Interactive Modeling of the Human Musculature

Amaury Aubel and Daniel Thalmann
Computer Graphics Lab (LIG), Swiss Federal Institute of Technology (EPFL)
E-mail: {Amaury.Aubel,Daniel.Thalmann}@epfl.ch

Abstract

In this paper we extend previous work [1] and propose a muscle model suitable for computer graphics based on physiological and anatomical considerations. Muscle motion and deformation is automatically derived from one or several action lines, each action line being deformed by a 1D mass-spring system. The resulting model is fast, can accommodate most superficial human muscles, and could easily be integrated into current modeling packages.

Example animations can be found at http://ligwww.epfl.ch/~aubel/MuscleBuilder/

1. Introduction

The basic function of the skeletal muscles is to generate movement. Upon contraction, the fibers that make up the muscle contract and slide across each other. As a result, the length of the whole muscle diminishes, so the bones to which the muscle is attached are pulled towards each other. A side effect is that the muscle changes shape during contraction, which impacts the shape of the outer skin. This is well-known among painters and sculptors who study the anatomy of the human body to improve their work. Rather surprisingly, commercial modeling packages overwhelmingly overlook muscle modeling as an essential part of body modeling. The most widespread technique for skin deformation in the industry remains *skinning* which amounts to binding each skin vertex to one or more underlying bones. The displacement of a skin vertex during animation is then the result of a weighted combination of the displacements of the bones to which it is bound. The influence of muscles on the skin surface shape is usually approximated by simplistic geometric primitives like spheres that push the skin outwards.

Our work aims at providing designers with an interactive tool for automatically deforming a muscle layer in an anatomically correct way. We do not tackle the issue of modeling each muscle in a rest posture but rather concentrate on modeling the deformation of each muscle in any posture. Different research teams have fortunately put a lot of effort into semi-automatically reconstructing individual mucles from medical data [3, 10] and their work could be used as an input to ours. The muscle model we present in this paper is fast and yet realistic enough for a computer graphics use. A typical application would consist of using this model for guiding the elastic deformations of an overlying skin. However, it could also have applications in other fields such as medicine, sports, and education.

The remainder of this paper is organized as follows. Section 2 recalls the existing muscle models. Section 3 exposes some important anatomical considerations and highlights how they can be transposed to a computer graphics model. Extending the work we have presented in [1] we detail the muscle model in Section 4. Section 5 describes in detail the interactive tool and shows a concrete example of how a deformation model is designed. Current results are presented in Section 6. Finally, Section 7 gives concluding remarks and discusses present and possible future work.

2. Related work

Existing muscle models can broadly be classified into two categories: purely geometric models and physically-based ones. We successively review these two approaches.

2.1. Geometric deformations

Geometric models tend to use the ellipsoid as the basic building block. It is a natural choice because an ellipsoid approximates fairly well the appearance of a fusiform muscle. In addition, its analytic formulation lends itself well to inside/outside tests and volume preservation constraints. Thus, several researchers use a volume-preserving ellipsoid for representing a fusiform muscle [14, 19]. Others approximate muscles by an implicit surface extracted from a set of ellipsoids [18, 17]. Finally, one research team represents multi-belly muscles such as the *pectoralis* by a set of ellipsoids positioned along two spline curves [14]. In all these works, muscle flexing and bulging is simulated by binding the degrees of freedom (scaling and possibly translation

0-7803-7237-9/01/$10.00 ©2001 IEEE

and/or rotation) of each ellipsoid to the degrees of freedom of the underlying skeleton joints.

Despite its simplicity and attractiveness, the ellipsoid model cannot capture most muscle shapes. In a more recent work [20], Whilhelms et al. use a generalized cylinder made up of a certain number of cross-sections that consist in turn of a fixed number of vertices. Volume variation of the muscle during deformation is reduced by scaling each cross-section so as to preserve its area. Similarly, Scheepers and his colleagues provide a general muscle model that consists of tubularly-shaped bicubic patches [14]. Exact volume preservation remains possible as muscles shapes still have an analytic description. Interestingly, they also provide the user with scaling and tension parameters to simulate isometric contractions as well.

