

Real-time Biomechanically-based Muscle Volume Deformation using FEM

Qing-hong Zhu, Yan Chen, and Arie Kaufman

*Center for Visual Computing
State University of New York at Stony Brook
Stony Brook, NY 11794-4400
USA*

Abstract

This paper presents a voxel-based biomechanical model for muscle deformation using finite element method (FEM) and volume graphics. Hierarchical voxel meshes are reconstructed from filtered segmented muscle images followed by FEM simulation and volume rendering. Physiological muscle force is considered and linear elastic muscle models for both static and dynamic cases are simulated by FEM. Voxel-based wireframe, polygon surface rendering and volume rendering techniques are applied to show real-time muscle deformation processes as well as realistic animations.

Keywords: voxel-based, biomechanically-based muscle modeling, real-time muscle simulation, finite element method, hierarchical representation, volume graphics, volumetric deformation.

1 Introduction

Human figure modeling, rendering and animation have been one of the primary areas of research in computer graphics since the early 1970's. In medical applications, such as surgery planning, we are more concerned with how it works as well as what it looks like. As a step to support these applications, we present a new approach which makes use of a sound biomechanical FEM model and voxel-based modeling and rendering.

Our approach involves the following steps:

- (1) Set up a hierarchical voxel mesh from a filtered dataset, reconstructed from a sequence of scanned image of the muscle. This is a preprocessing step for building the muscle model (See Section 3);
- (2) Employ a 3D voxel-based FEM system with

a biomechanical model to simulate both static and dynamic deformations. All deformations are performed interactively on volumetric meshes (See Sections 4, 5);

- (3) Apply various volume graphics techniques for real-time rendering of the deformed muscle data (See Section 6).

1.1 Anatomy-based vs Biomechanically-based

Recently there has been a trend to model the human figure in an approach similar to the one taken in artistic anatomy - by analyzing the relationship between exterior form and the underlying structures responsible for creating it. Keeve et al.⁸ developed an anatomy-based facial tissue modeling for surgical simulation using FEM. However, they only considered a static model, and their nonlinear design is still under development. Scheepers et al.¹³ used ellipsoids to represent muscle bellies. They also built models for Fusiform muscles and multi-belly muscles. Wilhelms et al.'s model¹⁷ consisted of individual muscles, bones, and generalized tissues covered by an elastic skin. Muscles were considered as deformable objects, discretized cylinders lying between fixed origins and insertions on specific bones.

Despite these innovations, these models lack medical proof and their images are imprecise. Instead, we turned to a biomechanically-based approach. First, we extracted information from a volumetric dataset, such as slices of CT or MRI, through the use of image processing. Then we used biomechanically-based modeling, in which the behavior of objects is determined via simulations of biomechanical laws and math equations, to determine how bodies move and change shape over

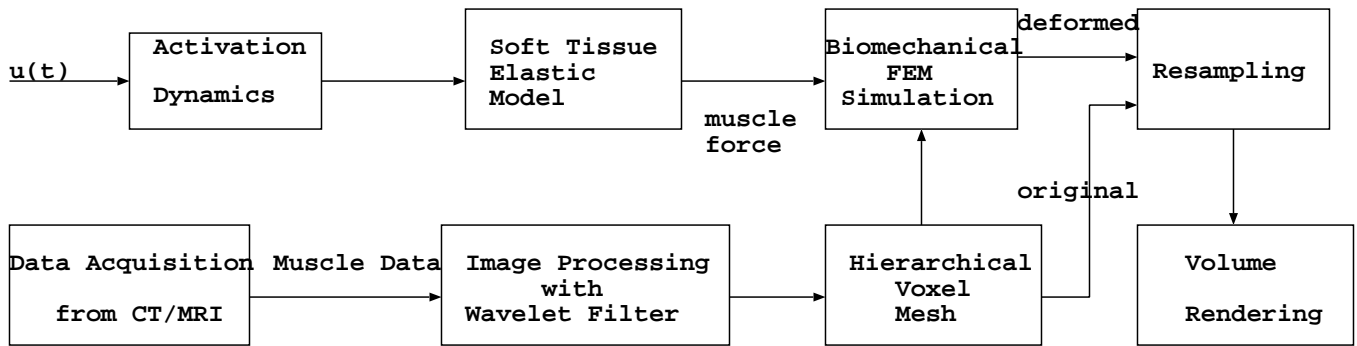


Figure 1: Pipeline of biomechanically-based muscle volume deformation simulation system

time.

Anatomy-based modeling may be useful for entertainment and art, but it is not suitable for medical applications. Biomechanically-based volume visualization, which applies volume graphics to real medical applications – including surgical simulation, treatment planning, and outcome prediction based on scanned images and biomechanical models – can give convincing simulation results.

1.2 Surface based vs Volume Based

In the computer graphics field, there is growing interest in complex, physically-based models which make deformation and behavior realistic. Most existing models are surface-based, where physical principles are applied to surface primitives or spline control points. However, these surface approaches are insufficient for simulating the effects of manipulation and modeling on object interiors. For example, in surgical simulation, we may wish to visualize a tumor through the surrounding brain tissue or simulate cutting through the deformable brain tissue in order to extract the tumor.

In addition, in many medical applications, there is a need to model both rigid bodies and deformable soft tissues. Voxel-based object models provide a way to represent the interior structure of heterogeneous, deformable tissues. It is also possible to use 3D medical image data directly with this representation, without requiring the extraction of approximate edges and surfaces.

