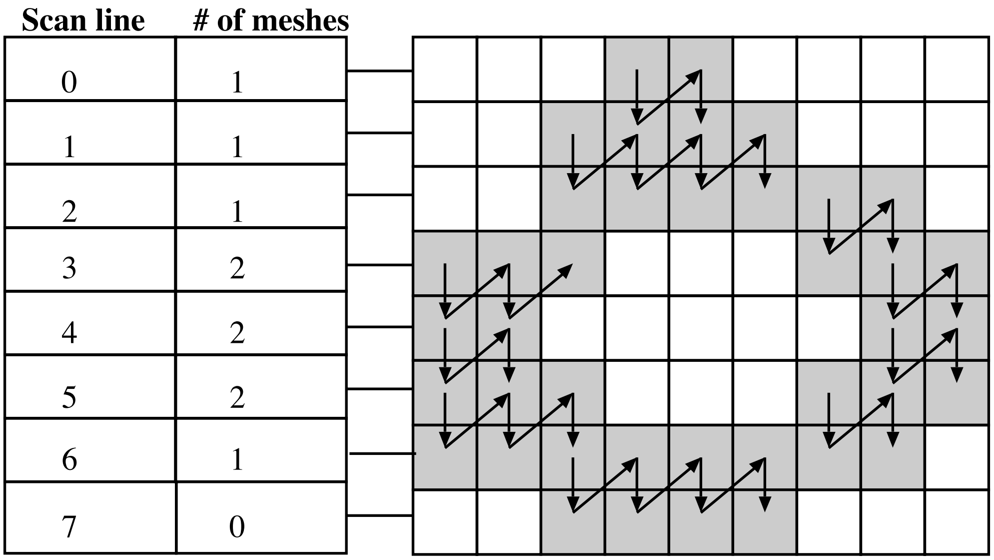
○ Starting and ending mesh pixels

◍ Missed image pixel

□ Image pixel outside object

▨ Image pixel within object


Mesh vertex order



Scan line	# of meshes
0	1
1	1
2	1
3	2
4	2
5	2
6	1
7	0

Fig. 7. Scan line method for meshing the image contour. All data points captured without redundancy.

A circular meshing of the data is trivial (Fig. 5a) and results in an area traversal as shown in figure 5b. Generalizing the contours to include holes introduces a potential data loss situation. For example, if such contours were meshed by circumnavigation, some pixel areas would be omitted (see Fig. 6a). If an area or top-down mesh approach is taken the same situation is possible (see Fig. 6b). The problem is caused by thin contour structures. It is possible to include these data points in the meshing traversal but at the cost of introducing collinear points in the geometric model. Though this problem may occur with any type of scanned volume it is unavoidable in cardiac MRI. Whether the meshing algorithm is top-down or circular this problem will cause volume data to be lost. The mesh traversal problem was solved by using a scan line approach (see Fig. 7) insuring the inclusion and rendering of all image elements identified for display and exclusion of collinear points.

4. RESULTS

When the model is rendered the heart chambers appear hollow (see Fig. 8). This rendering gives a realistic view of the chamber volumes. The jagged appearance of the planar components follows from a strict adherence to the physician-generated mask data. The rendering process must minimize modification of the volume of interest .

Each planar component of the model is bounded by the volume of interest region coincident on the sampling plane, producing a planar object whose border is a contour of the volume of interest. If a surface-based algorithm were used, the contour edge points would be the data used to reconstruct the 3-D surface. In those methods the 3-D surface is created from the contour edges, which are 2-D curve approximations, by interpolation. Therefore, the only canonical data points used in the surface reconstruction are the contour edge points. Our geometric model is only defined within the sampling space (canonical data), free of interpolated data. The slices are not connected geometrically in order to keep the rendering free of interpolated data. Volume techniques resample the volume data before rendering, introducing interpolation errors as well. One of the benefits of building the geometric model on top of the sampling geometry is that we avoid introducing sampling errors. Assuming that the scanner adhered to the Nyquist frequency of the volume signal, our model will not introduce false data.

Fig. 8. Rendered MRI volumetric data

Our model images the whole volume without occluding the interior volume data. It differs from volume techniques because data points are either opaque or transparent, as specified by the mask data. In addition, it is geometrically simpler than surface-based models yet has the same structure defining quality while capturing a greater amount of volume data.