2.2. Simulation models

One of the first physically-based models is due to Chadwick et al. [4] who embed every muscle in a FFD lattice [15]. Muscle deformation is achieved by simply deforming the embedding space. The FFD control points are moved by treating them as nodes interconnected by ideal hookean springs. Diagonal springs help to maintain the initial geometric configuration. One of the potential problems is that the FFD box may not approximate the muscle shape very tightly. The FFD control points have moreover no physical reality. As a consequence, the distribution of the muscle mass over the nodes is likely to be problematic.

From a more bio-mechanics oriented point of view, Chen et al. simulate muscle contraction using the Finite Element Theory [5]. Their work only shows single muscles working in isolation.

Porcher-Nedel and Thalmann introduced the idea of abstracting muscles by an action line (a polyline in practice) representing the force produced by the muscle on the bones, and a surface mesh deformed by an equivalent mass-spring network [12]. An elastic relaxation of the surface mesh is performed for each animation frame thus yielding a collection of static postures. In order to smooth out mesh discontinuities, they employ special springs termed angular springs that tend to restore the initial curvature of the surface at each vertex. However, angular springs cannot deal with local inversions of the curvature. Also, the authors do not explicit how they constrain the surface mesh to follow the action line when it consists of more than one segment.

Ng-Thow-Hing relies on the B-spline solid as the basic primitive for modeling individual muscles in animals and humans [10]. The mathematical formulation of the B-spline solid can accommodate multiple shapes of muscles (fusiform, triangular, bipennate, etc.) and various sizes of attachments. Muscular deformation is achieved by embedding a mass-spring-damper network in the B-spline solid. In practice, the network does not coincide with the B-spline's control points but with spatial points of maximum influence since physical characteristics such as mass are best specified at real locations of the muscle. Varying force magnitude in the network results in non-uniform physical effects. In contrast to most previous approaches that solve a sequence of static equilibrium problems only, inertially-induced oscillations can take place here thus enhancing the visual realism. Muscle-muscle and muscle-bone collision forces are also added as reaction constraints [11]. Yet, trying to simulate every muscle-muscle and muscle-bone interaction seems unrealistic. For example, no solution is given as to how multiple collisions between muscles are to be handled.

3. Anatomical considerations

The muscle layer is the main contributing factor to the surface form. Indeed, muscles account for nearly half of the total mass of the body and fill in almost completely the gap between the skeleton and the skin [13]. Although three types of muscles exist, we shall only consider the skeletal muscles because the other kinds hardly influence the surface form.

Skeletal muscles produce the motion of the bones. Structurally, they consist of a contractile central part named *belly* and, most often, of tendinous extremities that connect the belly to the bones. The attachement to the more stationary bone is called the *origin* while the other end is called the *insertion*. In constitutive description, the belly consists of bundles of elastic contractile fibres. The bundles are in turn wrapped into a single envelope called *fascia*. The belly's fibres are responsible for producing the contraction of the whole muscle. Tendons, which are hardly elastic on the other hand, act as transmitters. In general, muscle tissues are several orders of magnitude more elastic than tendons [6].

The approximately 700 skeletal muscles in the human body differ greatly in size: Some skeletal muscles are very large, such as the *gastrocnemius*, the major muscle forming the calf in the lower leg. Others are very small, such as the muscles of the eyelid. Muscle shapes vary also greatly: Some muscles are triangular (e.g. *deltoid,* see Figure 3 for muscles of the upper arm), while others are rectangular (e.g. *rectus abdominis*) or trapezoidal (e.g. *trapezius*). In general, long fusiform muscles (e.g. *quadriceps)* are found mainly in the limbs while short muscles appear around joints (e.g. *brachialis*) and large, flat muscles cover the back (e.g. *latissimus dorsi*). Finally, the biological diversity is also great concerning the number of bellies and tendons: Some muscles are bicipital (e.g. *biceps brachii*); Some muscles in the leg have two sucessive bellies separated by a median tendon; Sometimes, tendons are even missing and the belly may attach directly to the bone. Owing to this great diversity, overly simple models such as the ellipsoid are unsuitable for muscle modeling. Only general triangle meshes and B-spline models avoid any loss of generality.