This paper investigates physically realistic manipulation of voxel-based objects. There are several challenges for volume deformation of soft tissue objects. First, we need to build a 3D dynamic model based on plastics theory, which has traditionally proved difficult in biomechanical research

for human muscle, and where little work has been executed. The second is speed; the classical 3D elastic model is composed of time dependent partial differential equations which generally need to be evaluated by iteration – a major setback for speed. There are some ways to speed up the process, such as precomputing, using a two-stage dynamic deformation.

1.3 Mass-Spring Model vs FEM

Generally, two modeling methods are used for physically-based modeling of the human body: a mass-spring model and FEM model. The mass-spring model is simple and fast. But it is more difficult to control and less accurate than the finite element method¹². Also mass-spring model is considered as a C^0 -continuous finite element, while with FEM, we can achieve C^1 -continuous, and even C^2 -continuous at a higher computation cost⁹.

FEM is a well established engineering method. It is more accurate than mass spring in describing physical deformation. There is growing interest in computer-aided surgery, and in computer graphics and animation in using FEM. Terzopoulos et al.¹⁶ used finite-differences method and FEM in modeling physical deformations based on the elasticity theory. They modeled both rigid and deformable objects and considered physically realistic interactions and animation. The dynamic model produced simulation and animation. Tabatabai et al.¹⁵ presented the principal methods for volume rendering of non-regular grids as used in FEM.

More relevant work by MIT people include: Pieper et al.¹² implemented a system for computer-aided plastic surgery. An automatic mesh generator was used to generate the FE model. For visualizing the FE model, they subdivided the outer face of each

element into micropolygons. This method greatly reduced the number of nodes which constitute the FE matrix and thus reduce computational time. Chen et al.² simulated a 3D biomechanical FEM. In their model, skeleton kinematics and physiological effect were considered; twenty node brick FEM mesh and a user interface were developed. Apparently, the model is much less accurate in describing and rendering real muscle shape. And in both of their systems, volumetric cases were not implemented.

Bro-Nielsen et al.¹ introduced condensation, the use of a linear matrix system for simulation and the use of a selective matrix vector multiplication for surgery simulation using FEM. They had achieved real-time volumetric performance for 250 nodes. However, it is difficult to obtain high quality image with such a low resolution.

1.4 Hierarchical Volume Graphics

In volume-based modeling, we simulate the physically-realistic behavior of volumetric objects which include many million of volume elements. For real-time interaction, a fast algorithm is usually imperative for trading off between physical realism and speed. This is very challenging work due to the complicated 3D physical model and computation costs, as well as the large memory requirement for computing and rendering. The multiresolution method or level-of-detail (LOD) can accelerate both the simulation and rendering processes without great loss of image quality. Haley and Blake⁶ presented a new algorithm for efficient incremental rendering of volumetric datasets. They used the efficient shear-warp factorization method and represented the volumetric data using a hierarchical data structure which provides for the incremental classification and rendering of volumetric data..

Hong et al.⁷ presented an interactive virtual colonoscopy method, using a physically-based camera control model. It is considered as a pioneering work of volume graphics and physically-based modeling, however, their physical models deal with a rigid object which is governed by ordinary differential equations. For soft tissues, we need to develop an advanced elastic model. Gibson⁴ proposed the use of a voxel-based data representation not only for visualization, but also for modeling objects and structures derived from volumetric data. More recently, Gibson⁵ suggested a 3D chainmail algorithm which enabled fast deformation of volumetric objects. No physical models are included in her algorithms.

For volume rendering of the deformable objects, two methods are used. One method is direct rendering of the irregular deformed meshes in the physical space (Kaufman et al.¹⁰, and Fruhauf³). The other approach resamples irregular meshes into a rectilinear grid and subsequently visualizes it with well-developed algorithms for rectilinear data. We used in our system the latter approach.

2 System Overview

In the following sections we present a human muscle model that can enable prediction of deformation after a force is employed. In contrast to prior approaches, our system is a combination of hierarchical voxel mesh generation algorithms, physically-based biomechanical modeling using FEM and real-time volume rendering. We demonstrate our system on the human anconeus muscle. The Pipeline of our system is illustrated in Fig. 1.

In summary, we improve on earlier models in four new ways:

- Implement a muscle biomechanical model which is sound and well recognized by medical experts;
- Generate muscle voxel mesh in level of detail, using smoothing Gaussian filter;
- Construct an elastic soft tissue model for muscle deformation; techniques that specify physically-based behavior of both object surfaces and interiors are realized;
- Apply volume graphics and volume rendering in the elastic soft tissue model for the first time.

3 Biomechanical Model for Muscle Movement

Stern¹⁴ devised a mathematical model incorporating mechanical and physiological parameters of an idealized bone-muscle system, where the movement of a limb is under the action of a muscle. The model considers two aspects of muscle physiology: the effect of shortening (or lengthening) on tension development and of length on tension. His model employs the following assumptions:

- (1) At any moment in time during the contraction of a muscle, its exerted force depends only on its level of activation, instantaneous length and instantaneous velocity of contraction;
- (2) The idealized muscle is parallel-fibered, crossing one joint, and with point attachments;

- (3) The effort of moving the muscle is disregarded;
- (4) The tendon and contractile element are inelastic;
- (5) When comparing muscles with different sites of attachment on the moving limb segment, the effect on the moment of inertia of the segment is ignored.

