

Full Length Research Paper

An overview of medical image processing methods

Hakan Hadi Maraş^{1*}, Yüksel Maraş², Erdem Emin Maraş³, Bahadır Aktuğ¹, Süleyman Sırrı Maraş⁴ and Ferruh Yildiz⁵

¹General Command of Mapping, Department of Photogrammetry, Ankara, Turkey.

²Department of Rheumatology, Atatürk Training and Research Hospital, Ankara, Turkey.

³Geomatics Engineering, Engineering Faculty, Ondokuz Mayıs University, Samsun, Turkey.

⁴Vocational School of Technical Sciences, Selcuk University, Konya, Turkey.

⁵Department of Surveying, Selcuk University, Konya, Turkey.

Accepted 10 May, 2010

Since human life is worthier than all things, efforts on virtual animation and visualization of human body's viscera, without surgical interference to diagnose a disease is very important. Recently, modern medical instruments are able to produce views which can be used for better diagnoses and accurate treatment. Various standards were formed regarding these instruments and end products that are being used more frequently everyday. Personal computers (PCs) have reached a significant level in image processing, carried analysis and visualization processes which could be done with expensive hardware on doctors' desktops. The next step is to try to find out proper solutions by software developers and engineers that help doctors to make decision by combining opportunities in these two scientific areas. The objective of the present study is to construct 3D models and present it to users on screen in personal computers by using data acquired from tomography and magnetic resonance instruments. In order to realize this objective, developing software is aimed. In the second and third sections, the data structures and processing of 3D volumetric data in digital format, 3D visualization techniques and theoretical subjects about methods and algorithms used are explained. In the fourth section, explanations on developing a software package for the realization of the objective of the study, its usage and information about software development tools used are given. In the last section, the determinations made at the end of trials in this study, difficulties met and recommendations obtained in the light of the trial results are presented.

Key words: Virtual animation, surgical interference, 3D models, image processing.

INTRODUCTION

Computed tomography (CT) was introduced in the early 1970s and with recent technology it is now possible to carry out volumetric CT analyses with isotropic voxel spacing of better than 100 µm. In a typical CT application, X-ray projections are obtained at numerous equally spaced angular positions covering the object and these views are then used to reconstruct a 2D CT image. Today, CT applications have already become an indispensable part of clinical diagnostic procedure. The growing application of rendering and processing 3D

images through computer simulations has already increased the interests of many researchers. 3D image rendering usually refers to the analysis of the volumetric data clusters and projecting such images in 2D. The fundamental rendering methods can be grouped as surface rendering, slicing and volume visualization. Such methods are used in many disciplines including medicine, geodesy, photogrammetry, earth sciences, chemistry, astrophysics, biotechnology, metallurgy, etc. Volume visualization is generally applied to 3D solid object data.

In this respect, the most common sources of volume visualization are the scientific 3D data. Machiraju (1996) pointed out that scientific data clusters are generally obtained through observations, numerical simulations or

*Corresponding author. E-mail: hakan.maras@hgk.mil.tr

similar procedure. The data acquired through MRI (Magnetic Resonance Imaging), CT (Computer Tomography), ultrasound scanners and computer-aided high-resolution microscopes can be given as examples of such kind of data. The examples of the second kind of data type can be named as the computer-aided fluid dynamics data which aim at the perception and prediction of the motion of the natural events.

PROCESSING OF 3D VOLUMETRIC DATA

The first step of the visualization and the processing of 3D volumetric data requires *a priori* knowledge of how the data is stored in computer memory. Examples of 2D and 3D rectilinear, curvilinear and unstructured data samples are shown in Table 1. Rectilinear grids are common in medical applications as the output from scanners, curvilinear and unstructured grids are common in numerical simulations.

The scanners used to acquire 3D medical data are generally sensitive to the intensity of the tissues, magnetism and acoustic resistivity. Rendering a computer images for this physical quantities requires a further step of transforming them into the intensity and transparency of the light. This process forms one of the most important parts of the segmentation.

Rendering 3D volumetric data on the computers requires image processing techniques such as image segmentation, compression, sampling, classification, shading and compositing. The segmentation technique is concerned with separating the objects from others and from the background and is the most fundamental analysis function (Wismüller et al., 2000).

The main objective of the data segmentation is the grouping and classification into the homogenous sub-groups of the data with respect to one or more properties and attributes (Liang, 1993; Bezdek et al., 1993; Gonzales and Woods, 1993; Clarke et al., 1995; Castleman, 1996; Sonka et al., 1999; Suetens et al., 1993).

Image segmentation

Since the data segmentation requires the classification of the voxels and the pixels, it is generally handled as a pattern recognition problem. The data segmentation methods can be grouped as (Awcock ve Thomas, 1996), (Rajapakse ve ark, 1997);

- (1) Region-based segmentation: Determination of the region which provides the homogeneity.
- (2) Edge-based segmentation: Determination of the edges which present different characters (Gonzales and Woods, 1993; Sonka et al., 1999).