All muscle tissues exhibit two fundamental properties: i) They can contract and shorten in length ii) After contraction, they relax and return to their former length. Anatomists distinguish between two types of contraction: *Isometric* (same length) and *isotonic* (same force) contraction. Upon isotonic contraction, the volume of the belly increases thus amplifying its influence on the shape of the skin, whereas the total length of the muscle diminishes so that the bones to which the muscle is attached are pulled towards each other. Upon isometric contraction, the shape of the belly alters but the length of the muscle does not change, so no skeletal motion is produced. The performance of most exercices typically involves a combination of isotonic and isometric contractions [2].

Finally, multi-belly muscles usually have clear bundles e.g. the pectoral has four main bands of fibers. However, as these bundles are wrapped into connective tissues (i.e. *fascia*), the muscle displays a single, continous, smooth surface. In a computer graphics model, it therefore makes sense to dissociate the main directions of contraction of the muscle from its surface.

4. Muscle model

Research so far has mainly focussed on modeling muscles in static postures and largely ignored the animation problem. And yet, it is very complex to automatically derive the appropriate position and deformation of a muscle in any possible posture. In our system, as often in computer graphics, the motion of the skeleton induces the muscular deformations contrary to what occurs in reality.

We decompose the muscle in two layers: a skeleton and a surface mesh. The skeleton is simply defined by the action lines of the muscle (an action line, as meant by the biomechanics community, denotes the imaginray line straight or not, along which the force exterted onto the bone is produced). The deformations of the muscle shape are entirely driven by the underlying action lines which implies that the 3D nature of the elasticity problem is reduced to 1D for fusiform muscles (one action line) and 2D for flat muscles (several action lines).

In order to avoid possible confusion in the following, we shall use the term *node* when referring to a vertex of an action line and *vertex* when speaking of a vertex of the muscle surface mesh.

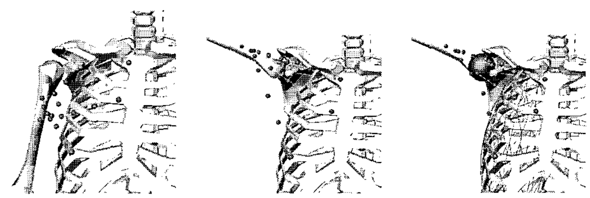

Figure 1. Actions lines of the pectoral muscle during shoulder abduction. Right figure shows the use of two attractive force fields (solid and wireframe ellipsoids)

4.1. Action line

Each action line is approximated by a polyline from which a 1D mass-spring-damper system is constructed. The masses that correspond to the muscle insertion and origin (usually the first and last nodes) are anchored to the skeleton joints so that their motion is driven only by the skeleton later on. The positions of the in-between nodes are obtained at each animation frame through an elastic relaxation of the mass-spring system. Currently, all nodes are assigned an equal mass, which is not a problem because the mass-spring system has no physical reality but is rather used as a deformation tool. The stiffness of each spring is determined by the material type (i.e. tendon or belly) it represents. We also add attractive and repulsive force fields (a combination of ellipsoids currently) to constrain the action line. Repulsive force fields prevent gross interpenetration while attractive fields help to refine the trajectories of the action line.

4.2. Local frames

The positions of the action line nodes provide information as to how the surface mesh will expand or shrink over time. So as to infer the orientation of the mesh, we need to augment each action line node with a local frame. The construction of these local frames is described next.

We start by computing the Z-axis at each node as depicted in Figure 2: Z is set to the normal of the bisecting plane for every in-between node (V1, V2) and to the tangent for the end nodes (V0, V3). We then proceed to compute the X-axes. The Y-axes are ultimately found by completing the right-handed coordinate systems.

The X-axis is first computed for the origin and insertion nodes. We take, in the rest posture, the local frame of the joint (X and Y solid arrows in Figure 2) to which the node is bound and rotate it so as to bring one of its axes in alignment with the node's Z-axis. The selected axis is the one that leads to the minimal rotation (X in Figure 2 for instance).

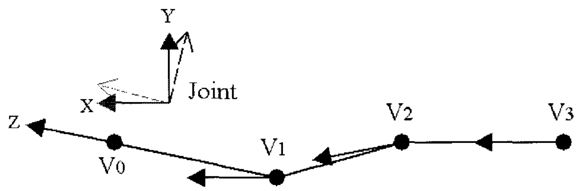

Figure 2. Z axis is set using the bisecting plane. The joint frame is initially rotated so as to align one of its axis (here the X axis) with axis Z of end node V0.