Given the above, the force produced by a muscle at maximum activation obeys the following expression (refer to Fig. 2):

$$P = \left\{ \frac{(P_0)_l/a + 1}{\frac{1}{K_i} \frac{1}{(B/C+2+C/B)} \frac{\sin(\alpha_i \dot{\alpha}_i)}{j \cdot Y \cdot K_l} + 1} - 1 \right\} a \quad (1)$$

where

K_i = the straight line distance from one attachment to the other divided by the sum of the distances from the joint to each attachment point.

K_l = the value of K_i when the joint is positioned so that the length of the contractile tissue is L_0 .

j = the value of b expressed in muscle lengths per sec.

α_i = the angle between the bones.

$\dot{\alpha}_i$ = the angular velocity of movement.

Y = the fraction of the distance A_i devoted to contractile tissue when the joint is positioned so that the contractile tissue is at L_0 .

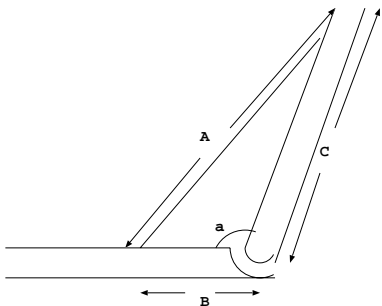


Figure 2: *Schematic drawing of a parallel-fibered one-joint muscle with point attachments.*

Fig. 3 is a 2D image of a cross-section of the arm from VHD that includes the anconeus, which is a small and short muscle outside of the elbow joint.

4 Building a Hierarchical Voxel Mesh

In this part, image processing techniques and hierarchical method are used for the generation of voxel meshes. Our data source is the CT anconeus

dataset from Visible Human Dataset (VHD)¹¹. We used 71 slices, with dataset resolution of 140x220x71.

The anconeus muscle image is generated through manual segmentation. Fig. 4(a) shows a muscle image through manual segmentation. The algorithm uses a multi-scale filtering (wavelet) and fuzzy based image segmentation approach. In this method, the wavelet filtering technique is utilized to automatically find the optimal segmentation threshold, and the classification of image is performed by means of multi-threshold and fuzzy clustering techniques. This method is used to segment the anconeus muscle areas from biceps and triceps of the VHD to reconstruct the 3D muscle volume. Fig. 4(b) uses a ray casting algorithm to show the muscle image obtained from this segmentation method. Muscle's reconstructing voxel meshes are based on the segmented images.

Then, a hierarchical voxel mesh is reconstructed and serves as a multiresolution volumetric approximation of the original muscle data. For each level of detail, first, we read the muscle raw file generated from manual segmentation, then after filtering, construct the voxel mesh based on the threshold value of the muscle image. The basic operation consists of constructing the voxel mesh inside of the 3D muscle data and giving up other information outside of it. In order to carry out the FEM simulation, we need to build the voxel mesh according to the data structure determined by FEM simulator which contains 3D geometric coordinates and topology information between each voxel.

In order to obtain a high resolution image, each voxel is subdivided into micro-voxels by the octree method. Therefore, the hierarchical voxel meshes are produced. Based on the different resolution requirement of the application, we construct different level of details of the muscle meshes. Because it takes a very long time to simulate high-resolution muscle deformation using 3D FEM, it is advantageous to have muscle shape change in low and medium-resolution and apply FEM simulation on it. With the muscle shape deformation at low, medium-resolution, it is then relatively easy to predict high-resolution deformation image by comparison, mapping and interpolation. Similarity theory, which is widely used in engineering design, is used to compare the deformation picture between low and medium-resolution and high-resolution images. Pieper et al.¹² used another approach by carrying out FEM simulation with a smaller number of nodes and elements and then subdivided each element into mi-

cropolygons to create a high-resolution image.

The voxel mesh is used in the rendering as well, so we can integrate real-time simulation and rendering based on the same mesh structure.

5 FEM Simulation for Muscle Deformation

We have developed our algorithms for both static and dynamic systems based on elastic theory and finite element principles. In the following, we discuss our FEM algorithms.

5.1 Theory:

In our model, we assume the muscle is behaving as an elastic soft tissue. We consider it a 3D muscle with an 8-node 3D brick element which is equivalent to the voxel structure in volume graphics. In the problem-solution considered here, the linear analysis conditions that are required are:

- The displacement be infinitesimally small so that the equilibrium of the body can be established with respect to its unloaded configuration;
- The stress-strain material matrix vary as a function of X , Y , Z , but as a constant otherwise.

To calculate the response of the muscle, we establish the governing differential equations of equilibrium for both the surface and interior of the muscle, which then have to be solved subject to the boundary conditions. In the following, we discuss two basic systems: static and dynamic, used in our model.

5.2 Static System:

The linear elastic model can be obtained by the general elasticity equilibrium equation. The effect of body forces is ignored and the system equation can be written as:

$$K\vec{x} = b \quad (2)$$

where \vec{x} is the displacement vector and b is a nodal load vector. The coefficient matrix K is a stiffness matrix. The boundary conditions of the system are given by muscle nodes which belong to the FEM mesh. By solving the linear sparse system we obtain displacements for all the nodes of the FEM mesh. These displacements are implemented into the graphical modeling system. The muscle shape changes can be visualized using different graphical techniques.

