Using Lattice for Web-based Medical Applications

Akira Wakita Takamichi Hayashi Takashi Kanai Hiroaki Chiyokura*

Keio University

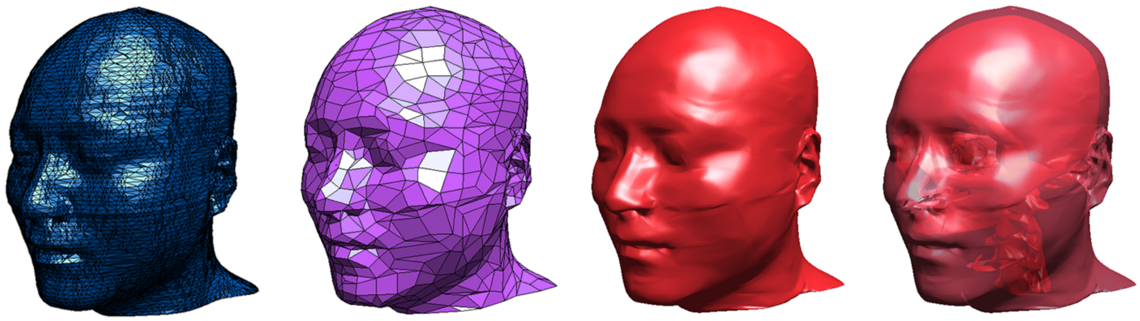

Figure 1: Example of a Lattice model reconstruction process and its medical application. Left: Initial dense polygonal mesh. Middle: Lattice mesh and Lattice surface reconstructed from a dense polygonal mesh. Right: a web-based medical application based on Lattice structure.

Abstract

This paper will describe a methodology for developing next generation web-based medical systems. Recent web-based technologies have created completely new possibilities to various medical information systems. Nevertheless, transferring 3D patient models through usual low band-width networks is difficult because of their large data size. We address this problem by making use of Lattice, an extension for X3D proposed at the Web3D Consortium, which represents high quality surface shape with small data size. Taking advantage of Lattice, detailed anatomical 3D models are rapidly transferred within a few seconds through low band-width networks. Anatomical Lattice models are semi-automatically generated from CT/MRI data. Our system also provide a multi-user functions which can be used as remote clinical conferences or instructing surgical procedures. Some practical applications of our system are also presented.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—*curve, surface, solid, and object representations*; *hierarchy and geometric transformations*; I.3.6 [Computer Graphics]: Methodology and Techniques—*graphics data structures and data types*; *standards*; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—*virtual reality*; G.1.2 [Numerical Analysis]: Approximation—*approximation of surfaces and contours*; J.3 [Life and Medical Sciences]: *medical information systems*

Keywords: Lattice, X3D, Medical Applications, Geometry Compression

*{wakita, t98761th, kanai, chiyo}@sfc.keio.ac.jp

1 Introduction

Recent web-based technologies have created completely new possibilities for various medical information systems. The Visible Human Project [10] distributes CT (Computed Tomography) and MRI (Nuclear Magnetic Resonance Imaging) data on the Internet and offers opportunities for the use of medical images through the Internet. Using the above data set, many research institutions have developed medical information systems [8, 11]. However each application itself is not opened to information sharing on the Internet. Thus to date, there is no efficient way to share 3D medical information and applications to address this problem, some researchers are trying to develop web-based 3D medical systems. Hendin et al. [4] have presented a method to realize volume rendering on the Internet by using VRML and JAVA technologies. Because volume rendering is one of the most popular methods to visualize 3D medical information, this method is readily acceptable to surgeons. John et al. [6, 7] have developed a suite of VRML based surgical simulators. They present a method to generate simple but effective surgical training tools. Holten-Lund et al. [5] used VRML to visualize surgery planning and diagnostics applications. They showed practical solutions for using VRML and JAVA technologies to address medical images on a PC by using both JavaScript interface and external authoring interface (EAI). These works enable surgeons to share the 3D medical information and applications on the Internet. Nevertheless, transferring 3D patient models through usual low band-width networks is difficult because of their large data size. Because data sets for medical use demand high accuracy, the average data size of a VRML model is usually at least 10-20 MB. This problem proves to be a strong deterrent for the evolution of web3D-based medical applications. A method of representing medical 3D models with small data size is an absolute requirement for realizing interactive and practical systems.

