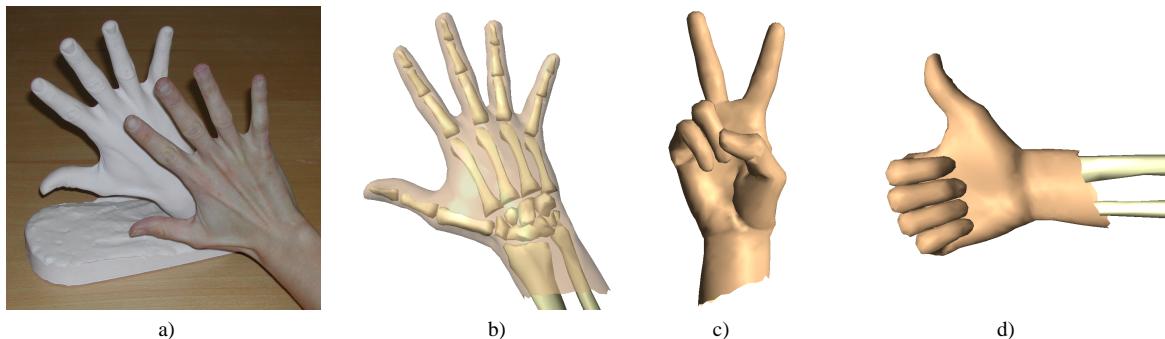


# **Construction and Animation of Anatomically Based Human Hand Models**

Irene Albrecht, Jörg Haber, and Hans-Peter Seidel

MPI Informatik, Saarbrücken, Germany



Construction and animation of the reference hand model: a) plaster cast of a human hand ready for 3D scanning; b) assembly of skin mesh and individual bone meshes; c) and d) skin tissue deformation during animation.

### **Abstract**

The human hand is a masterpiece of mechanical complexity, able to perform fine motor manipulations and powerful work alike. Designing an animatable human hand model that features the abilities of the archetype created by Nature requires a great deal of anatomical detail to be modeled. In this paper, we present a human hand model with underlying anatomical structure. Animation of the hand model is controlled by muscle contraction values. We employ a physically based hybrid muscle model to convert these contraction values into movement of skin and bones. Pseudo muscles directly control the rotation of bones based on anatomical data and mechanical laws, while geometric muscles deform the skin tissue using a mass-spring system. Thus, resulting animations automatically exhibit anatomically and physically correct finger movements and skin deformations. In addition, we present a deformation technique to create individual hand models from photographs. A radial basis warping function is set up from the correspondence of feature points and applied to the complete structure of the reference hand model, making the deformed hand model instantly animatable.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling; *hierarchy and geometric transformations, physically based modeling*; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism; *animation*.

## 1. Introduction

Our hands play a vital role in every aspect of our daily lives. We need them for eating, playing, writing, working, communicating, in a nutshell: for everything. Most people take the effectiveness and dexterity of their hands for granted without being aware of their complicated structure and the high level

of optimization. However, there is more than the mechanical perfection to our hands:

*Often the hands will solve a mystery that the intellect has struggled with in vain.*

CARL GUSTAV JUNG

Stretching from spiritual significance (e.g. blessing, palm reading), over idiomatic expressions (e.g. “to put one’s life in someone’s hands”), to the act of shaking hands, not only for greeting but also for expressing feelings like gratefulness or sympathy, the central importance of hands is mirrored in a broad spectrum of symbolism.

In spite of the ubiquity of hands in daily life, but probably due to their immense complexity, hands have not received much attention in computer graphics. Although the number of possible applications is large, sophisticated hand models have not yet been developed. Virtual hand models can be used for teaching and practicing sign language, and for visualizing translations from speech or text into sign language. They come in handy for teaching other manual skills as well, for instance operating machines, and for giving online usage or assembly instructions. In immersive environments, hand models are required in the simulation of the haptic dimension: for manipulating a virtual object, visual feedback is helpful. Close-ups in CG movies and games ask for natural models with a lot of detail and convincing movements. High demands arise also from the medical field. In systems for hand surgery planning, a maximum of functionality of the hand must be provided to aid the surgeon in his decisions.