5.3 Dynamic System:

Based on Lagrangian dynamics, the deformable model equations of motion can be expressed in 3D vector form by the second-order ordinary differential equations:

$$M \frac{\partial^2 \vec{x}}{\partial t^2} + C \frac{\partial \vec{x}}{\partial t} + K(\vec{x})\vec{x} = \vec{f}(\vec{x}, t) \quad (3)$$

where \vec{x} is a $3n$ vector of nodal displacement, M , C and K are $3n \times 3n$ matrices describing the mass, damping and stiffness between nodes within the volumetric object respectively, and f is a $3n$ vector of forces applied to each node.

The solution of the displacement, velocity and acceleration at time $t + \Delta t$ can be obtained by using center and Euler finite differences written by Lagrangian equation for the full system as:

$$\frac{\partial^2 \vec{x}}{\partial t^2} = (x_{t+\Delta t} - 2x_t + x_{t-\Delta t})/\Delta t^2 \quad (4)$$

And

$$\frac{\partial \vec{x}}{\partial t} = (x_{t+\Delta t} - x_{t-\Delta t})/2\Delta t. \quad (5)$$

Let's define,

$$K^* = K(x_t) + \frac{M}{\Delta t^2} + \frac{C}{2\Delta t} \quad (6)$$

$$u_t = (x_t - x_{t-1})/\Delta t. \quad (7)$$

$$f_{t+\Delta t}^* = \left(\frac{1}{2\Delta t}C + \frac{M}{\Delta t^2}\right)x_t + \left(\frac{M}{\Delta t} - \frac{C}{2\Delta t}\right)u_t + f_{t+\Delta t}. \quad (8)$$

Substituting (4), (5), (6), (7) and (8) for (3), we can obtain,

$$K^*(x_{t+\Delta t}) = f_{t+\Delta t}^* \quad (9)$$

Consequently, the explicit procedure evolves the dynamic solution from the given initial conditions x_0 and u_0 by solving a time sequence of static equilibrium problems for the instantaneous configurations x_{t+1} . Thus, the original nonlinear partial differential equations have been reduced to a sequence of sparse linear algebraic equations.

In order to solve the sparse linear system equations quickly, we assume $K_t^* = K$ is time-variant; thus the matrix decomposition solver need only perform a single initial decomposition of K . which significantly reduces the total amount of computation required. Furthermore, the mass matrix and damping matrix are generated at each node by using diagonal damping and mass matrices,

$$M_{ii} = \frac{1}{3}\rho V, C_{ii} = \beta M_{ii} \quad (10)$$

where ρ is the mass-density, β is a scaling factor and V is the volume.

The algorithm is as follows:

- (1) Initial calculations:
 - Form 3D brick stiffness matrix K , mass matrix M , and damping matrix C ;
 - Calculate the initial displacement, velocity and acceleration: $x^0, \dot{x}^0, \ddot{x}^0$;
 - Store matrix value in memory;
 - Calculate the static deformation based on frontier solver.
- (2). Precomputing:
 - Select time step size Δt and calculate integration constant;
 - Derive dynamic difference scheme based on the static deformation and its matrix;
 - Store the matrix in the one dimension form using back substitution.
- (3). Time loop:
 - Calculate new sparse matrix based on the stored matrix;
 - Solve for displacements at time $t + \Delta t$ using explicit differential scheme based on the stored matrix values (K, M, C);
 - Calculate new velocity and acceleration at time $t + \Delta t$.

Our FEM algorithms are integrated with wireframe and shaded polygon graphics so that real-time display is realized in our FEM simulation system. The algorithms are written with C/OpenGL. In the following section, we review several graphical techniques and see how we integrate them together.

6 Visualizing Volumetric Muscle Deformation

We have developed three visualization techniques to visualize our deformed muscles: wireframe, shaded polygons and volume rendering.

3D wireframe is the fastest but least realistic form of display. Objects are drawn as though made of wire, with only their edges showing. The visible (within the view volume) portions of all edges of all objects are shown in their entirety, with no hidden-edge removal. All edges attributes affect screen appearance in their designated way in this model. We use this model with OpenGL in Fig. 6.

Shaded polygons display filled areas and polyhedra in a more realistic fashion. The addition of shaded areas to the rendering process increases the

complexity significantly, because spatial ordering becomes important. Figs. 7 and 9 are rendered by using OpenGL in this way. Both wireframe and polygon shadow models are integrated into FEM routine to create a real-time simulation and display system.

Volume rendering is the process of displaying 2D composited images of a 3D scalar field. Here, the scalar field is the density. After deformation, the data are located on a curvilinear grid, then we re-sample the data at the points of a regular grid. We take the approach of resampling and volume rendering the regular grid instead of directly volume rendering the curvilinear grid.

With the condition of FEM linear analysis (Sec. 5.1), the displacement should be infinitesimally small. Thus, the changes in x , y and z coordinates of voxels are quite uniform, i.e., there is no cells of drastically different sizes. Since resampling can be performed only once with multiple rendering, the whole process is still quite efficient. In our implementation, we compute the isosurfaces as well as display integrals of density along rays by assigning density only to certain isosurfaces, and then ray trace the results (which amounts to creating different volumes to be displayed). Fig. 5 is a filtered image of the original anconeus volume data which includes 14,578 voxels, rendered by ray tracing. We also use it in Fig. 8.

7 Results

The anconeus muscle from VHD¹¹, as an example, is used to demonstrate our techniques. The simulation and rendering were carried out on a SGI Power Onyx/RE2, R10000, 195MHz, 640MB.