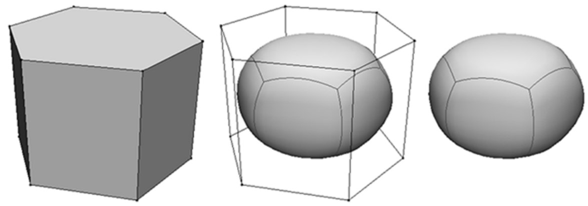

Figure 2: Lattice structure. The left model is Lattice mesh and the right is Lattice surface. Connectivities of these models have one to one correspondence.

We address this problem by making use of Lattice [14], an extension for X3D proposed at the Web3D Consortium [9], which represents high quality surface shape with small data size. By utilizing free-form surface techniques, high quality surfaces are transferred rapidly. What is more, users can control the level of rendering details by changing the tessellation number. Taking the above mentioned advantages of Lattice, interactive web-based medical systems can be realized. Reconstruction of 3D anatomical lattice models from CT/MRI data can be done semi-automatically and takes three steps. At the first step, we extract the outer and inner contours of CT/MRI images manually. This can be done by hand-tracing character points on each slice image. Next, we construct a polygonal mesh from contour slices by using NUAGES [3], a mesh reconstruction tool for medicine. Finally, we fit a free-form surface to a polygonal mesh by making use of a mesh simplification technique based on QEM (Quadric Error Metrics) [2, 12]. This step is done automatically and can be widely applied to any type of conversion from polygonal mesh to Lattice model. We use JavaScript (ECMAScript) and XVL Object Model to implement user interfaces and communicate with each node in the X3D scene graph. XVL Object Model is an authoring interface to access each node in a 3D scene. Our system also supports multi-user functions which enables the operations of one user to be reflected in another user's scene. This functions can be used as remote clinical conferences or instructing remote audience how to operate surgery.

We have developed simulators for cleft lip surgery [13], 3D displaying CT/MRI images, mandibular bone movement. All these simulators are based on Lattice Extension for X3D and work on the Internet. We have estimated the efficiency of our system by using the above simulator in clinical conferences, and the system has been found to be a practical tool for surgeons for the following reasons: First, users can transfer the detailed 3D models within a few seconds through low band-width networks. Second, because highly detailed 3D representations of a patient's model can be made readily available, users can now understand the details of the surgical procedure.

In section 2, we review the Lattice Extension for X3D briefly and mention why we chose it for our purpose. Section 3 presents each step of generating a Lattice model from CT/MRI images precisely. A method to interact with each node in an X3D scene is also described. Section 4 introduces some practical examples of medical simulators based on our system. In section 5, we evaluate our system and present some feedbacks from surgeons. This paper concludes with section 6 with some tasks and future works.

2 Lattice Extension for X3D

2.1 Concept of Lattice

Lattice [14] is a compact 3D-data representation with high quality surface shapes. Because Lattice is based on free-form surface techniques, the amount of data size is extremely small compared with polygonal meshes generated with surface tessellation. The structure of Lattice consists of Lattice surfaces and Lattice meshes. Each Lattice surface is a free-form surface data represented by Gregory patches [1]. Each Lattice mesh is a simple polygonal mesh consisting of connectivities (vertices/edges/faces), 3D coordinates and rounding weights. These two representations have one-to-one correspondence of connectivities. Based on an invertible rounding algorithm, Lattice meshes can be quickly transformed into Lattice surfaces and vice versa. By utilizing free-form surface techniques, high quality surfaces are transferred rapidly. Users can control the level of rendering details by changing the tessellation number according to the machine power or the use of the models. Because Lattice can be easily integrated into VRML or X3D files when being treated as Lattice meshes, and because its data size is extremely small, Lattice meshes are highly portable and easy to transfer even when using common low band-width networks. We show the Lattice structure in Figure 2.