Usage of a threshold has a wide application in regional

segmentation. In this method, the data is grouped with respect to the pre-defined threshold values by comparing them to threshold. Clustering algorithms classifies the pixels which resembles most to the average of a specific group (Jain and Flynn, 1996; Gose et al., 1996). In this method, each pixel is handled individually and put into the most appropriate group which represents similar properties.

Edge-based segmentation is the process of detecting the unexpected changes in the values of the edges or layers. Such localized regions where the changes are relatively slow encompass the edges or the frames or the objects. Mathematical operators such as the gradient and hessian are appropriate tools for this kind of work. Assuming that the *f* is a function of the arguments *x* and *y*, the first derivative of *f* at point *P* can be expressed as:

$$\nabla f(x_p, y_p) = \left[\frac{\partial f(x_p, y_p)}{\partial x}, \frac{\partial f(x_p, y_p)}{\partial y} \right] \tag{1}$$

The changes in the gradient corresponding to the directions of rapid changes of the intensity of the light for different samples are shown in Figure 1.

The magnitude of the gradient values corresponding to the first and the second term of the Equation (1) are zero for the Figures 1a and b, respectively. The magnitudes of the gradient can be found as;

$$|\nabla f(x_p, y_p)| \cong \sqrt{\left(\frac{\partial f(x_p, y_p)}{\partial x}\right)^2 + \left(\frac{\partial f(x_p, y_p)}{\partial y}\right)^2} \tag{2}$$

The direction of the gradient θ can similarly be found as,

$$\theta = \arctan\left(\frac{\partial f(x_p, y_p)}{\partial x} / \frac{\partial f(x_p, y_p)}{\partial y}\right) \tag{3}$$

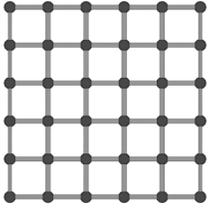
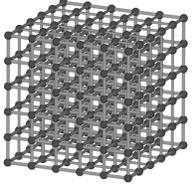
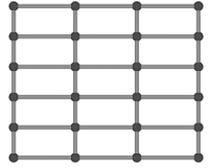
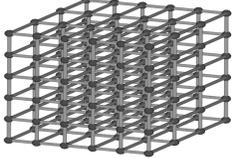
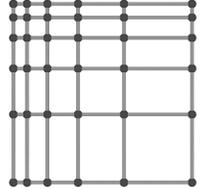
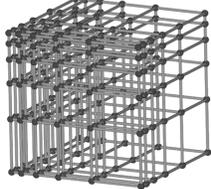
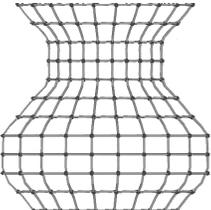
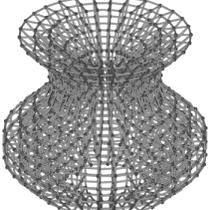
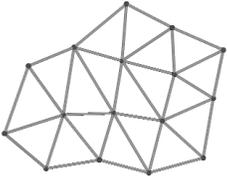
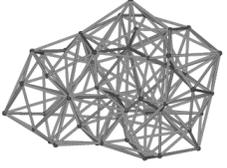
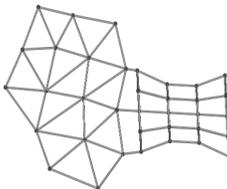
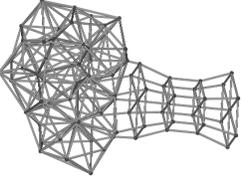
The equation given above is only valid for continuous functions. On the other hand, the digital images consist of pixels and intrinsically discreet. For the case of discrete functions, the approximate equations below can be utilized in order to speed up the computation process:

$$|\nabla f(x_p, y_p)| \cong \left| \frac{\partial f(x_p, y_p)}{\partial x} \right| + \left| \frac{\partial f(x_p, y_p)}{\partial y} \right| \tag{4}$$

The derivatives in the Equation (4) can be expressed as:

$$\frac{\partial f(x_p, y_p)}{\partial x} \cong f(x_p + n, y_p) - f(x_p - n, y_p) \tag{5}$$

Table 1. 2D and 3D data structures.