Fig. 6 and Table 1 represent reconstructed hierarchical meshes based on the segmented 3D muscle image. The meshes are constructed with 3D eight-node brick elements which are equivalent to the voxels of volume graphics. The muscle geometrical coordinates and topological information are obtained by this method. Fig. 6 shows the hierarchical muscle voxel model in five different levels of detail with binary increment of the voxel size. (refer to Table 1). The meshes produced here can be used for real-time simulation and rendering. The more voxels, the more accurate the muscle image is. However, it is very slow for FEM dynamic simulation without preprocessing of the high-resolution muscle mesh. The hierarchical structure is very helpful for predicting real muscle deformation through FEM simulation with low and

medium-resolution muscle meshes.

Table 1: *Hierarchical voxel meshes for both 3D finite element simulator input and volume rendering*

Voxel Size	No of Voxels	No of Nodes	Fixed Points	Loaded Points
16 x 16 x 16	28	91	4	4
8 x 8 x 8	219	450	4	4
4 x 4 x 4	1840	2664	9	4
2 x 2 x 2	14578	17723	11	9
1 x 1 x 1	116630	128924	31	22

Figs. 7 and 8 show five different muscle images according to different forces applied. The bottom of the muscle (blue points) is fixed, while forces are applied at the top (yellow points) in three directions. The FEM simulation is completely volumetric so that we can see changes of the muscle shape in the various cases. It is very important to both know and view the muscle geometric change imposed by the external force when we carry out surgery simulation and cutting. The images of Fig. 7 were generated using OpenGL shaded polygons with 1,840 voxels, and those of Fig. 8 were produced by ray-casting with 14,578 voxels each.

Fig. 9 presents several snapshots of our dynamic FEM simulation rendered with OpenGL shaded polygons. The dynamic model shows the dumping process of the muscle when a sudden external force is applied at the top of the muscle (yellow points) while the bottom (blue points) is fixed at the lowest points. The muscle is oscillating with decreasing frequency. The FEM simulation is carried out with 450 nodes and 219 voxels. The whole process took 0.76 seconds for 22 frames.

8 Summary and Future Work

We have presented a biomechanically-based 3D FEM muscle model which allows accurate prediction of muscle deformation changes. Our reconstructed muscle mesh is based on individual patient CT/MRI

volume data. We use a hierarchical voxel mesh which it is suitable for both FEM simulation and volume graphics. The physiological muscle force is considered and linear elastic muscle model for both static and dynamic cases are simulated by FEM. 3D wireframe, polygon surface and volume rendering techniques are applied to show real-time muscle deformation processes as well as realistic animation in the low and medium-resolution. The simulations and renderings of VHD anconeus muscle demonstrate an excellent performance of the new approach.

Our future research is directed towards improving the quality and speed of deformed volume rendering. We plan to parallelize the simulation and rendering. Another possibility is to use modal analysis for FEM dynamic system analysis. While the direct integration method is suitable for a smaller number of voxels, modal analysis is more appropriate for a larger number of voxels.

9 Acknowledgments

We would like thank Shigeru Muraki, Suya You, Ming Wan, Akio Doi and Ikuko Takanashi for their earlier work on this project. Also, thanks to Jeffrey Ge and Jack Stern for their useful suggestions for our work. Kathleen McConnell advised us greatly in writing this paper. Pat Tonra and Ken Gordon also provided much help. The project was partially supported by Mitsubishi Electric Research Lab (MERL) and Office of Naval Research (ONR), and National Science Foundation (NSF) under grant MIP-9527694..

References

1. Bro-Nielsen, M., and Cotin, S., "Real Time Volumetric Deformable Model for Surgery Simulation using FE and Condensation," *Proc. Eurographics '96* pp.C57-C66.
2. Chen, D., and Zeltzer, D. "Pump It Up: Computer Animation of a Biomechanically Based Model of Muscle Using the Finite Element Method," *Proc. Siggraph'92*, pp.89-98.
3. Fruhauf, T., "Raycasting of Nonregularly Structured Volume Data," *Proc. Eurographics '94*, pp. 293-303.
4. Gibson, S.F.F., Samosky, J., Mor, A., Fyock, C., Grimson, E., Kanade, T., Kikinis, R., Lauer, H., McKenzie, N., Nakajima, S., Ohkami, H., Osborne,