Most medical information systems based on VRML do not care about the data size. Because data sets for medical use demand high accuracy, the average data size of VRML models is usually at least 10-20 MB. Therefore, transferring 3D patient models through usual low band-width networks is difficult. This problem proves to be a strong deterrent for the evolution of web-based medical applications. We choose Lattice to address this problem because it can display high quality shape with small data size.

2.2 Rounding Operations

Lattice surfaces represented with Gregory patches are generated by rounding operations. The rounding operation is a popular modeling method to generate a smooth surface from a simple polygonal mesh. We make use of the Doo-Sabin subdivision surface scheme to generate a Lattice surface from a Lattice mesh. In our rounding operation, we approximate the limit surface of Doo-Sabin subdivision surface with Gregory patches. The limit surface of Doo-Sabin subdivision surfaces is a set of C^1 continuous quadratic B-spline surfaces. However it is difficult to represent the control points of such surfaces explicitly. On the other hand, output of our rounding method is the control points for free-form surfaces. This is the most important feature of our rounding operation. Control points of Gregory patches can be calculated with simple linear transformations. This advantage makes it possible to reconstruct a Lattice mesh from a Lattice surface with inverse transformation. This operation is called the inverse rounding operation. A wide range of surface shapes can be also represented by specifying weighting parameters to each edge and vertex. Detailed information of the rounding operation can be referred to from our previous paper [14].

2.3 VRML97 and X3D Descriptions

Lattice models can be migrated to VRML97 by using Lattice mesh descriptions. Because a Lattice mesh consist of a polygonal mesh and a set of weighting parameters, `IndexedFaceSet` node can be used for its representation with VRML97 specification except weighting parameters. As for weighting parameters, `Switch` node is used. Using `PROTO`, we define both a flag node which states whether the Lattice mesh is rounded or not and the weighting parameters node. These nodes are inserted into a `choice` field of `Switch` node, where its `whichChoice` field is set to -1. This

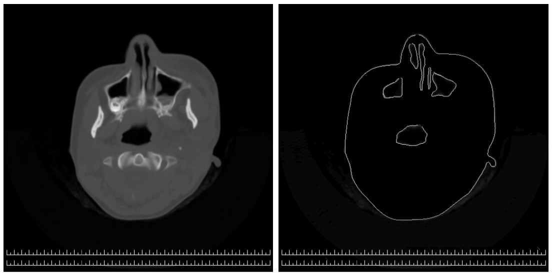

Figure 3: Extraction of contours from CT/MRI images.

enables browsers which do not support Lattice to ignore weighting information. Here, Lattice mesh results in its polygonal shape on the browser. Otherwise, the browser obtains weighting attributes from the `choice` field in order to round the polygon.

We have designed the DTD for Lattice which realize the same functions as above. It can be used as an extension for X3D and has been proposed in Web3D Consortium [9].

3 Lattice Model Reconstruction from CT/MRI Images

In this section, we show a semi-automatic method to generate 3D anatomical Lattice model from patient's CT data. Our method is robust and applicable to a variety of anatomical 3D modeling. Our method consists of three steps, which are extraction of contours from CT images, generation of a polygonal mesh from contour slices and fitting of Lattice surfaces to a polygonal mesh.

3.1 Extraction of Contours from CT/MRI Images

The first step is to extract the inner and outer contours of the target organ from the patient's original CT/MRI images. The data set we used is made of 100 slices, where each slice consist of 512 pixels square. After highlighting the target organ by threshold control, we specify the character points of the boundary manually, as shown in Figure 3. The contour data is automatically produced from the specified points by using our original software. This operation is applied to all slices and a stack of contours of the target organ is produced.

3.2 Generation of a Polygonal Mesh from Contour Slices

The next step is to reconstruct a polygonal mesh from a set of contour slices. We use NUAGES [3], a mesh reconstruction tool for medicine, for this purpose. NUAGES can generate polygonal meshes from contour slices automatically based on the delaunay triangulation technique. All we have to do is to generate the input data consisting of contour slices, where each slice consists of a series of polylines. An example of polygonal mesh reconstructed in this step is shown in the top left of Figure 4.