Rectilinear		Regular (isotropic)			
		Data points are evenly spaced and identical in all coordinate directions			
		Basic elements			
		Square	Cube		
Rectilinear		Regular (anisotropic)			
		Data points are evenly spaced but not identical in all coordinate directions			
		Basic elements			
		Rectangle	Rectangular prism		
Rectilinear		Irregular			
		Data points are unevenly spaced nor identical in all coordinate directions			
		Basic elements			
	 	Square Rectangle	Cube Rectangular prism	 	
Curvilinear		Regular			
		Data points are evenly spaced but the connection lines are regular in a curvilinear structure			
		Basic elements			
	  	Square Rectangle Quadrilateral	Cube Pentahedron Hexahedron	  	
Unstructured		Irregular			
		Data points are irregularly spaced and the connection lines form triangles			
		Basic elements			
		Triangle	Tetrahedral		
Unstructured		Hybrid			
		Combined regular and irregular forms of data points			
		Basic elements			
	   	Square Rectangle, Quadrangle Triangle	Cube Rectangular prism Polyhedrals	  	

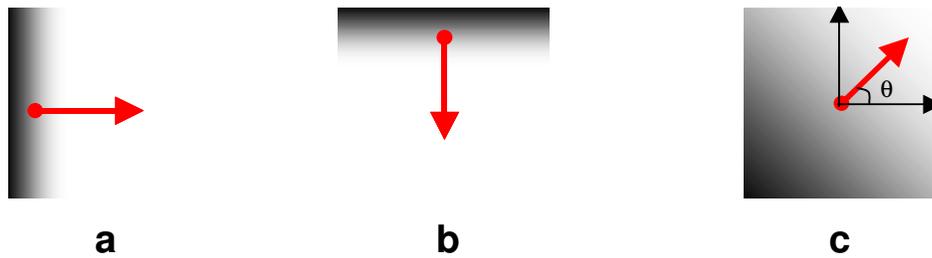


Figure 1. The gradient direction where the intensity of light changes rapidly.

$$\frac{\partial f(x_p, y_p)}{\partial y} \cong f(x_p, y_p + n) - f(x_p, y_p - n) \quad (6)$$

Here n is the smallest integer and is usually chosen as 1. A convolution mask like $-1 \mid 0 \mid 1$ could be used in each axis to apply Equations 5 and 6.

Various masks which are called operator can be formed using different approaches. In these approaches, the neighborhood relations are extended from 3×1 up to 3×3 or more and different operators are derived. The simple convolution process can be implemented using the equation below:

$$T \cdot I(x, y) = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} T(i, j) \cdot I(x + i, y + j) \quad (7)$$

In this equation, $I(x, y)$ represents the pixel while $T(i, j)$ corresponds to the elements of the $n \times m$ convolution kernel. The most common operators related to the first order derivatives are called Roberts, Prewitt, Sobel and isotropic (Frei-Chen). The Roberts cross-operator has a relatively smaller area of effect and subject to artifacts in the image. The gradient directions and the points of the Roberts operator are as follows;

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad \begin{matrix} \nearrow \\ \nwarrow \end{matrix} \quad (8)$$

On the other hand, the Prewitt operator given below is more sensitive in the horizontal and vertical than it is to the diagonals.

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{matrix} \downarrow \\ \rightarrow \end{matrix} \quad (9)$$

The Sobel operator is more sensitive to the diagonals

than it is to the horizontal and the vertical. The gradient direction and its points are as follows:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad \begin{matrix} \rightarrow \\ \downarrow \end{matrix} \quad (10)$$

The direction and the points of the isotropic operator Frei-Chen are as follows;

$$\begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix} \quad \begin{matrix} \rightarrow \\ \downarrow \end{matrix} \quad (11)$$

The most common operator of those given above to find the edges is the Sobel operator.

3D segmentation

3D segmentation is the process of determining the boundaries of the voxels which represents different structures, properties. 3D segmentation is applied by extending the operators above to the 3D. One common approach to facilitate the computation is to employ grey-level gradients which utilizes the differences between neighboring voxels (Höhne and Bernstein, 1986):

$$\nabla f(x, y, z) = \begin{bmatrix} \nabla f_x \\ \nabla f_y \\ \nabla f_z \end{bmatrix} = \frac{1}{2} \begin{bmatrix} f(x+1, y, z) - f(x-1, y, z) \\ f(x, y+1, z) - f(x, y-1, z) \\ f(x, y, z+1) - f(x, y, z-1) \end{bmatrix} \quad (12)$$

3D RENDERING

The relatively easier acquisition of the 3D volumetric data makes it very popular among numerous disciplines. Today, researchers investigate the interiors of many objects and living organisms which they do not have