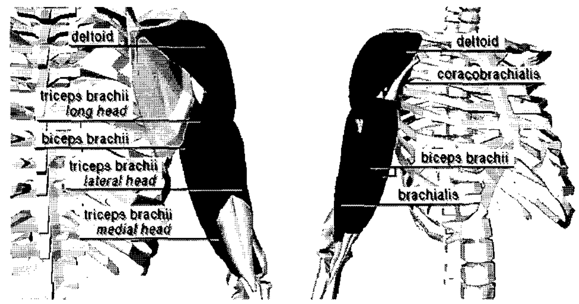

Figure 3. Anterior and front-side view of the musculature of the upper arm.

The resulting rotated frame (dashed arrows in Figure 2) is expressed and saved in the joint's coordinate system. During any subsequent animation, this frame is transformed by the joint's current coordinate system, then rotated again so as to be aligned with the node's new Z-axis. This rotation is usually quite small because the smallest rotation in the rest posture was initially chosen. Thus, the local frame of every non-dynamic node is smoothly updated as the action line moves and deforms itself.

There remains to compute the frames for the dynamic nodes in between. It is no use interpolating the two end orientations directly using the commonplace spherical linear interpolation because the two end orientations may be quite different from each other. As spherical linear interpolation picks the shortest path on the quaternion unit sphere, the frame orientation at a node may flip from one animation frame to the next. Besides, direct interpolation does not guarantee the interpolated frames will be aligned with the Z-axes already computed. Our method consists in propagating the X-axis direction of each end frame to the in-between nodes. As Z-axes are readily computed using the method described above, we already have, for each node Vi, a plane Pi normal to Zi in which the remaining Xi and Yi axis must lie. Starting from axis X0 at end node V0, we estimate the axis X1 in the plane P1 by sampling the trigonometric circle and finding the minimal deviation from X0. As we sample the entire circle, we do not get stuck into local minima. The process is iterated for each Xi+1 by minimizing the deviation from Xi. Thus, we propagate the orientation of node V0 to the other end node Vn and conversely from Vn to V0. Finally, a linear interpolation of the two X-axes computed at each node is performed using a ratio that is related to the distance from the in-between node to the two end nodes along the polyline.

4.3. Muscle mesh

We extend the parametrization of a mesh vertex by an underlying action line introduced in our previous work [1] in order to account for muscles with several action lines. Thus, each vertex of the muscle mesh is now parametrized by one or two underlying action lines. The computation of the mesh parametrization is done in two passes. First, we map every vertex to every action line using the XY planes of the local frames as space delimiters [16]. The set of action lines is then ordered by distance to each vertex. In the second pass, we construct the region of influence of each action line based on a distance criterion. The region of influence takes on the form of a single subset of connected vertices. This pass ensures that a vertex is parametrized by the two closest action lines.

A vertex is finally parametrized by coordinates (u1, s1, t1) and (u2,s2,t2) where u indexes the action line, s is the index of the segment, and t expresses a ratio along segment s. Distances to action line u1 and u2 are also computed, normalized, and saved as weights. During animation, a vertex is then moved using simple bilinear interpolation.

5. Interactive tool

We have developed an interactive graphical tool for easily deforming muscle meshes. In this tool, meshes representing the bones and/or muscles can be loaded and placed interactively on top of the wireframe skeleton (Figure 3). We describe in the following sections the various stages for creating a complete deformation model for a given muscle.

5.1. Semi-automatic creation of an action line

For designing an action line, the user starts by specifying the muscle origin and insertion and the numbers of belly and tendinous segments. A straight action line with the right number of nodes is then automatically created. Segment stiffnesses are automatically assigned, the stiffness of a tendon segment being set to an order of magnitude higher than a belly segment. Tendons are in fact several orders of magnitude stiffer than muscle tissues (Section 3.1). We found out, however, that extremely stiff tendons increased the system instability too much. In practice, increasing the stiffness tenfold suffices from a graphical point of view. The position of each node along the straight line joining the two end nodes can then be modified interactively. An inverse dynamics procedure computes rest lengths for each spring so that current lengths are much higher than rest lengths and that the equilibrium is maintained for the chosen nodes' positions. Having small rest lengths ensures that springs remain always elongated. It is important because a series of springs experiencing compression creates uncontrollable, unwanted jags along the action line. Besides, human muscles *in vivo* are never in a fully relaxed state [13]. Elongated springs therefore represent the natural tension of the muscle.

