An Interactive Voxel Data Manipulation System for Surgical Simulation

Eiji Kitagawa †, *Takami Yasuda* ††, *Shigeki Yokoi* †† *and Jun-ichiro Toriwaki* †

†Department of Information Engineering, School of Engineering, Nagoya University, Nagoya, 464-01, Japan

††School of Informatics and Sciences, Nagoya University, Nagoya, 464-01, Japan

Abstract

In this paper, we describe an interactive surgical simulation system by voxel manipulation in a virtual space. In this system, we can manipulate a 3D image on a computer screen as if we treated a 3D object in real world. In order to treat a 3D image provided as a 3D array with high level interaction and real time operation, we developed techniques such as fast representation of surface figure with polygons, a new data structure for 3D image description, interactive processing with mouse and real-time result evaluation method.

1. INTRODUCTION

The progress of computer power and the image quality of medical 3D images such as CT and MRI make it possible to interactively manipulate 3D images and to simulate surgical operations. Surgical simulation is useful for the safety of surgical operations, education of interns and informed consent for patients. In addition, with advance of graphics hardware, possibility of human-machine interface is advanced. Such as high resolution graphic display, mouse (or other advanced pointing devices), dial box and window system software are becoming more convenient tools. These interactive environments and high-performance graphic workstation can greatly increase the possibility of sophisticated interaction.

Recently, several studies about surgical simulation on a computer have been reported. Yasuda et al. developed a craniofacial surgical simulation system[1][2][3]. Suto et al. developed a surgical simulation system for spine orthopedic surgeries[4]. These systems can manipulate human bone image interactively. However, since it was implemented on a supercomputer, its interface was not satisfactory so that it was difficult to apply the system to clinical practice. Caponetti et al. also developed a system to simulate bone orthopedic surgeries[5]. This system could detect abnormal bony bending from two sheets of X-ray images for assisting doctors to make a surgical plan. However, it was limited to apply to straight bone such as femur and it could not manipulate 3D images that provided as 3D array.

We developed an interactive surgical simulation system based on virtual space manipulation. This system aims to manipulate voxel data in a 3D array as if we treated a 3D object in real world. We have developed the system to manipulate voxel type data interactively because medical 3D image such as X-ray CT images or MRI are usually provided as in the form of a 3D voxel data array.

2. SURGICAL SIMULATION SYSTEM

In this section, we describe the details of our surgical simulation system. For the purpose of high level interaction and fast data handling, we developed new methods and data structure for 3D data manipulation.

In this paper, we define the term 'object' as a set of voxels that represents human bones and the term 'object data' as information of 'object'. We also define the term 'virtual space' as a 3D array that represents a virtual world on a computer.

The surgical simulation system consists of three steps: preprocessing, simulation and evaluation (Fig.1).

Preprocessing: We represent a 3D surface of an object as a set of polygonal patches, since the graphic workstation used in our system has a special hardware to

IEEE International Workshop on Robot and Human Communication
0-7803-2002-6/94 $4.00 © 1994 IEEE

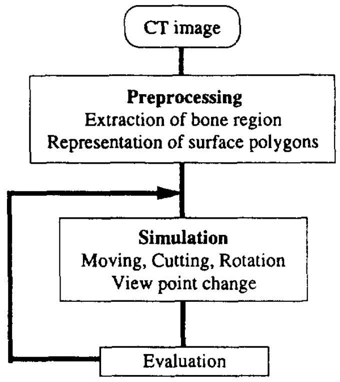

Fig.1 System flow of surgical simulation

generate polygons rapidly. First, we extract a bone region from original CT slices by the simple thresholding method and separate a region of interest in the segmentation process. Then we produce the object data including surface polygons for each part. We will describe the data structure for the object in section 2.1 and the surface representation in section 2.2.

Simulation: We can simulate surgical operations on a graphic workstation using display, mouse, dial box and keyboard. Several interactive manipulations such as cutting, moving, rotation and fusion operation are used for virtual surgeries. We will describe the detail of these operations in section 2.3.