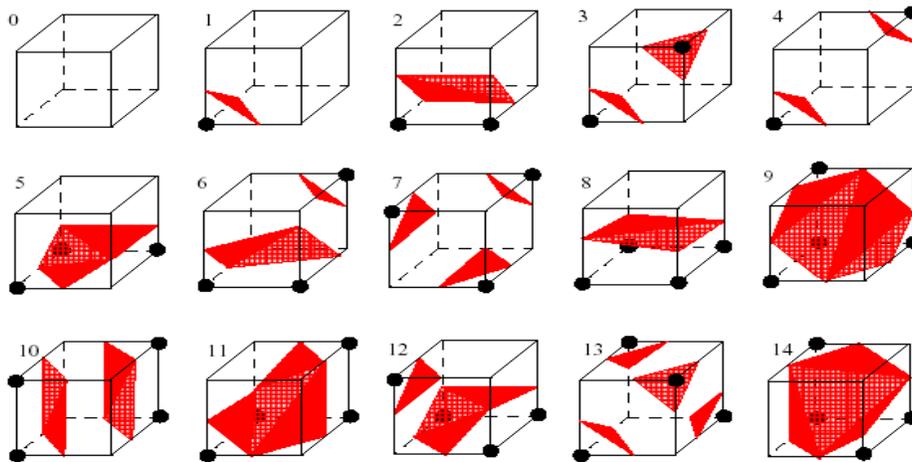


Figure 2. Possibilities of passing for the isosurface.

direct access to. In this respect, 3D rendering involves not only the rendering of 3D data but also image processing and presentation in various formats. There are mainly three methods for rendering the 3D images: Surface rendering, slicing and volume rendering. These methods also use different application-specific algorithms. In this respect, it is possible to divide the existing methods into sub-categories.

Surface rendering

While there exist many methods about surface rendering, the early methods were mainly concerned with displaying the surface through the simple geometric figures. The most common ones of such methods can be given as contour tracing (Keppel, 1975), opaque cubes (Herman and Liu, 1979), marching cubes (Lorensen and Cline, 1987), marching tetrahedral (Shirley and Tuchman, 1990), dividing cubes (Cline et al., 1988) and many others. In these methods, the cube or the similar geometric element is moved through the volume. In this operation, it is checked whether the basic element intersects the surface or not. After compiling the pieces intersecting the surface, the whole surface is obtained.

Another method used for surface extraction is the approach of the marching the cubes. This approach is proposed by Lorensen and Cline (1987). In this approach, the data is regular and structured and each element is handled independently (Brodie and Wood, 2001).

Through the values of the voxel at the corners, the position of the threshold representing the surface at voxel edges is determined by employing the 3D directional and linear interpolation. There are 256 different possibilities for the isosurface element to pass through the volume

element. Lorensen and Cline (1987) summarized these possibilities as shown in Figure 2. On the other hand, the symmetry, similarity, canonical and complementary properties constrain the possibilities and 15 distinct rendering could be conceived. When the isosurface extraction is completed for each voxel, the rendering could be formed taking the coloring and using shadowing into account.

Slicing

The slicing methods emerged in the early 1960s using the first scientific rendering studies and the least popular method of the available methods since the high-density data requires too many slices (Bruckner, 2004).

This method, represent the 3D figures as sliced pieces and reduces the problem of 3D rendering to a 2D application. However, since the perception of 2D data is much easier than 3D, it still has many uses in various applications (Brodie and Wood, 2001).

Volume rendering

While classification for the volume rendering is made based on different criteria, the most popular method is the one based on the 3D volumetric image-processing order. In this respect, the image rendering methods can be group as;

- (1) Image-order volume rendering
- (2) Object-order volume rendering
- (3) Hybrid-order volume rendering

In the object-order volume rendering, the ray-casting is applied in front-to-back order, the procedure of viewing

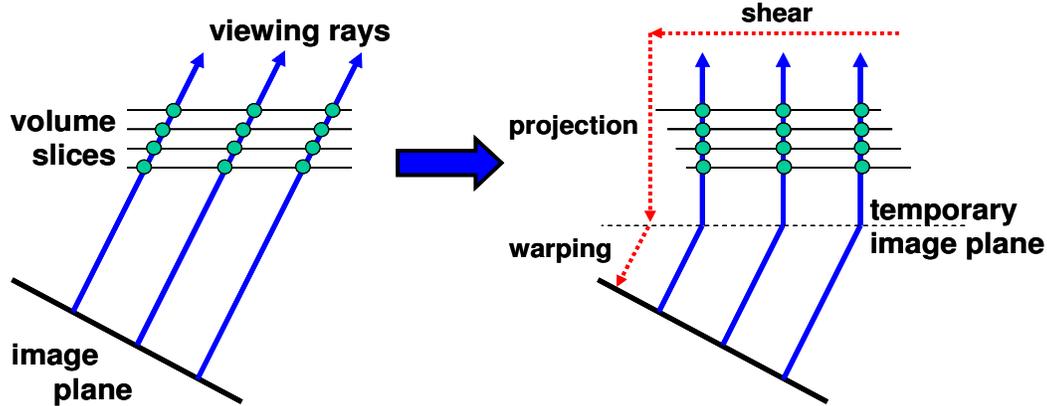


Figure 3. Hybrid-order volume rendering.

rays is terminated when a full opacity voxel is encountered. One problem in the raycasting procedure is the omission of non-contributing regions since various medical data usually contain voxels of this type, this could have a direct impact on the performance.