5.2. Bending the action line

The designer adds ellipsoidal force fields to bend the (so far) straight action line into the required shape. For a fusiform muscle, this is more or less the centroid curve. By combining several ellipsoids, more complex force field shapes can be defined. In order to avoid force fields generating forces that would destroy the nice balance produced by the carefully chosen springs' stiffnesses, we introduce two modes for force fields.

In *radial* mode, the node is attracted towards the center of the closest ellipsoid. This is useful for ellipsoids that are close to a spherical shape. In *orthogonal* mode, the node is attracted towards its projection on the closest ellipsoid [8]. In this mode, the node may slide freely on the force field surface until reaching a minimal energy configuration (Figure 4). This helps to prevent the force field from counteracting the forces produced by the springs.

Figure 4 shows the concrete use of four force fields to bend the action line of the triceps into the desired shape. The pulley at the elbow is modeled by a repulsive force field (spherical shape in the picture plane) in conjunction with an elongated attractive force field (lower left). The (weakly) attractive force fields constrains the nodes to stay on the lower part of the elbow. The two remaining force fields are even thinner. As the orthogonal attraction mode is used, the action line nodes freely slide over their surfaces under the influence of the adjacent springs. The resulting deformation

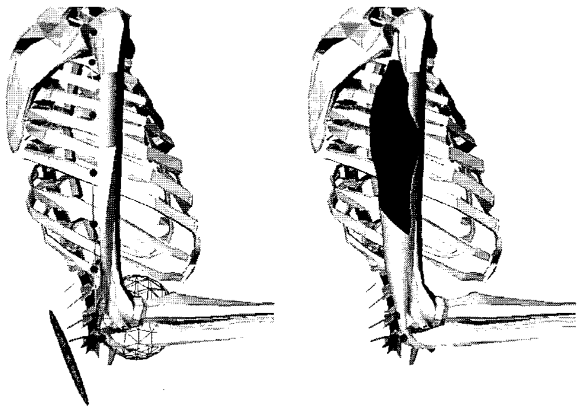

Figure 4. Use of force fields and resulting mesh deformation.

of the mesh is displayed in the right picture. Notice how the tendon bends at the elbow in a natural fashion thus pulling the muscle's belly donwards. The mesh is automatically scaled to account for the change in length of the action line.

5.3. Avoiding collisions

Attempting to handle all muscle-muscle and muscle-bone collisions is unreasonable. In our framework, preventing muscle-bone collisions is easy and fast using repulsive force fields (Figure 4). As for muscle-muscle interaction, we allow any force field to be anchored to and deformed by an action line. Thus repulsive ellipsoidal force fields can be positioned along the action line(s) of a deep muscle and their action immediately exerted upon more superficial muscles. This requires that the relaxation of the action lines be performed in a certain order. The resulting limitation is that if muscle A influences muscle B, muscle B cannot in turn influence muscle A. However, muscle interpenetration is naturally reduced by carefully desiging the deformation of each individual muscle.

5.4. Controlling the mesh deformation

The three previous stages are iterated for each belly of the muscle. The process can be accelerated by reusing certain components (insertion and origin nodes, force fields, etc.) between several action lines. Afterwards, the muscle is anchored to the action lines in the rest posture. There remains to specify for each action line node the scaling constants upon isometric contraction. This specifies how the mesh around the node should be scaled along the X and Y directions of the local frame.

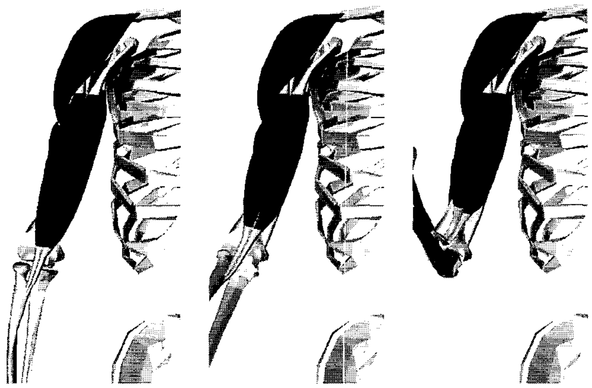

Figure 5. Deformation of the biceps upon elbow flexion (0, 60 and 120 degrees).