3.3 Fitting of a Free-form Surface to a Polygonal Mesh

The final step is fitting a Lattice surface to a polygonal mesh generated in the previous step. Because Lattice is based on free-form surface representation as we have shown in Section 2, we have to fit a free-form surface to a polygonal mesh. After the surface fitting, we can produce a Lattice mesh, a simpler but topologically identical model to the Lattice surface, by inverse rounding operations.

In this step, we extend the subdivision surface fitting method proposed by Takeuchi et al. [12]. In their method, mesh simplification based on QEM(Quadric Error Metrics) [2] is applied and a control mesh that approximates a Doo-Sabin subdivision surface is constructed. They also construct a network of B-spline patches guaranteeing G^1 continuity using a surface spline method.

We extend their method as follows: At the mesh simplification scheme, we use the points on the Lattice surface as evaluators of QEM and construct a control mesh for the Doo-Sabin subdivsion surface. As noted in [14], a vertex $\mathbf{L}_0$ of a Lattice mesh is mapped one-to-one to a vertex $\mathbf{P}_0$ of a Lattice surface. $\mathbf{P}_0$ is represented by $\mathbf{L}_0$ and its neighbors as follows:

$$\mathbf{P}_0 = (1-\omega_0)\mathbf{L}_0 + \omega_0 \sum_{j \in star(\mathbf{L}_0)} k_j \mathbf{l}_j \tag{1}$$

where ω_0 denotes a weight needed for rounding operations.

As shown in [2], at each edge collapse operation, the evaluation function Q^v for calculating an optimized position of $\mathbf{v}$ is:

$$Q^v(\mathbf{v}) = \sum area(f) Q^f(\mathbf{v}) \tag{2}$$

$$Q^f(\mathbf{v}) = (\mathbf{n}^T\mathbf{v} + d)^2 = \mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{b}^T\mathbf{v} + c \tag{3}$$

$\mathbf{n}$ denotes the normal vector of a face f and d is a scalar. $\mathbf{A}$ is a symmetric 3×3 matrix, $\mathbf{b}$ is a vector and c is a scalar.

As $Q^v(\mathbf{v})$ is a quadric function, a minimum value is found by simply solving a linear Equation $\nabla Q = \mathbf{0}$. We apply $\mathbf{P}_0$ to Equation (2), that is to solve $\nabla Q^v(\mathbf{P}_0) = 0$. By this extended simplification, a Lattice mesh is constructed instead of a control mesh that approximates a Doo-Sabin subdivision surface.

We assume this control mesh as a Lattice mesh and apply rounding operations. The rounded shape contains free-form surface information and approximates the initial polygonal mesh generated by NUAGES. This step is done automatically and can be widely applied to any type of conversion from a polygonal mesh to a Lattice model. Figure 4 shows an example of this step. The original human jaw model (6118 faces) is automatically converted to a Lattice mesh (929 faces).

3.4 User Interface

After generating a 3D anatomical model based on Lattice, the functions for each simulator and GUI to control it must be developed. We use JavaScript and XVL Object Model to communicate with each node in the X3D scene graph and implement simulator functions. XVL Object Mode is a scene authoring interface which enables the access of each of the nodes in the scene graph represented with X3D Lattice Extensions. It works not only as an internal interface such as SAI (Scene Authoring Interface) but also as an external interface represented by EAI (External Authoring Interface) to JAVA. Because users can use this interface with ECMAScript (i.e. JavaScript) through Dynamic HTML (assembly of HTML, CSS, DOM and Script), implementation is straightforward.

Some medical systems based on VRML utilize 3D widgets to construct GUI buttons by using the `ProximitySensor` node in the VRML scene. It causes the deterioration of frame rate, and in some cases causes a noticeable time lag due to the heavy computational load. For this reason, 2D GUI on an HTML frame is mainly used. Where 3D GUI is effective, e.g. incision with medical tools, we use the GUI made by 3D objects.