As opposed to the image-order techniques, object-order methods determine for each data sample shows how it affects the pixels on the image plane (Bruckner, 2004). An object-order algorithm loops over the data matching each sample with the relevant position on the image. Splatting is another technique with the advantage that zero-opacity voxels are naturally omitted since they do not contribute to the image. Splatting technique relies on those traverses and projects footprints (known as splats) onto the image plane (Westover, 1990). This advantage could significantly reduce the data to be processed and speed up the whole rendering process. On the other hand, several disadvantages could also be given as introducing inaccuracies into the compositing process through the kernels since the 3D reconstruction kernel is composited as a whole.

To get rid of the disadvantages of the image-order and object-order techniques and gathering their advantages, hybrid-order method was developed (Ray et al., 1999). The method which is also known as Shear-Warp, is considered as the fastest of software-based volume rendering methods (Lacroute and Levoy, 1994). In this method, the rays going through the volume are sheared up to the perpendicular position as shown in Figure 3.

Comparison of the rendering methods

Miller (1994) has shown the differences between the classical surface rendering and the volume rendering as shown in Table 2. The most significant difference between the two methods is the loss of geometric information due to the discrete form of the volume rendering and its huge

memory and operational requirements. On the other hand, the superior advantage of the volume rendering is the possibility of displaying the interior parts of the objects. An example for the surface rendering and volume rendering is shown in Figure 4.

APPLICATION

A software application was developed to demonstrate the merits of methods of tomographic and magnetic resonance data on a personal computer described in the previous chapters. The application was developed in Visual C++.NET development platform employing OpenGL and OpenGL Utility (GLUT) libraries. Full compliance with the ANSI standards was given priority during the development to maximize the portability. In the respect, the application could easily be ported to Linux or similar platforms with minor modifications.

In the development phase, the efforts were focused on the correct displaying of 3D data and the performance issues. Thus, the user interface was deliberately kept as simple as possible. Moreover, it was strictly avoided to use user interface elements which are specific to the operation system.

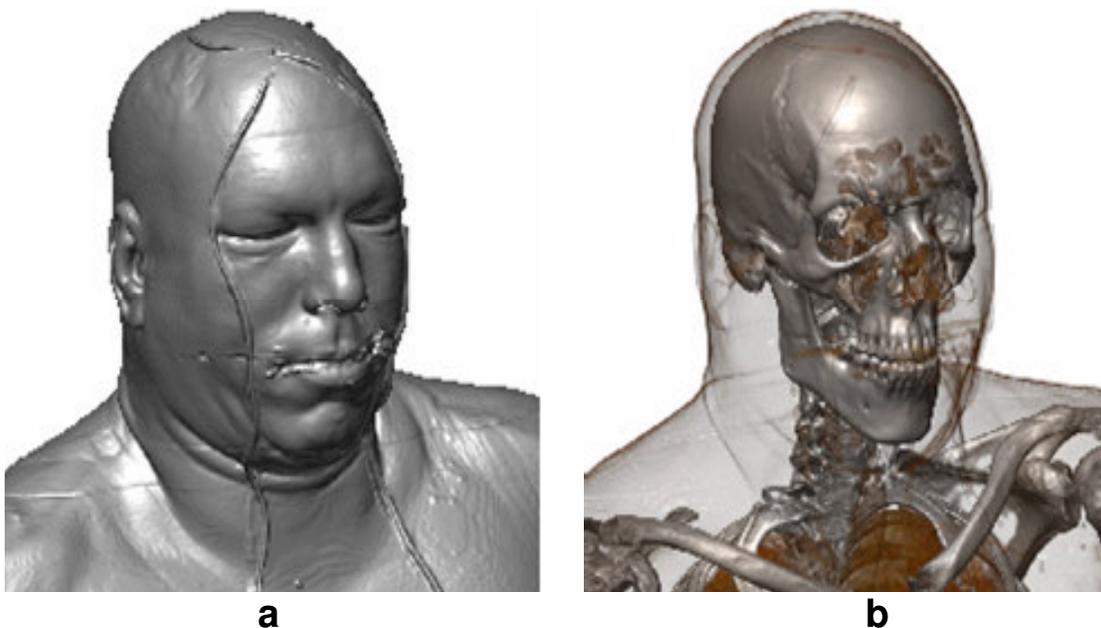
In this study, two different applications were presented; the first one is about the surface rendering employing the marching cubes method while the other is about the 3D rendering which allows the investigation of the interiors of the objects by using the volume rendering method.

Surface rendering application

In this application, the processes or un-processed 3D medical data are imported and the surface of the 3D object is visualized through the marching cubes algorithm. An example of such an application is given in Figure 5

Table 2. Comparison between surface and volume rendering (Miller, 1994).