Evaluation: One should evaluate the derived surgical plan to make sure if it is effective for the patients or to compare it with the alternatives. If we could not be satisfied with the result, we can go back the simulation process and make another plan again. Different evaluation methods are necessary corresponding to types of operations. We will describe the evaluation function in section 2.4.

2.1. DATA STRUCTURE

Object data involves its own information such as a 3D array that represents shape of the object, transformation matrix that indicates how to map the 3D array to the virtual space, the surface polygon list that represents the surface shape of the object and the center of gravity.

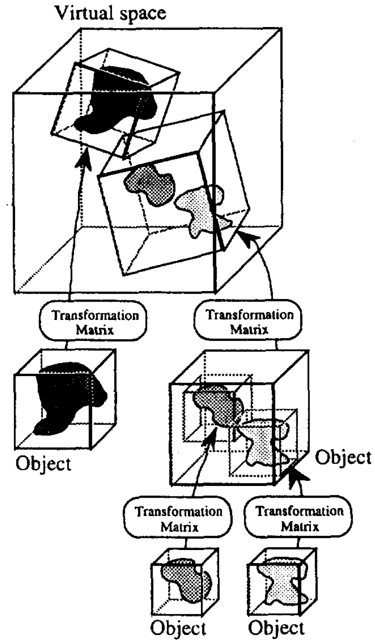

Fig.2 An hierarchic object data structure

Objects are represented by a hierarchic structure (Fig.2). Objects defined in a lower layer are mapped into object space on the upper layer with their own transformation matrix. Finally, all objects are mapped into the virtual space. Since the same label (ID value) is assigned to all voxels in one object, each voxel in the virtual space is distinguished by the ID value. The ID value 0 means the background.

Since recent graphic workstations have powerful ability to generate polygons rapidly, we use polygons to express the shape of objects on a computer screen. In order to display polygons of an object's surface, we use the same transformation matrix as the one used to map the objects into the virtual space.

This data structure has the following advantages.

(i) Manipulation such as moving and rotation is realized in a simple manner by changing the transformation matrix.

(ii) Since all kind of manipulation used in this system can be represented as the change of the transformation matrix and 3D array division, we can easily ob-

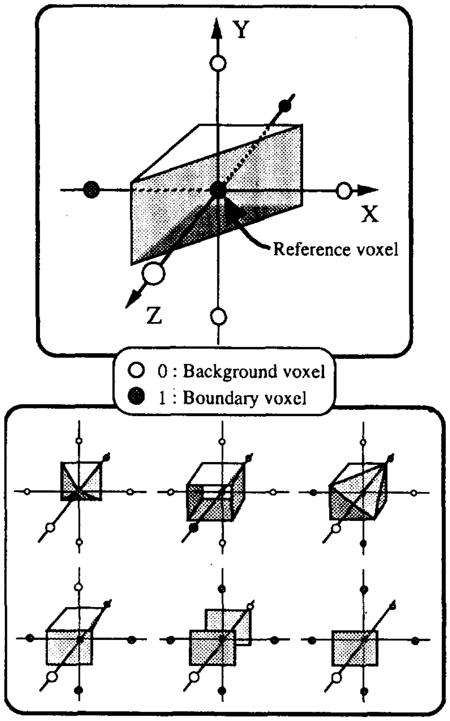

Fig.3 Production rules for surface polygons around a referenced voxel

tain the history of manipulation.

(iii) It is easy to represent hierarchic structure of human body in this system. It is also applicable to evaluation of the result of simulation with human body movement.

2.2. SURFACE RECONSTRUCTION

Several methods to generate surface polygons for objects from a 3D image have been already reported. Although generating six square polygons around a voxel is the most simple method (VC), the image quality of the result is not good. Marching-cubes algorithm (MC) [6] is known as the one to provide a high quality image. However, it requires generating many polygons for a voxel so that it takes a lot of computation time.