- R., Sawada A., "Simulating Arthroscopic Knee Surgery using Volumetric Object Representations, Real-Time Volume Rendering and Haptic Feedback", *MERL Tech Report TR96-19*
5. Gibson, S.F.F., "Beyond Volume Rendering: Visualization, Haptic Exploration, and Physical Modeling of Voxel-based Objects," *Eurographics Workshop on Visualization in Scientific Computing*, 1995, pp. 171-201.
6. Haley, M., and Blake, E. H., "Incremental Volume Rendering Using Hierarchical Compression," In *Proc. Eurographics '96*, pp. 45-55.
7. Hong, L., Muraki, S., Kaufman, A., Bartz, D., He, T., "Virtual Voyage: Interactive Navigation In The Human Colon," *Proc. Siggraph '97*, pp. 27-34.
8. Keeve, E., Girod, S., Pfeifle, P., Girod, B., "Anatomy-Based Facial Tissue Modeling Using the Finite Element Method," *Proc. IEEE Visualization '96*, pp. 21-28.
9. Koch, R., Gross, M., Carls, F., Buren, D., Fankhauser, G. and Parish, Y. "Simulating Facial Surgery Using Finite Element Models", *Proc. Siggraph '96*, pp. 421-428.
10. Mao, X., Hong, L. and Kaufman, A., "Splatting of Curvilinear Volumes", *Proc. IEEE Visualization '95*, pp. 61-68.
11. National Library of Medicine, "The Visible Human Project", with URL: http://www.nlm.nih.gov/research/visible/visible_human.html, 1997.
12. Pieper, S., Rosen, J. and Zeltzer, D., "Interactive Graphics for plastic Surgery: A Task-level Analysis and Implementation," 1992 *Symposium on Interactive 3D Graphics*, pp. 127-134.
13. Scheepers, F., Parent, R., Carlson, W. and May, S., "Anatomy-Based Modeling of the Human Musculature" *Proc. Siggraph '97*, pp. 163-172.
14. Stern, J., "Computer Modeling of Gross Muscle Dynamics," *Journal of Biomechanics*, 1974, Vol. 7, pp. 411-428.
15. Tabatabai, B., Sessarego, E. and Mayer H., "Volume Rendering on Non-regular Grids," *Proc. Eurographics '94*, pp. C247-C258.
16. Terzopoulos, D., Platt, J., Barr, A. and Fleischer, K., "Elastically Deformable Models," *Proc. Siggraph '87*, pp. 205-214.
17. Wilhelms, J. and Van Gelder, A., "Anatomically Based Modeling," *Proc. Siggraph '97*, pp. 173-180.

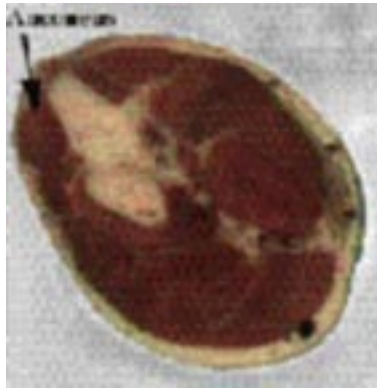


Figure 3: *Cross-section of the arm from the Visible Human Data which includes the anconeus muscle.*

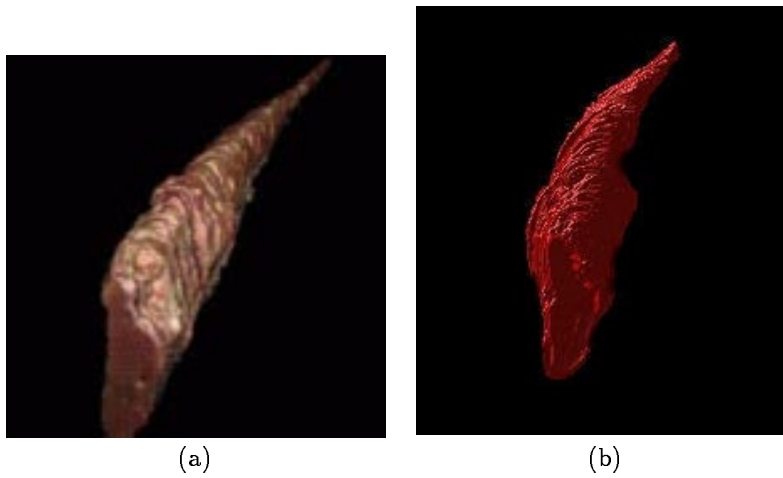


Figure 4: *Anconeus muscle volume dataset: (a) Ray casted image. (b) Surface rendering of manually segmented anconeus.*