Criteria	Surface Rendering	Volume Rendering
Rendering performance	Sensitive to scene and object complexity	Same, regardless to scene complexity and/or object complexity
Memory and processing requirement	Variable, depending in scene and object complexity	Large but constant
Object-space aliasing	None	Frequent
Transformation	Continuous; performed on geometric definition of objects	Discrete; performed on subvolumes
Scan conversion and rendering	Pixelization is embedded in viewing	Voxelization is decoupled from viewing
Boolean and block operations	Difficult; must be performed analytically	Trivial; by using voxblt, voxel-by- voxel operations, aggregation, octrees
Rendering of interior	No; surfaces only	Yes; rendering of inner structures as well as surfaces
Adequacy for sampled data and intermixing with geometric data	Partially and indirectly (fitting followed by surface rendering)	Supports representation and direct rendering
Measurements (for example, distance, area, volume, normal)	Analytical, but may be complex	Discrete approximation, but simple
Viewpoint dependency	Requires recalculation for every changed viewpoint	Precomputes and stores viewpoint changed viewpoint independent information

**Figure 4.** Comparison of surface and volume rendering. (a) Traditional surface representation of a volumetric data set. (b) Direct volume rendering of the same dataset (Bruckner, 2004).

with MR data taken from a skull.

Volume rendering application

This application imports the processes or un-processed

3D medical data compiled from different sources and the volume of the 3D object is visualized so as to enable the investigation of the interiors. A typical example of such an application is given in Figure 6 for the case of skull. Another rendering of human brain was given in Figure 7 which was separated through a pre-process.

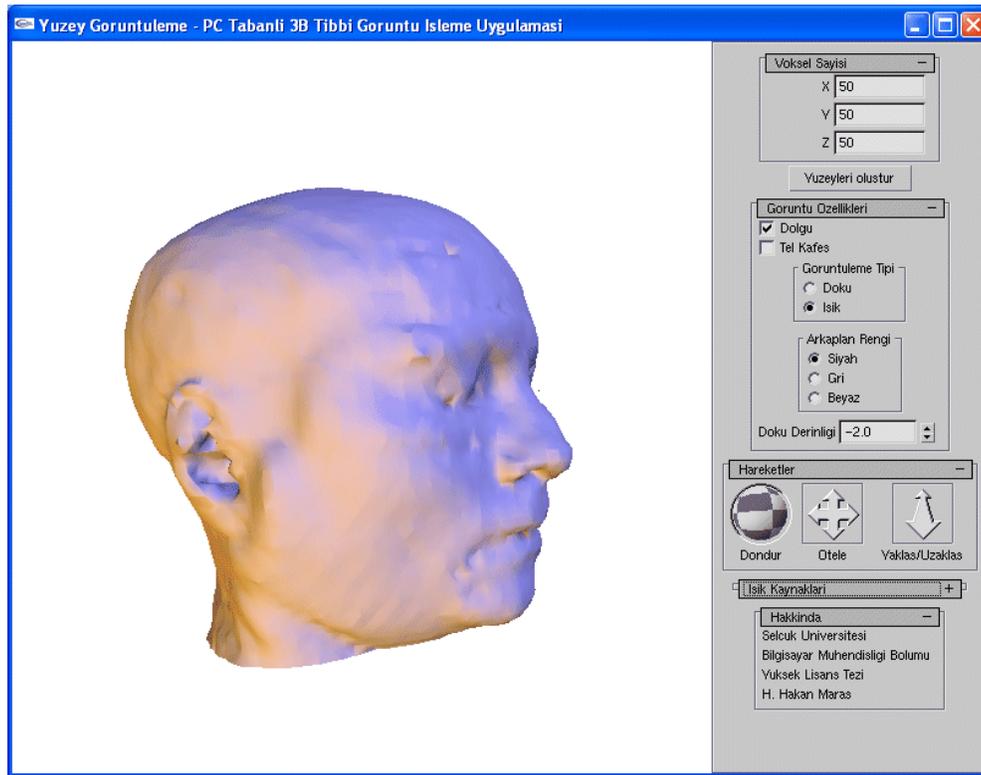


Figure 5. 3D Surface rendering.