During animation, the scaling of each mesh vertex results from the combination of the isotonic, passive contraction and of the isometric, active contraction of the action lines. As described before, the isotonic contraction is automatically computed from the length variation of the action line. Isometric contraction is specified by the designer as an activation curve for any given motion. Automatic determination of the activation curve based on the motion performed is left as future work.

6. Results

Short animation clips can be found on the following website http://ligwww.epfl.ch/~aubel/MuscleBuilder/.

6.1. Upper arm

We have created deformation models for all major muscles of the upper arm. The deformation of the action lines may take up to several seconds for a given posture. However, in an animation sequence, the relaxation of each action line performs much faster as very few iterations are needed to reach equilibrium due to the temporal coherence. On an Octane workstation with a R10000 processor, deformation of all muscles of the upper arm (*deltoid* included) takes one second per frame on average.

6.2. Pectoral muscle

Figure 6 shows on-going modeling work for the pectoralis. The last part of the muscle is still missing as can be seen on the first picture where one action line is not covered by a mesh. The fact that holes appear between the meshes is not as worrisome as may seem since these holes do exist in women or emaciated people (our meshes are very thin, which would correspond to a woman's pectoral). The skeletal motion combines a shoulder flexion and an abduction. Notice how the meshes slide over the rib cage and get thinner as the arm is stretched back.

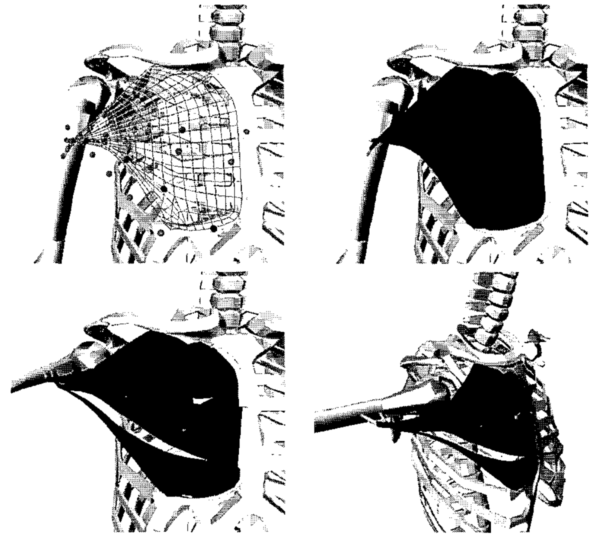

Figure 6. Deformation of the pectoralis.

7. Conclusions and future work

We presented a two-layered muscle model suitable for computer graphics applications. A set of action lines is used for driving the motion and deformation of the outer layer. As the model makes use mostly of 1D mass-spring systems, muscle deformation is performed quickly. Our approach to muscle modeling requires some knowledge of anatomy but there exists an exhaustive literature on the subject.

The deformation of each individual muscle is obtained through a relaxation of its action lines. We thus have, for any animation, a collection of static postures i.e. we simply have a kinematic control over the deformations. We are presently working on adding dynamics effects to the action lines, especially inertia-induced oscillations and gravitational effects. Our idea is to treat the deformation of an action line as the combination of a rigid kinematic motion (main motion) and dynamic oscillations (secondary motion) generated by a mass-spring system about the reference positions of the main motion. More specifically, the oscillations are to be generated by anchoring each node of the action line to its kinematic trajectory by a spring whose stiffness would be related to the quantity of matter in the node's neighbourhood.

We are also considering covering the muscles with a deformable skin mesh. We deform the skin mesh using a two-stage process. A skinning process is first used for roughly positioning each skin vertex. We decouple the degrees of freedom of a 3-DOF joint (e.g. the shoulder) into a swing motion on the one hand and a twist motion on the other [7]. This should prove useful because the skin does not completely follow the skeleton when twisting one's limb [9]. In the second stage the skin vertices are pushed and attracted by the underlying muscles using a ray casting scheme that maintains a fixed initial distance from each vertex to the muscles and bones.