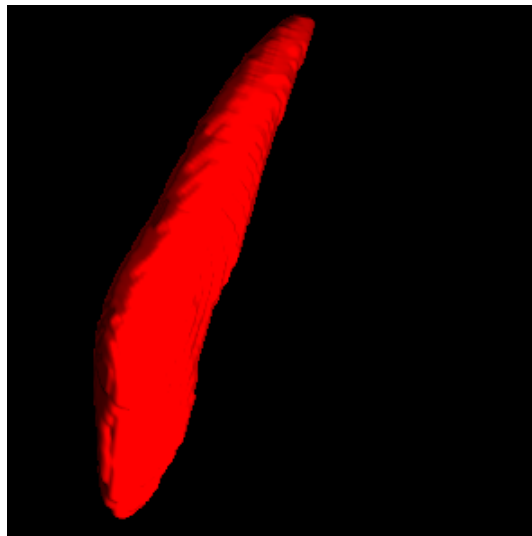


Figure 5: *Filtered image from voxel meshes of 14578 voxels*

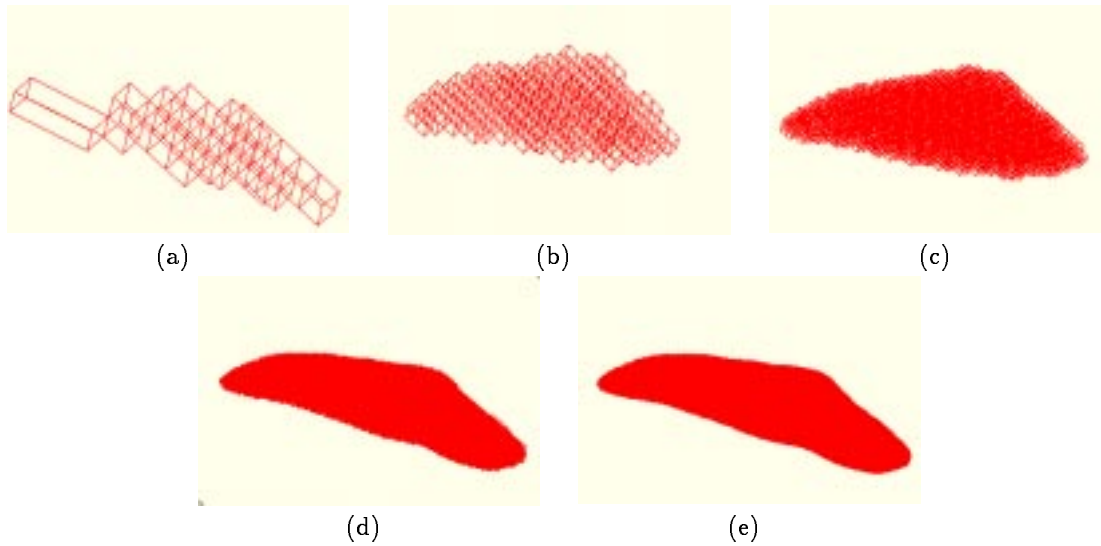


Figure 6: *Wireframe model for binary hierarchical muscle voxel meshes of different voxel sizes: (a) $16 \times 16 \times 16$. (b) $8 \times 8 \times 8$. (c) $4 \times 4 \times 4$. (d) $2 \times 2 \times 2$. (e) $1 \times 1 \times 1$.*

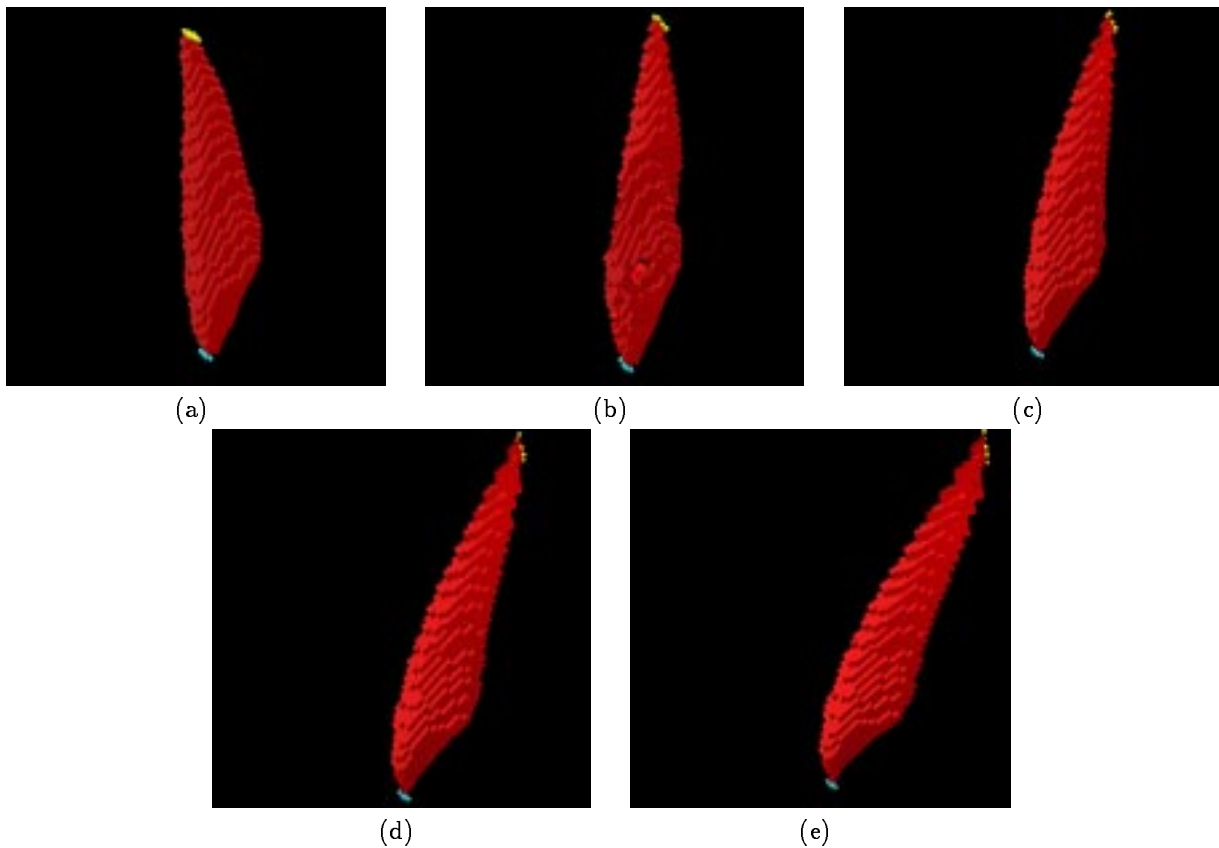


Figure 7: *Medium resolution of static muscle model: (a) with no Newtons of force applied; (b) with 250 Newtons of force applied; (c) with 500 Newtons of force applied; (d) with 750 Newtons of force applied; (e) with 1000 Newtons of force applied.*