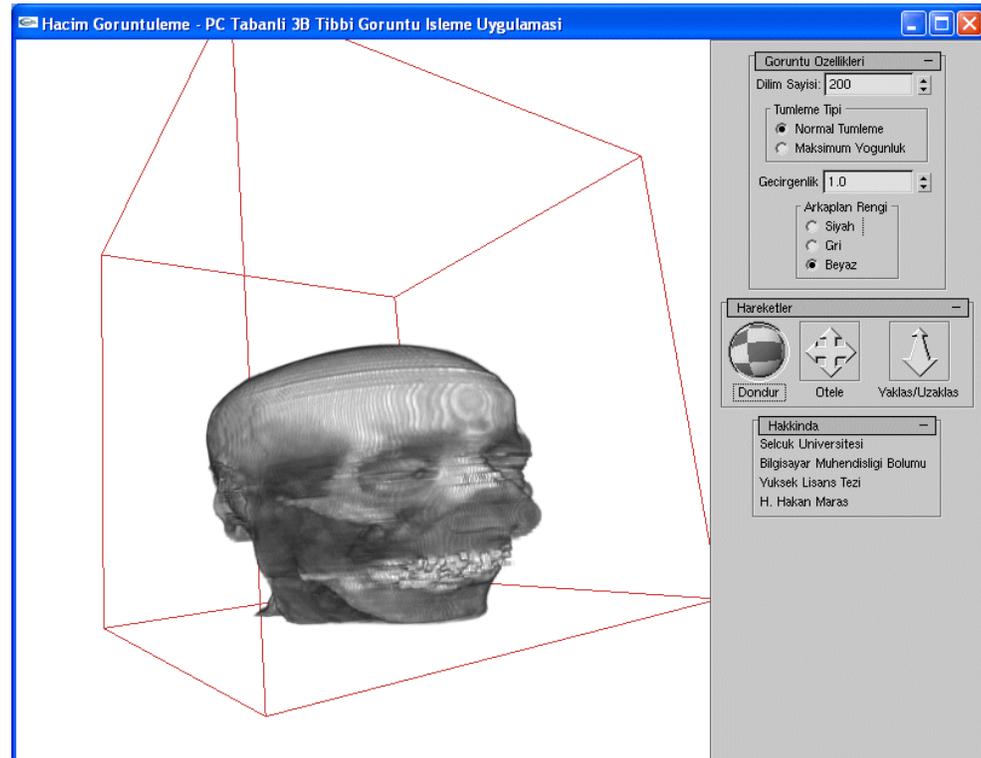


Figure 6. 3D volume rendering.

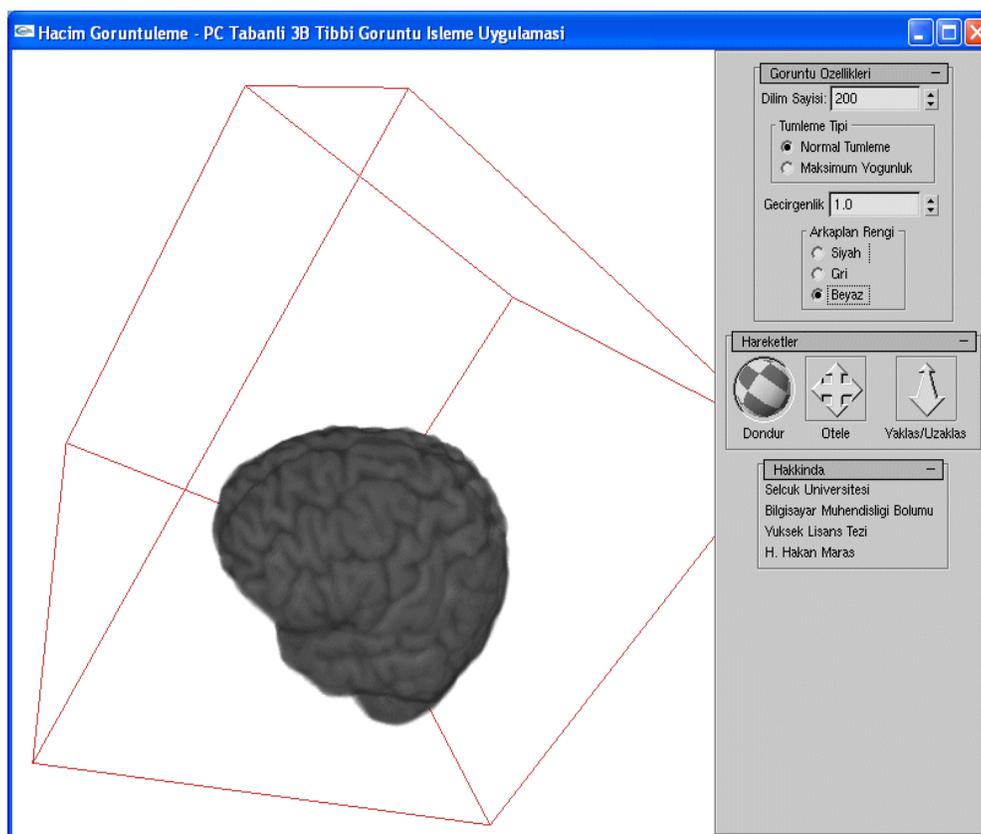


Figure 7. 3D visualization of the human brain.

PERSPECTIVES

The amount of medical data collected through Magnetic Resonance Imaging, Computerized tomography, ultrasound scanners and computer-aided high resolution microscopes increases day by day. The recent advances in computer technology have enabled analysis and visualization of such data on relatively cheaper computers. Since most diagnosis systems are able to output 3D data, 3D rendering methods are indispensable in order to take full advantage of such data.

