MODELING AND VISUALIZATION OF THREE-DIMENSIONAL OBJECTS
BASED ON SLICES

CLAUDIO KIRNER
Department of Computer Science
Federal University of Sao Carlos
Sao Carlos, SP, Brazil
(ckirner@blanca.uccs.edu)

ABSTRACT

This paper presents an integrated system for modeling and visualization of three-dimensional objects, which uses volume graphics. The object modeling is based on slices and the visualization uses Iris Explorer software. Both the modeling technique and the visualization system are described and their restrictions are discussed. An example of manual modeling of a human skull is presented and results of its visualization are also shown. The system potentialities are presented with emphasis on some of its possible special effects and applications.

KEYWORDS: computer graphics, 3D objects, modeling, visualization

1. INTRODUCTION

In recent years, the three-dimensional (3D) modeling and visualization areas are receiving an increasing interest, in consequence of the demand that is being created by complex applications and the availability of powerful graphic computational systems.

Several works on modeling and visualization, which involve volume graphics, are being developed [6, 15, 18]. In addition, other works on 3D surface graphics are also raising interest [8, 9, 17].

An integrated approach for 3D modeling and visualization was developed. The idea, technique, and results are presented here.

1 Visiting Research Professor at Department of Computer Science, University of Colorado at Colorado Springs, CO, USA, with grant provided by CNPq, Brazil, Proc. 201443/93-7.
According to the definition of pixels, a two-dimensional (2D) image can be considered as a set of small colorful squares which are centered at the points of a 2D grid. Handling pixels directly, images can be inserted into the computer, either by defining the pixels by hand (act of drawing) or by using some 2D scanner process. Then, it is possible to get very simple or refined images, depending on factors such as: grid granularity; characteristics of each pixel; and quality of by hand work or the scanner process. For example, the creation of icons is based on this technique.

Using the same idea for 3D images, it is possible to represent 3D objects in the computer, from a 3D grid that contains small cubes centered at each grid point. Each cube, named "voxel" (volume element), may indicate if it takes part of an object and express other characteristics, such as: color, tonality, transparency, etc. In this manner, the acquisition of the object voxels can be made either through by hand definition (act of sculpting), or by 3D scanner. A way to make modeling of 3D objects easier is to acquire the voxels that belong to slices of an object. After this, the visualization system must put the slices together and display the modeled object in 3D space.

This slicing technique uses volume graphics and allows the representation and visualization of internal details of objects. The technique can also be used to display only external surfaces, although it presents advantages and disadvantages related to the use of surface graphics techniques [6].

On the following items, the paper discusses some relevant problems related to 3D modeling and visualization. It is described a method of volumetric modeling, the visualization environment using Iris Explorer software, and a constructed example. Also, the primary potentialities of using the system in 3D applications are pointed out.

2. THE MODELING METHOD

The objective of the modeling method being presented is to generate a set of 3D object slices, representing a 3D object. The cut planes direction used as reference to produce the slices can be arbitrary. It is possible to create or recreate an object in space by putting the slices together.

Each slice is represented by a 2D matrix, where each element corresponds to a voxel (small cube of the slice). If the voxel is part of the object, then it must have a value different of zero; otherwise, its value will be zero to indicate empty space.

Preliminary, the system is using homogeneous colored objects, so that the values of voxels are useful only to indicate whether they are part of the object or not. Different values of voxels are used to indicate different parts of the object. Some effects in the visualization phase can be produced by different values of voxels and variation of a threshold value. Selection of a specific part of the object is achieved in this manner.
This type of modeling allows the generation of both surface objects (without inner details) and volumetric objects (with inner details).

The modeling must satisfy some constraints of the visualization system related to types of voxel adjacencies. As the voxel has a cube shape, it presents three types of adjacencies: by face, edge, and vertex. In figure 1, the voxel at the center of the middle slice is adjacent to: six voxels by face (figure 1.a), twelve voxels by edge (figure 1.b), and eight voxels by vertex (figure 1.c). Cumulatively, the voxel at the center of middle slice is adjacent to: six voxels by face, eighteen voxels by face and edge, and twenty-six voxels by face, edge and vertex [6].