Therefore, we developed an new method to generate polygons around a voxel by using the 6-neighbor of it (Fig.3). This method is extended from the VC algorithm by considering the condition of the neighbor voxels concerning whether it lies on a boundary of an object or not. Configuration of 0 and 1's in the 6-neighbor (64 ways in all) determines the surface shape around a voxel. This algorithm works faster than MC since it considers less neighbors and depends on less configuration patterns than MC. In addition, this method can create smoother shape than VC since it uses neighbor voxels.

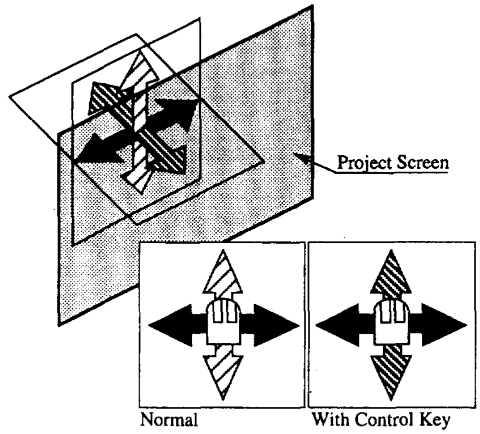

Fig.4 Control moving the direction by a mouse

We employed Gouraud shading for rendering patches obtained in the above scheme. This shading method needs the normal vector on each vertex of polygons to calculate its shade and color. Therefore, we estimate the normal vector of the voxel center in the same way as the MC algorithm. Then we estimate the normal vector of each vertex by the following equation.

$$\boldsymbol{n} = \boldsymbol{N} + \mathrm{w}\boldsymbol{p}$$

where $\boldsymbol{n}$ is the normal vector of a vertex, $\boldsymbol{N}$ is the normal vector of a voxel center, w is weight value and $\boldsymbol{p}$ is the vector from voxel center to vertex.

2.3. SURGICAL SIMULATION

At the present time, four fundamental operations such as Cutting, Moving, Rotation and Fusion are prepared in the system to simulate surgical operation..

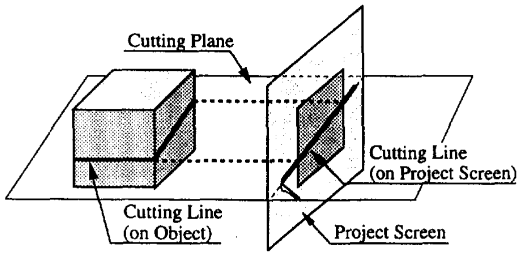

Fig.5 Cutting by plane

[*Moving*] In Moving operation, we can move an object to anywhere we want in the virtual space. Moving manipulation is performed by using a mouse, that is, by clicking the mouse on the object image to be moved and by moving the mouse. The clicked object is moved in real time in the virtual space following the mouse movement (Fig.4). Moving operation is carried out only by changing the transformation matrix of the object. We create a new matrix by multiplying parallel translation matrix with the original matrix.

[*Cutting*] We provide two methods for cutting. One is "cutting by plane", and the other is "cutting by sphere".

(i) Cutting by plane : This operation is specified by two steps. First, select a manipulated object by clicking a mouse on the project screen. Secondly, specify two points on the screen to determine a cutting plane. Cutting plane is perpendicular to the screen and passing through these two points (Fig.5). After determination of a cutting plane, the selected object is divided into two objects.

(ii) Cutting by sphere : This operation is specified by three steps. The first step is the same as cutting by plane. Second, specify two points on the screen to determine the center and the radius of the sphere. However, we have to specify the z-coordinate (depth) of the center point since it has a fixed depth initially. In order to recognize appropriate depth, the sphere is displayed translucently and is moved with mouse in real time. We move the sphere to appropriate depth in the third step. After fixing the position of the cutting sphere, the selected object is divided into two objects.