Acknowledgments

This work is partly funded by the Federal Office for Education and Science in the framework of the European project MESH.

References

[1] A. Aubel, D. Thalmann. Realistic Deformation of Human Body Shapes. *Proc. Computer Animation and Simulation 2000 (11th Eurographics Workshop)*, Interlaken, pp. 125-135 (August 2000).

[2] C.L. Benjamin, G.R. Garman, J.H. Funston. *Human Biology*. Mac-Graw Hill, 1997.

[3] P. Beylot, P. Gingins, P. Kalra, N. Thalmann, W. Maurel, D. Thalmann, J. Fasel. 3D Interactive topological modeling using Visible Human Dataset. *Computer Graphics Forum (Eurographics '96 Proc.)*, **15**(3):33-45, 1996.

[4] J.E. Chadwick, D.R. Haumann, R.E. Parent. Layered construction for deformable animated characters. *ACM Computer Graphics (Proc. of SIGGRAPH '89)*, **23**:243-252, 1989.

[5] D. Chen, D. Zeltzer. Pump it up: Computer animation of a biomechanically based model of muscle using the finite element method. *ACM Computer Graphics (Proc. of SIGGRAPH '92)*, **26**:89-98.

[6] Y.C. Fung. *Biomechanics: Mechanical Properties of Living Tissues*. Springer-Verlag, 1981.

[7] S. Grassia. Practical Parametrization of Rotations Using the Exponential Map. *Journal of Graphics Tools*, **3**(3):29-48, 1998.

[8] J.C. Hart. Distance to an Ellipsoid. *Computer Graphics Gems*, Academic Press, 1994, pp. 113-120.

[9] T. Molet, R. Boulic, D.Thalmann. A Real-Time Anatomical Converter for Human Motion Capture. *Proc. Computer Animation and Simulation 1996 (7th Eurographics Workshop)*, Wien, (September 1996).

[10] V. Ng-Thow-Hing. *Anatomically-Based Models for Physical and Geometric Reconstruction of Humans and Other Animals*. Ph.D. Thesis, Department of Computer Science, University of Toronto, 2000.

[11] J.Platt, A. Barr. Constraint Methods for Flexible Models. *ACM Computer Graphics (Proc. of SIGGRAPH '88)*, **22**:279-288.

[12] L. Porcher-Nedel, D.Thalmann. Real Time Muscle Deformations Using Mass-Spring Systems. *Proc. CGI '98*, IEEE Computer Society Press, 1998.

[13] P. Richer. *Artistic Anatomy*. Watson-Gutpill Publications, 1981, Translated by Robert Beverly Hale.

[14] F. Scheepers, R. Parent, W. Carlson, S. May. Anatomy-Based Modeling of the Human Musculature. *ACM Computer Graphics (Proc. of SIGGRAPH '97)*, 163-172, 1997.

[15] T. Sederberg, S. Parry. Free-From Deformation of Solid Geometric Models. *ACM Computer Graphics (SIGGRAPH '86 Proceedings)*, **20**:151-160, 1986.

[16] W. Sun, A. Hilton, R. Smith, J. Illingworth. Layered Animation of Captured Data. *Animation and Simulation '99 (10th Eurographics Workshop Proceedings)*, pp.145-154 (1999).

[17] D. Thalmann, J.Shen, E. Chauvineau. Fast Realistic Human Body Deformations for Animation and VR Applications. *Proc. of Computer Graphics International '96*, Pohang, Korea, June, 1996.

[18] R. Turner, D. Thalmann. The Elastic Surface Layer Model for Animated Character Construction. *Proc. of Computer Graphics International '93*, Lausanne, Switzerland, Springer-Verlag, Tokyo, pp. 399-412 (1993).

[19] J. Wilhelms. Animals with Anatomy. *IEEE Computer Graphics And Applications*, **17**(3):22-30, May 1997.

[20] J. Wilhelms, A. Van Gelder. Anatomically Based Modeling. *ACM Computer Graphics (Proc. of SIGGRAPH '97)*, 173-180, 1997.