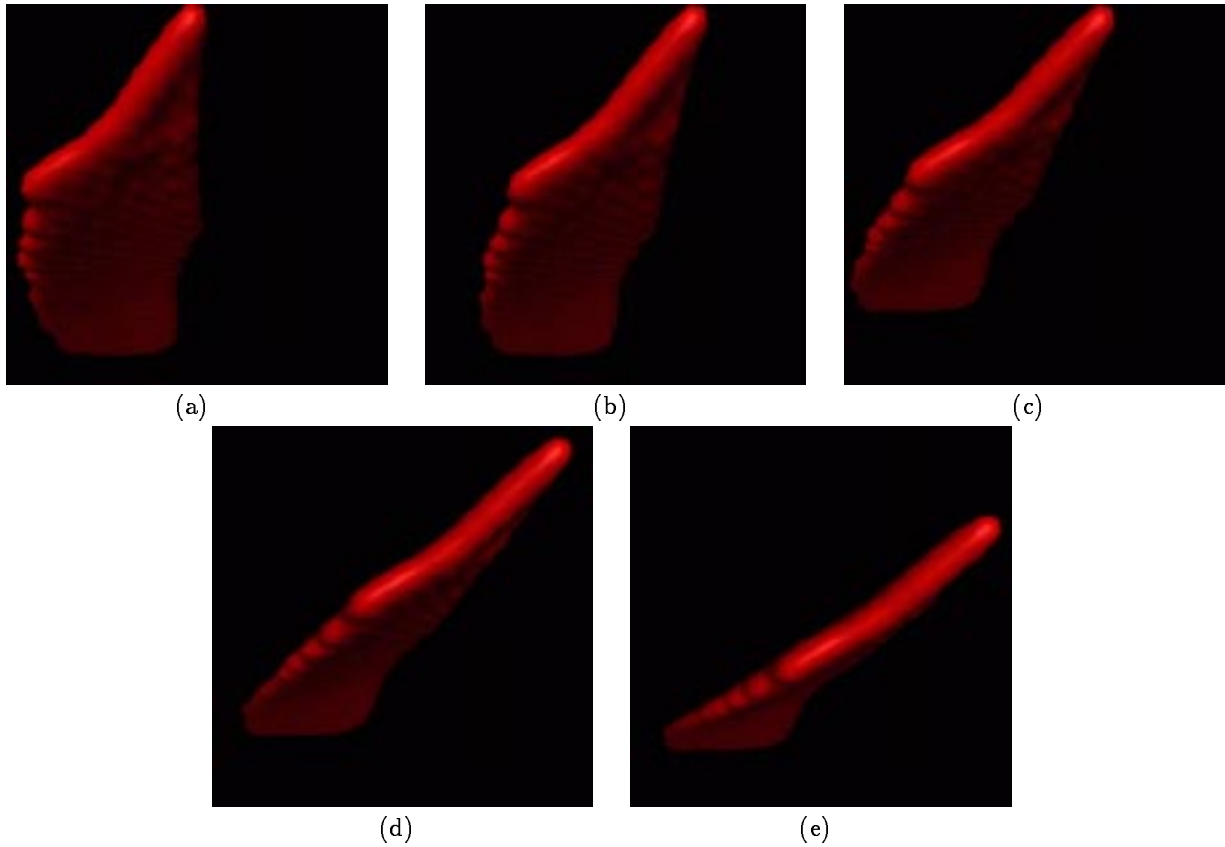


Figure 8: *High resolution of static muscle model: (a) with no Newtons of force applied; (b) with 250 Newtons of force applied; (c) with 500 Newtons of force applied; (d) with 750 Newtons of force applied; (e) with 1000 Newtons of force applied.*

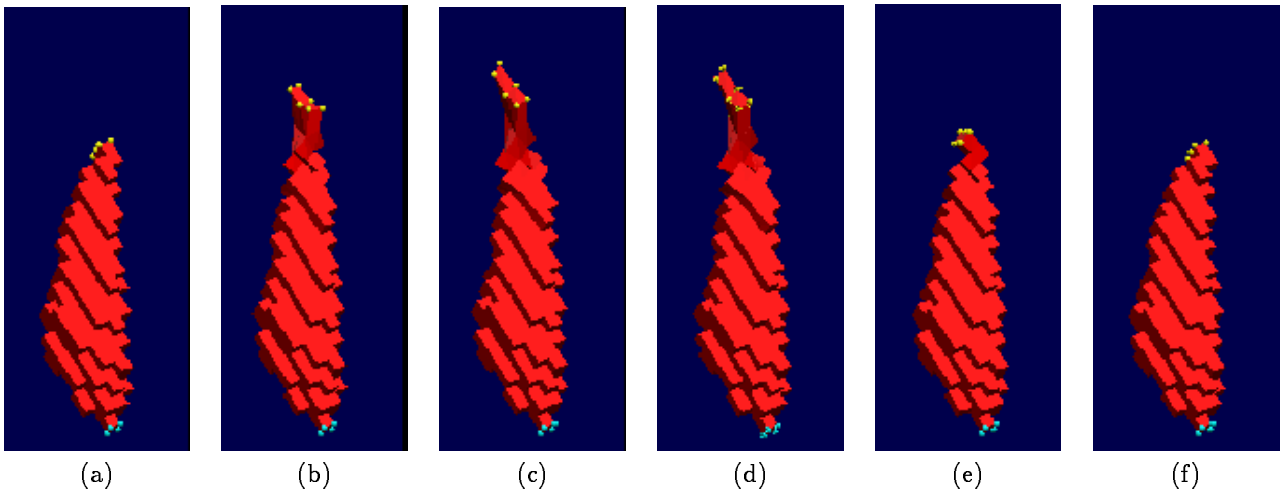


Figure 9: *Medium resolution of dynamic muscle model: (a) - (f) from start to end*