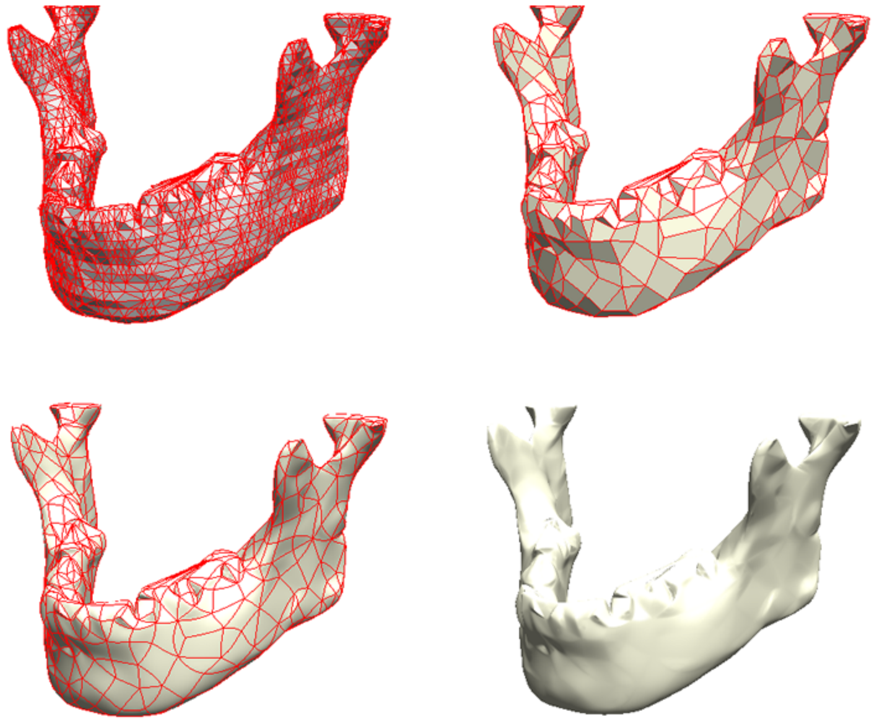

Figure 4: Reconstruction of a Lattice model from a polygonal mesh of human jaw. Top left: an initial dense polygonal mesh (6118 faces, 503KB). Top right: a Lattice mesh reconstructed from a dense polygonal mesh (929 faces, 63KB). Bottom left: a Lattice surface generated from a Lattice mesh using rounding operations. Bottom right: a Lattice surface rendered without boundary curves.

3.5 Multi-user Functions

We have implemented multi-user functions in order to take the best advantage of the Internet. By using these functions, one user's operations can be reflected in another user's display and scene graph. This is effective especially for the use of tele-medicine such as remote clinical conferences before surgery. It also can be used as a training tool instructing remote audience how to operate surgery.

4 Applications

4.1 Educational Cleft Lip Models

We have created a web-based educational tool for repairing cleft lip. Repairing cleft lip means plastic surgery for congenital defects of the lips that requires complex incisions and reconstruction. The user can interact with the model to view the incision lines designed by a skillful surgeon, the shape of the skin flaps created by the incision, the rotation of the skin flaps shown by 3D animations, and the model after completion of surgery. Because the viewpoint of the surgeons is highly restricted in practical surgery of cleft lip repairing, this application works well to understand the procedure of the operation from a variety of viewpoint. The data size of this medical information system is 62 KB. Therefore, the data can be downloaded in a few seconds and interactive manipulation is achieved. A snapshot of this application is shown in Figure 5.

4.2 Displaying CT/MRI Images with 3D Model Assistant

In the process of Computer Aided Surgery (CAS), medical doctors often need an engineer's assistance. Because engineers with little or no knowledge of the human anatomy are often required to treat medical images, there is a need for a system which assists them in understanding the medical images they handle. We have developed a system which displays CT/MRI images attached with the 3D models. As shown in Figure 6, the user can see the medical images as if they are the cross line section of the 3D model. The transparency of the skin is also controllable for better understanding. The data size of 3D model used in this application is 97KB, on the other hand origical polygon mesh is 18359KB.

