AN INTERACTIVE CLINICAL INTERFACE FOR MR IMAGE-BASED COMPUTATIONAL MECHANICS MODELING OF THE HUMAN CARDIOVASCULAR SYSTEM

Tomoaki Hayasaka (1), Hao Liu (1), Ryutaro Himeno (1), Takami Yamaguchi (2)

(1) Division of Computer and Information Science, The Institute of Physical and Chemical Research (RIKEN), Wako, 351-0198, Japan

(2) Dept. Mechanical & Systems Engineering, Nagoya Institute of Technology, Nagoya, 466-8555, Japan

INTRODUCTION

There are three major steps in a computational biomechanics simulation: modeling, computation, and visualization. Of these, everyone would agree that modeling is the initial and most crucial of the steps. Using computational mechanics modeling for clinical applications requires special care. First, patient-specific modeling is mandatory, because of individual variation in anatomical structures and physiological conditions. It must be accurate, because patients' lives may depend on the system. It must also be quick, because the amount of time available in clinical situations is often severely limited, particularly in cardiovascular medicine. To achieve these goals, we developed an interactive clinical interface for MR image-based computational mechanics modeling of the human cardiovascular system.

ACCURATE MODELING

We are trying to utilize volumetric images to accurately model complex objects in 3-D space. However, medical images always have a blurred boundary, as well as severe noise and artifacts. This is why extensive anatomical knowledge and space perception ability are essential to construct a feasible model. In other words, an accurate model cannot be constructed without utilizing the experience and expertise of clinicians. Therefore, we need a system in which medical experts can fully interact with the modeling process.

Real Time Volume Rendering

One of the most advanced features of our system is that the operator can modify the model that is rendered over the reference volumetric images. Since medical images are difficult to segment automatically, and our design policy is to employ human experience and expertise for this purpose, we need a fast and direct rendering method. To fulfill this requirement, we selected a method called the "volume slicing" technique [1, 2], which fully utilizes texture mapping and alpha blending hardware to get results that are equivalent to those of the traditional ray casting method. Although texture mapping and alpha blending hardware used to be prohibitively expensive, it is very common in inexpensive consumer graphics adaptors nowadays. Using this hardware, the rendering time can be reduced surprisingly. It is at least 10 times, and sometimes even 100, times faster than existing software-based methods of volume rendering using the same CPU.

QUICK MODELING

In the RIKEN project, we have already started to build a geometrical model database using a dedicated MRI system that was recently installed for this purpose. This database can be used to provide a pre-constructed model template that can be fitted to the geometry of individual patients.

Multi-Resolution Mesh Editing

The system has a multi-resolution mesh editing feature, which can be used for quick fitting. This feature uses an algorithm similar to the one proposed by Zorin et al. [3], and is based on Loop's subdivision method [4].

Loop's method smoothes the surface of an object by successive division of the vertices of the polygonal mesh of the surface. It has been mathematically proven that the control mesh formed by this successive subdivision converges to a limit surface, known to be the box spline surface of the initial mesh. Hence, it is C^2 continuous at almost all points, which is a desirable feature when a computational mesh is formed inside it. The important point is that a smooth surface of arbitrary topology can be robustly obtained, with very simple operation and single representation.

The surface mesh formed by this system is composed of several resolution levels of mesh simultaneously. In this system, a user can make modifications at any level, in any order, and the system automatically updates the mesh of other levels according to the changes, while retaining unchanged details. For example, if we have a mesh built with four levels, and the second level of the mesh is modified, the finer levels and the coarsest level are updated by the system automatically and the details are preserved (Fig. 1).

IMPLEMENTATION

The system is implemented on a Linux PC, SGI Octane, and an SGI Reality Monster. The program runs seamlessly on these machines. We use GNU C++ and MIPS Pro C++ Compilers, Mesa and OpenGL for 3-D graphics, and FLTK as a GUI tool kit. All of these software components are open sourced or compliant with open standards, to avoid any proprietary limitations.

RESULTS

Modeling operations, such as scaling and rotating volumetric images or modifying polygonal meshes rendered over volumetric images, can be done at an interactive frame rate on all hardware. The user interface of this system is shown in Figure 2. We constructed a geometric model of the human aortic arch based on MR 3-D images (256 * 256 pixels * 112 planes) with this system (Fig. 3).

SUMMARY AND FUTURE WORKS

We have developed an interactive modeling system. The novel features of this system are real time volume rendering and multi-resolution mesh editing. Using these capabilities, a medical expert can make accurate models efficiently using his/her anatomical knowledge and heuristic ability, without requiring a background in computation.

At the moment, the system cannot construct a computational mechanics mesh within the model. In order to transfer a model to any mesh generator and to execute computation, we are implementing a feature to export mesh geometry in a standard exchangeable format, such as the IGES format.