The used visualization system requires voxel adjacencies by face so that the objects are correctly shown. Figure 2 gives a generic example of a slice, using voxel adjacencies by face. If the object representation has voxel adjacencies exclusively by edge or vertex, it will occur image distortions in the object visualization. Voxel adjacencies by face must be respected by the voxels inside the slice and by the voxels placed in neighbor slices.

3. THE VISUALIZATION ENVIRONMENT

A visualization system based on Iris Explorer software [13] was developed to show 3D surfaces and volumes of objects modeled by means of slices.

This system is composed of three modules: reading module, isosurfaceLat module and render module. Figure 3 shows a map linking these three modules and their details.

The reading module (dsckul) was developed with the support of DataScribe utility. It reads the voxel values of each slice, whose data set is a file, and sends the values to the isosurfaceLat module.

The isosurfaceLat module creates isosurfaces, which are surfaces of uniform value in 3D objects. The inner object surfaces can be visualized through holes or cut out object parts. The results from this module are sent to the render module.

The render module is responsible for showing the object and providing for its handling. Additionally, it allows to translate and rotate the object, to change lights, color, texture, brightness and transparency, and many other effects. Both isosurfaceLat and render modules are part of the Explorer module library.

In figure 3, the three map modules were expanded to show their details and handlers. The isosurfaceLat module has a threshold control that changes the image at the rendering phase. If the threshold value is less than the voxel values, the object is shown, otherwise it will not appear. If an object has voxels with different values, it is possible to select visible portions by turning the threshold dial.
Figure 1. Types of voxel adjacencies in 3D discrete space

Figure 2. Generical example of a slice
For a correct visualization, the modeling needs to include at least one empty voxel around the object.

4. A MODELING EXAMPLE

Several 3D modeling examples were developed such as: geometrical objects, 3D words, an airplane, and a human skull. The modeling of a human skull was made by hand, using a scale plastic model as reference. Some human skull views were drawn to help the development of the slices. The slicing of the skull resulted in 51 slices, each one containing 52 rows and 41 columns. The data set is a 3D matrix (51 X 52 X 41), containing value zero for empty space and value seven (arbitrary) for voxels belonging to the object.

The human skull model object was displayed on a SGI workstation. The rotation and translation of the object at the render module was performed in real-time. Figure 4 presents different views of the 3D human skull rendering.

This modeling did not consider the internal skull bones to simplify the manual slicing. It is possible to get real data from any 3D scanner (CT, NMR, 3D surface scanner, etc) and adjust them to be visualized on the system here described. Certainly, the quality of the object will be improved if real data is used instead of data made by hand.

5. SYSTEM APPLICATIONS

The system allows the construction of 3D objects based on manually organized data or data from 3D scanners. The data set, representing the 3D object, can be handled and transformed before being submitted to visualization. This allows the generation of special effects and the use of the system in several applications.

Some especial effects include:

- multiple cuts made by planes and irregular surfaces;
- selection of specific object portions for visualization;
- integral or partial visualization of object layers;
- selection and separation of object portions (object explosion).

The system can be used in applications such as:

- visualization of medical images [7,10,11,12];
- automatic creation of 3D facial models [1];
- surface reconstruction from unorganized points [5];
- sculpture based on 3D computer graphics [4];
- generation of object coordinates to be submitted to other visualization systems [16];
- modeling of 3D objects for virtual reality applications [2, 3].
Figure 3. 3D visualization map using Iris Explorer

Figure 4. Different views of a 3D human skull rendering
6. CONCLUSIONS

This paper has presented a technique for modeling 3D objects and a visualization environment. The modeling technique refers to volume graphics and uses object slicing. The visualization is based on modules extracted from the Iris Explorer library and on one module built and added to it.

A human skull was modeled and visualized, and some results were presented. The visualization environment assembles the object and allows its handling, as rotation for example, in real-time.

The system contains some restrictions due to the accepted type of voxel adjacencies (by face), which decreases the object precision in some cases.

Whereas the manual modeling is very simple for objects based on a few slices, it is arduous for complex objects. In this last case, it is recommended either the use of 3D scanners for getting data from real objects or the support of a tool to make easier the construction of the object.

The system described presents several application possibilities in problems that need surface or volume visualization.

REFERENCES