Cutting plane or cutting sphere is projected into object-coordinates and a voxel set corresponding to the object is divided into two sets. Then a new object data is created. Original object data is preserved as the object data of one of divided voxel data sets and new object data is generated for the other one. Both of object data have the same transformation matrix.

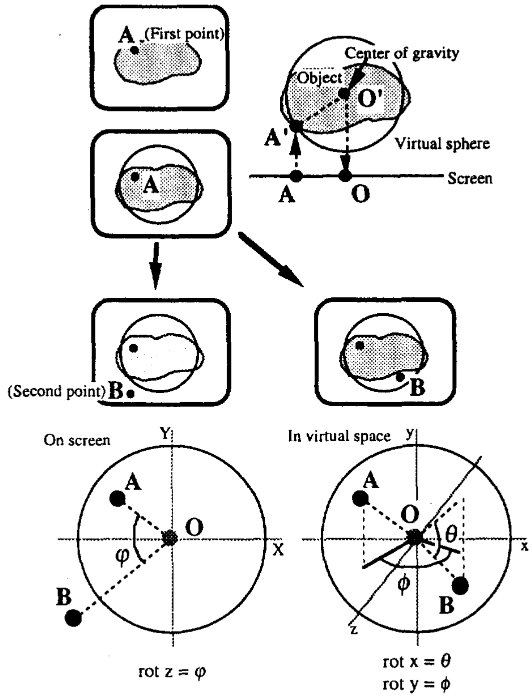

Fig.6 Rotation specification on screen

[*Rotation*] After selecting an object, we can rotate it interactively. A few methods for rotation controlled with 2D devices have been reported[7]. These conventional methods are not easy to rotate an object. We developed a new method that is more easier to recognize the amount of rotation.

Rotation is specified by three values, namely, rotation angles in x,y,z-axis . These values are specified by supposing a virtual sphere. A virtual sphere is a pseudo sphere with the center at the center of gravity of the object and its radius is not defined. To determine its radius, click a point of the object (point A in Fig.6). The radius is defined as the length between the point and the center of gravity of the object (distance A'O' in Fig.6). If the radius is too short, the value 1.0 is given to the radius instead. After determination of the center and the radius, the virtual sphere is drawn on the screen as an appropriate circle. To rotate the object, we have to specify another point on the screen (point B in Fig.6). When the second point is inside the circle, rotation values in x,y-axis are changed, and if it is outside the circle, the rotation value in z-axis is changed. In the case of inside specification, x,y-axis rotation values to overlap the first point with the second one are calculated. In the case of outside specification, the z-axis rotation value is equal to the angle AOB in Fig.6. Rotation is also executed simply by changing the transformation matrix of the object.

[*Fusion*] Fusion operation does not exist in real surgical operation. However, we provided this operation to cover our cutting manipulation's limitation. Since we do not consider the cutting depth and we treat the cutting plane as an infinite plane, an unexpected region might be cut off by the operation. So the fusion operation is provided to recover such regions. Fusion operation is simple: select two objects that we want to put together and click the fusion icon to execute.

2.4. EVALUATION

Evaluation is an important process to execute appropriate surgical simulation. We have to select a suitable evaluation method for the simulated surgical operation. For example, in hip joint orthopedic surgery the distance map is effective for the evaluation[8]. In this system, we provide a real-time distance map generation function.

The distance map is the distribution of the distance between two objects on bone surface. The distance between each objects is classified into pre-specified classes and mapped with a distinct color on the surface of an object. In the case of hip joint orthopedic surgery, the smaller the distance between the femoral head and the pelvis is, the better the surgical plan is.

We have developed a real-time distance mapping method. This method utilizes a 3D texture mapping function to map the distance distribution. The 3D texture is given as a 3D array. First we select the reference object of the distance computation. Each element of the 3D texture is filled with the value of the distance from it. This calculation is executed only once when the referenced object is selected. The 3D texture is mapped to appropriate position in the virtual space. When we move the object on which the distance is mapped, this 3D texture is mapped onto the object in real-time. If you need the distance more precisely, the larger 3D array should be prepared for the same area in the virtual space.