4.3 Mandibular Bone Movement Viewing System

It is very important to understand the exact bone movement for better diagnoses and medical treatment. On the other hand it is difficult to observe the movement of bones without incisions, because bones are covered with soft tissues, e.g. skin and muscle. To address this problem, we have developed a 3D display system of mandibular bone movement. Because mandibular bone movement is very complicated, combinations of rotations and slides, the data is saved as key frame data captured per 11/100 second with two CCD cameras. The data size of 3D models and key frame amination is 243KB. This system can be used as surgical planning or informed consent. We show the snapshot in Figure 7.

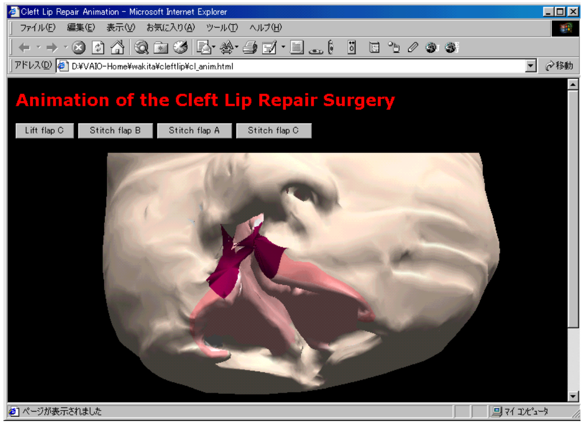

Figure 5: Educational tool for repairing cleft lip. User can intuitively understand the incision lines, displacement of skin flaps, and suturing.

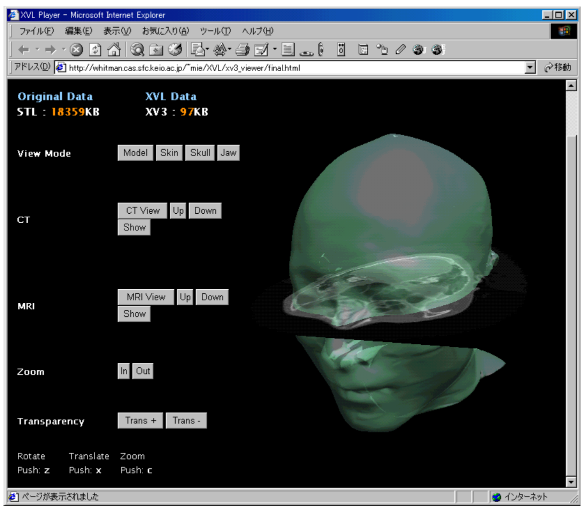

Figure 6: An application which displays medical images attached with 3D model.

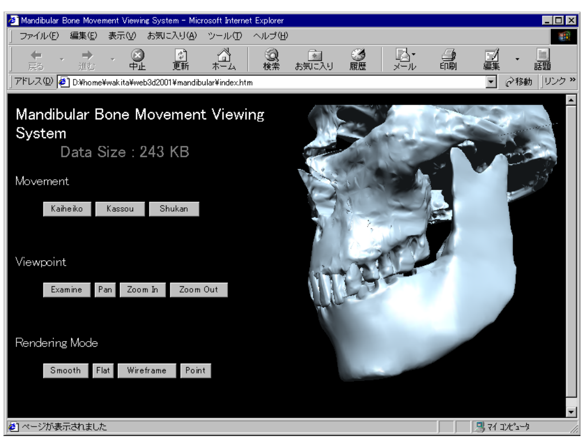

Figure 7: Mandibular bone movement viewing system

5 Results

We have estimated the efficiency of our system in clinical conferences held at the Department of Plastic and Reconstructive Surgery, Keio University School of Medicine. The system has proven to be a practical tool for surgeons for these reasons: First, users can transfer the detailed 3D models within a few seconds through low band-width networks. Today, PCs have become so powerful that most of 3D medical data can be processed interactively. On the other hand, band-width problem of the Internet still remains unsolved. The most practical solution is to make 3D data compact without decreasing its quality. Our effort using Lattice Extensions for X3D has proven to be the practical answer to this problem. Second, users can interact with a patient's 3D model with best level of details. By changing the tessellation number of a free-form surface, a user can control the resolution of rendering images. Another remarkable feature of our system is that it can be used as a tele-conference system with multi-user functions.