The method of visualization is as important as the hardware resolution of the instrument. Otherwise, most of the obtained information could be lost. In this study, the methods of 2D surface and 3D volume rendering are investigated and it was shown that the appropriate choice of rendering method could play a key role in what can be obtained from such medical instruments. On the other hand, the utmost methods of visualization were not available in the commercial software. In this study, it was demonstrated that custom applications could be developed with little effort by non-professional software programmers. The results show that the number of slices should be more than 200 in order to carry out a correct analysis and to obtain significant results.

REFERENCES

- Awcock GJ, Thomas R (1996). Applied Image Processing. New York: McGraw-Hill, Inc.
- Bezdek JC, Hall LO, Clarke LP (1993). Review of MR image segmentation techniques using pattern recognition, *Med. Phys.* 20(4): 1033-1048
- Brodie K, Wood J (2001). Recent Advances in Volume Visualization. *Comput. Graphics Forum*, 20(2): 125-148.
- Bruckner S (2004). Efficient Volume Visualization of Large Medical Datasets, M.S. Thesis, Institut of Computer Graphics and Algorithms, Vienna University of Technology, Vienna.
- Castleman KR (1996). Digital Image Processing. Upper Saddle River: Prentice Hall.
- Clarke LP, Velthuizen RP, Camacho MA, Heine JJ, Vaidyanathan M, Hall IO, Thatcher RW, Silbiger ML (1995). MRI segmentation: methods and applications, *Mag. Res. Imag.* 13(3): 334-368.
- Cline HE, Lorensen WE, Ludke S, Crawford CR, Teeter BC (1988). Two algorithms for the reconstruction of surfaces from tomograms. *Med. Phys.* 15(3): 320-327.
- Gonzales RC, Woods RE (1993). Digital Image Processing. Reading, MA: Addison-Wesley Publishing Company.
- Gose E, Johnsonbaugh R, Jost S (1996). Pattern Recognition and Image Analysis. Upper Saddle River: Prentice Hall.
- Herman GT, Liu HK (1979). Three-dimensional display of human organs from computed tomograms. *Comp. Graphics and Image Proc.* 9(1): 1-21.
- Höhne KH, Bernstein R (1986). Shading 3D-images from CT using gray-level gradients. *IEEE Trans. Med. Imag.* 5(1): 45-47.
- Jain AK, Flynn PJ (1996). Image segmentation using clustering, in *Advances in Image Understanding*, Bowyer K and Ahuja N Eds. Los

- Alamitas, CA: IEEE Computer Society Press.
- Keppel E (1975). Approximating complex surfaces by triangulation of contour lines. *IBM J. Res. Dev.* 19(1): 2-11.
- Lacroute P, Levoy M (1994). Fast volume rendering using a shear-warp factorization of the viewing transformation. *Computer Graphics, Annu. Conf. Series*, 28: 451-458.
- Liang Z (1993). Tissue classification and segmentation of MR images, *IEEE, Med. Biol.* 12(1): 81-85.
- Lorensen WE, Cline HE (1987). Marching cubes: A high resolution 3D surface construction algorithm. *Comput. Graphics*, 21(4): 163-168.
- Machiraju R (1996). Analysis, Control and Evaluation of Image Generation in Volume Rendering, PhD thesis, The Ohio State Univ.
- Miller EJ (1994). Volumetric Visualization of Non-Point Source Contaminant Flow Utilizing Four-Dimensional Geostatistical Kriging, M.S. Thesis, Department of Natural Sources, The Ohio State University, Ohio.
- Rajapakse JC, Giedd JN, Rapoport JL (1997). Statistical approach to segmentation of single-channel cerebral MR images, *IEEE Trans. Med. Imag.* 16(2): 176-186.
- Ray H, Pfister H, Silver D, Cook TA (1999). Ray casting architectures for volume visualization. *IEEE Trans. Vis. Comput. Graphics*, 5(3): 210-223.
- Shirley P, Tuchman A (1990). A polygonal approximation to direct scalar volume rendering. *Comput. Graphics*, 24(5): 63-70.
- Sonka M, Hlavac V, Boyle R (1999). *Image Processing, Analysis, and Machine Vision*. CA: PWS Publishing, Pacific Grove.
- Suetens P, Bellon E, Vandermeulen D, Smet M, Marchal G, Nuyts J, Mortelman L (1993). Image Segmentation: methods and applications in diagnostic radiology and nuclear medicine, *Eur. J. Radiol.* 17: 14-21.
- Westover L (1990). Footprint evaluation for volume rendering. *Comput. Graphics*, 24(4): 367-376.
- Wismüller A, Vietze F, Dersch DR (2000). Segmentation with Neural Networks, *Handbook of Medical Imaging*, Bankman, I.N (ed), Academic Press, San Diego. pp. 107-126.