3. EXPERIMENTAL RESULTS

This system was implemented on IRIS Crimson with RealityEngine (About 60MIPS). The source program is written with about 10000 steps C code. We generally use a mouse and a dial box for an input device.

Figure 7 shows an example of surgical simulation process with this system. An original CT image is a hip joint part consisting of 128x128x68 voxels. In the figure left is pre-operative hip joint and right shows a post-operative one. We cut pelvis at the center line and also divide right part into pelvis and femur parts. Then we move and rotate the femur.

Figure 8 shows an example of real-time distance map function. The input image is a part of femoral head that has 64x64x64 voxels. In order to see the distance map easily, pelvis part is not displayed. Dark area shows that it is close to the pelvis.

4. CONCLUSION

An interactive surgical simulation system based on voxel manipulation is described. This system is implemented on a graphic workstation. We have developed a new method for making surface polygons, and employed a hierarchic data structure to represent 3D images and developed a method for manipulation dealing with this data structure. With these techniques, we could implement moving, rotation and fusion operation very easily. We have also developed an evaluation method basing upon the real-time distance map. However, the cutting operation is not fast enough at present. It is expected to be improved. We are intending to develop wider class of surgical operations considering soft tissue manipulation.

ACKNOWLEDGMENTS

Authors wish to thank Dr. M. Okada, Mr. Kenji Ando and colleagues in Toriwaki laboratory for helpful discussion. We owe helpful suggestions to Dr. R. Izumida (Department of orthopedic surgery, National Saitama Hospital). Parts of the work were supported by the Grant-in-Aid for Scientific Research, Ministry of Education and the Grant-in-Aid for Cancer Research, Ministry of Health and Welfare.

REFERENCES

[1] S. Yokoi, T. Yasuda, Y. Hashimoto, J. Toriwaki, M. Fujioka and H. Nakajima : A Craniofacial Surgical Planning System, Proc. NCGA Computer Graphics, pp.152-161 (1987).

[2] T. Yasuda, Y. Hashimoto, S. Yokoi and J. Toriwaki : Planning Support System for Craniofacial Surgery Using CT Images, IEICE(Japan), J70-D(11), pp.2134-2140 (1987).

[3] T. Yasuda, S. Yokoi and J. Toriwaki : Surgical Planning System NUCSS-V2 Containing a Function to Cut or Move Arbitrary Shape of Bony Blocks, IPSJ(Japan), 31, pp.870-878 (1990).

[4] Y. Suto, T. Kurokawa, T. Kojima, M. Kobayashi, K. Furuhata, T. Huang and K. Kikuchi : Clinical Application of Three Dimensional Image to Orthopedics, Medical Imaging Technology(Japan), 7(1), pp.16-21 (1989)

[5] L. Caponetti and A.M. Fanelli : Computer-Aided Simulation for Bone Surgery, IEEE CG&A, 13(6), pp.86-92 (1993)

[6] W.E. Lorensen and H.E. Cline : MARCHING CUBES : A HIGH RESOLUTION 3D SURFACE CONSTRUCTION ALGORITHM, Computer Graphics, 21(4), pp.163-169 (1987)

[7] M. Chen, S.J. Mountford and A. Sellen : A Study in Interactive 3-D Rotation Using 2-D Control Devices, Computer Graphics, 22(4), pp.121-129 (1988)

[8] X. Zhou, T. Yasuda, S. Yokoi, J. Toriwaki and R. Izumida : Computer Assisted System for Selection of the Optimal-Fit Artificial Stem in Hipjoint Orthopaedic Surgeries, IEICE(Japan), J76-D-II, pp.374-381 (1993)

Fig.7 Experimental results of surgical simulation

Fig.8 Experimental results of real-time distance map