6 Conclusions and Future Works

In this paper, we have proposed an efficient methodology for web-based medical applications. Because Lattice can represent high quality surface shape with little amount of data, detailed anatomical 3D models are rapidly transferred. In addition to this efficiency, multi-user functions enable us to take the best advantage of the Internet connection. We have also presented the method of generating Lattice meshes from polygonal meshes automatically. This method is applicable to any other use of Lattice conversions. Some surgical simulators were also presented and proved to be useful for the practical surgical use.

The future work includes enhanced multi-user functions to chat with collaborators or to make text annotations dynamically into 3D scenes.

Acknowledgements

Authors would like to thank Daigo Tanaka from Keio University School of Medicine for helpful discussion and appropriate advise of medical field, Hiroshi Toriya from Lattice Technology for giving us an opportunity for applying Lattice Extension for X3D to Medicine, all people from Cyberworld Modeling and Medical Application Project in Keio University.

References

[1] Chiyokura, H., and Kimura, F. Design of solids with free-form surfaces. In *Computer Graphics (SIGGRAPH 83 Proceedings)* (1983), ACM Press, New York, pp. 289–298.

[2] Garland, M., and Heckbart, P. S. Surface simplification using quadric error metrics. In *SIGGRAPH 97 Proceedings* (1997), ACM Press, New York, pp. 209–216.

[3] Geiger, B. Three-dimensional modeling of human and its application to diagnosis and surgical planning. Tech. Rep. 2105, Institut National de Rechercheen Informatique et Antomatique, (France), 1993.

[4] Hendin, O., John, N. W., and Shochet, O. Medical volume rendering on the WWW using JAVA and VRML. In *Medicine Meets Virtual Reality 1998* (1998), IOS Press, pp. 34–40.

[5] HOLTEN-LUND, H., HVIDTFELDT, M., MADSEN, J., AND PEDERSEN, S. VRML visualization in a surgery planning and diagnostics applications. In *Web3D-VRML 2000* (2000), ACM Press, New York, pp. 111–118.

[6] JOHN, N. W., AND PHILLIPS, N. Surgical simulators using the WWW. In *Medicine Meets Virtual Reality 2000* (2000), IOS Press, pp. 146–152.

[7] JOHN, N. W., PHILLIPS, N., VAWDA, R., AND PERRIN, J. A vrml simulator for ventricular catheterisation. In *Eurographics UK* (1999).

[8] KOCH, R. M., GROSS, M. H., CARLS, F. R., VON BÜREN, D. F., FANKHAUSER, G., AND PARISH, Y. I. H. Simulating facial surgery using finite element models. In *SIGGRAPH 96 Proceedings* (1996), ACM Press, New York, pp. 421–428.

[9] LATTICE TECHNOLOGY. Lattice extension for X3D. http://www.web3d.org/TaskGroups/x3d/lattice/LatticeProposal.html, 2000.

[10] NATIONAL LIBRARY OF MEDICINE. *The Visible Human Project*. http://www.nlm.nih.gov/research/visible/visible_human.html, 1995.

[11] OAK RIDGE NATIONAL LABORATORY. *Virtual Human Project*. http://www.ornl.gov/ rwd/newVH.html, 1999.

[12] TAKEUCHI, S., KANAI, T., SUZUKI, H., SHIMADA, K., AND KIMURA, F. Subdivision surface fitting with QEM-based mesh simplification and reconstruction of approximated B-spline surfaces. In *Pacific Graphics 2000* (2000), pp. 202–212.

[13] TANAKA, D., KOBAYASHI, M., CHIYOKURA, H., NAKAJIMA, T., AND FUJINO, T. Web-based educational tool for cleft lip repair using XVL. In *to appear in Medicine Meets Virtual Reality 2001* (2001), IOS Press.

[14] WAKITA, A., YAJIMA, M., HARADA, T., TORIYA, H., AND CHIYOKURA, H. XVL : A compact and qualified 3D representation based on lattice mesh and surface for the internet. In *Web3D-VRML 2000* (2000), ACM Press, New York, pp. 45–51.