4.1 Performance

The model works well for our cardiac studies because it is effective for high planar resolution, low slice density data. The model is even more effective when the volume of interest consists of cavities. The more surfaces (cavities create surfaces) in the volume of interest the better the performance of our volume renderer. The cavities provide three benefits: they help define and highlight structural attributes in the volume data, they create ports in the data for greater views of physical relationships and data, and they reduce the rendering load by reducing the data set size.

The model uses the reduced graphical complexity gained from the planar components to allow the user to interactively manipulate the model while it is being animated. The quality of the animation and the response to real-time manipulation is dependent on the complexity of the volume of interest geometry. We have achieved acceptable performance for six thick slices, with each slice containing approximately 16K data points. The addition of real-time stereoscopic animation degrades performance on a SGI GTX120 graphics workstation.

We have looked at several means of increasing our rendering speed and interactivity for more complex images without increased hardware costs. One approach embarked upon was to apply adaptive refinement techniques to the rendering.[9,10] Unfortunately, these methods cannot be used because of the inherent temporal constraints. Basically, such methods involve refining the image over time. But the animation may have a higher frame rate than the adaptive refinement sequence. In addition, the spectral characteristics of an inferior medical image would obviously cause problems in a clinical setting.

5. CONCLUSION

We have developed a volume rendering technique that aids the clinician in visualizing anatomical, physiological and temporal characteristics of MRI data. The technique avoids resolution loss and other problems outlined above by maintaining the slice image character of the original data set. In addition, through interactive animation, multiple spatial visual perspectives of the heart in 3-D can be generated. Relevant portions from the grayscale Cardiac MR volume are transformed geometrically to correspond to their size and locations in space as would be seen by an observer at a specified location. Contour data is used to restrict this transformation to certain desired structures in the images, while making other structures in the images completely transparent as desired by the observer. Thus, this technique displays the geometric continuity of the structures of interest and allows the observers to investigate complex, abnormal, and previously undefined geometric interrelationships. Information about the sampling geometry is incorporated in the rendering by displaying each slab with an appropriately scaled thickness, instead of an infinitesimally thin plane.

The volume rendering method uses a graphical model that reduces the levels of visual mappings between the original sampled volume and the displayed image. In addition, the system creates true 3-D, interactive, real-time, animated renderings of the data, unlike other volume rendering techniques which create movie loops of the data off-line.

6. ACKNOWLEDGEMENT

This work was partially supported by National Science Foundation grant #EID-8920774.

7. REFERENCES

1. Udupa, J. K., "Computer Aspects of Three-Dimensional Imaging in Medicine: A Tutorial," Proceedings of 3-D Imaging in Medicine, Hospital of The University of Pennsylvania, Ch. 1, 1989.

2 Kaufman, Arie, Roni Yagel, Daniel Cohen, "Intermixing Surface and Volume Rendering," Proceedings of the Nato Conference on 3-D imaging in Medicine, pp. 217-227 Springer-Verlag: Berlin, 1990.

3. Menhardt, W., I.C. Carlsen, "An Interactive Environment for Shaded Perspective Displays of MR Tomograms," Innovation and Technology in Biological Medicine, Vol. 8, No. 1, pp. 104-120, 1987.

4. Kajiya, James T., Brian P. Von Herzen, "Ray Tracing Volume Densities," Computer Graphics, Vol. 18, No. 3, pp. 165-174, July 1984.

5. Levoy, Marc, "Display of Surfaces from Volume Data," IEEE Computer Graphics and Applications, pp. 29-37, May 1988.

6. Drebin, Robert A., Loren Carpenter, and Pat Hanrahan "Volume Rendering" Computer Graphics, Vol. 22 No. 4, pp.65-74, August 1988.

7. Lorensen, William E., Harvey E. Cline, "Marching Cubes: A High Resolution 3D Surface Construction Algorithm," Computer Graphics, Vol. 21,.No. 4, pp. 163-169, July 1987.

8. Miller, James V., David E. Breen, William E. Lorensen, Robert M. O'Bara, Micheal J. Wozny, "Geometrically Deformed Models: A Method for Extracting Closed Geometric Models from Volume Data," Computer Graphics, Vol. 25.No. 4, pp. 217-226, July 1991.

9. Bergman, Larry, Henry Fuchs, Eric Grant, "Image Rendering by Adaptive Refinement," Siggraph, Vol. 20, No. 4, pp. 29-37, August 1986.

10. Levoy, Marc, "Volume rendering by adaptive refinement," The Visual Computer Vol. 6, pp. 2-7, 1990