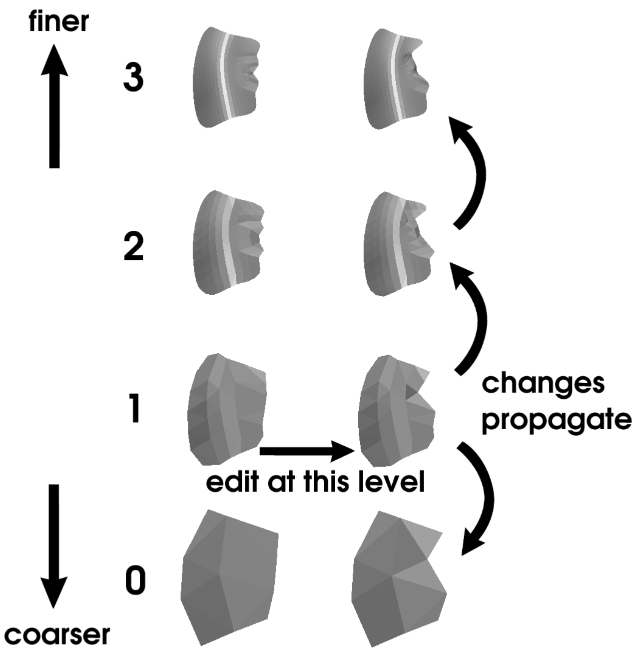

Figure 1. Multi-resolution mesh editing

REFERENCES

1. Zorin, D., Schröder, P., and Sweldens, W., 1998, "Interactive Multiresolution Mesh Editing," Subdivision for Modeling and Animation, SIGGRAPH 98 Course Notes
2. Cabral, B., Cam, N., and Foran, J., 1994, "Accelerated Volume Rendering and Tomographic Reconstruction using Texture Mapping Hardware," ACM Symp. on Vol. Vis.
3. Eckel, G., 1998, "OpenGL Volumizer Programmer's Guide," SGI Developer Bookshelf.
4. Loop, C., 1987, "Smooth Subdivision Surfaces Based on Triangles," Master's thesis, University of Utah, Department of Mathematics.

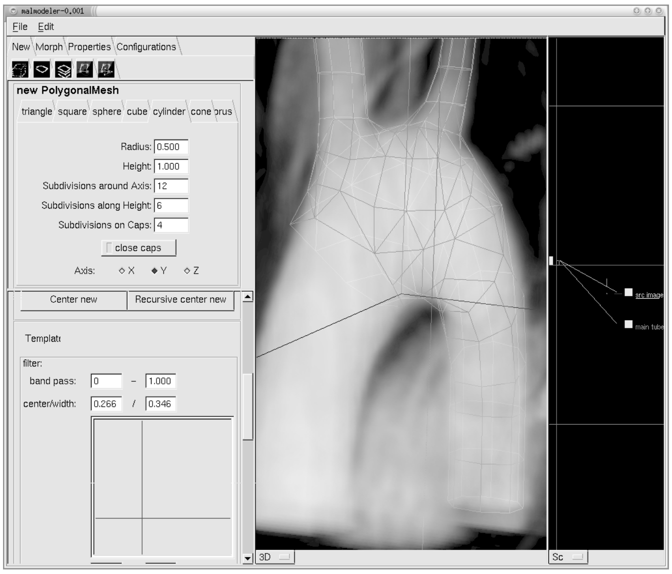

Figure 2. The system's user interface

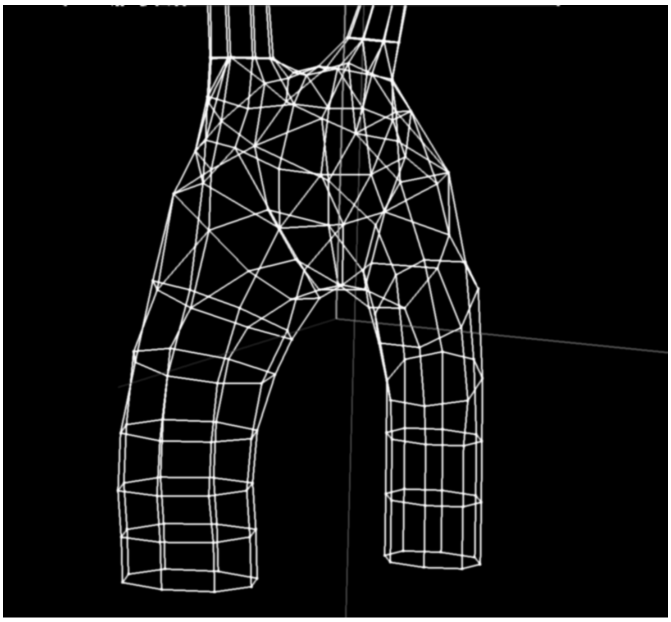

Figure 3. A sample model constructed with the system