### 1.1. Contributions

In this paper, we present the following main contributions:

- a human hand model with anatomical structure, suitable for real-time animation using physics-based simulation of muscles and elastic skin properties (Section 3);
- a hybrid muscle model that comprises pseudo muscles and geometric muscles. Pseudo muscles directly control the rotation of bones based on anatomical data and mechanical laws, while geometric muscles deform the skin tissue using a mass-spring system (Section 4);
- a deformation technique based on feature points to warp the complete structure of a reference hand model to an individual hand model taken from a photograph (Section 5).

Our motivation to choose a physics-based approach controlled through muscle contraction values is given by the main advantage of such an approach: animations are anatomically and physically correct by default. The user does not have to take care of anatomical or physical limitations when positioning the fingers or setting up dynamics of an animation.

## 2. Related Work

### 2.1. Anatomy, Biomechanics, and Anthropometry

Research in anatomy and biomechanics has shown that the human hand is a very intricate and elegant mechanical device, where many dedicated parts cooperate in an highly optimized interplay to form a powerful union. Information on

the anatomical building blocks of the hand can be found in illustrated anatomy books<sup>35</sup> or in more detail in<sup>11</sup>. The book by Brand and Hollister<sup>8</sup> is inclined more towards biomechanics: meant as a textbook for hand surgeons, for instance when planning a tendon transfer operation, it provides a thorough description of the functioning of the hand.

Landsmeer<sup>24</sup> developed a physics-based model for determining tendon excursion from joint angle, depending on the way the tendon crosses the joint. Starting from this model, he develops criteria of how muscles must be arranged in a joint system to be able to move the joints in any given way. In<sup>2</sup>, tendon excursions of the index finger muscles have been measured and the corresponding moment arms have been computed using Landsmeer’s tendon models.

A kinematic model for flexion and extension of the fingers has been developed by Lee and Kroemer<sup>25</sup>. Their model is based on the assumption that the moment arms of the tendons at the joints are constant. Considering external forces affecting the joints, they compute the finger strength for the given joint configuration.

In<sup>4</sup>, the authors discuss a biomechanical model of the entire hand encompassing all principal muscles and degrees of freedom. Muscles are modeled by weightless expandable threads. Weightless non-expandable loops surrounding the joints describe the “line-of-action” of muscles. The authors found only the muscles at the wrist to possess some redundancy, i.e. the same wrist position can be obtained by several muscle combinations. To overcome this redundancy, muscle effort is minimized.

For evaluation of the prehensile capabilities of the human hand, Buchholz and Armstrong<sup>9</sup> proposed a kinematic model based on collision detection between ellipsoids representing the skin surface of the hand segments. Joint flexion angles and skin deformation for power grasp of ellipsoidal objects are predicted and rendered as vector graphics.

The anatomical computer-generated hand model described in<sup>39</sup> consists of bones, tendons, and soft tissue. The latter is modeled by an ellipsoid-shaped mass-spring network at every phalanx, and as an appropriately shaped mass-spring system at the palm. The outer surface of these networks constitutes the skin. Tissue deformation during finger movement is determined using a predictor-corrector method, which also takes into account incompressibility and collision constraints. Tendons are present via their mechanical effects, not geometrically. Their feedback action is modeled through springs opposing joint motion. The fingers are positioned automatically by energy minimization. Although this modeling approach seems to be somewhat similar to ours, there are several distinctions: the muscle force model we present is more comprehensive, we model muscles additionally as geometric objects with impact on the shape of the skin, and the triangle mesh we use as skin has been obtained from a range scan of a human hand.

Brand *et al.*<sup>7</sup> performed measurements of hand and forearm muscles to obtain potential excursion and relative tension of the muscles. Potential excursion is the difference between maximal stretch and maximal contraction of a muscle, i.e. the distance through which a muscle is able to contract actively. They found the potential excursion to be equal to the resting length of the fibers of the muscle. Relative tension denotes the proportional tension of a muscle w.r.t. the overall amount of possible tension of all studied muscles. These numbers differ far less among individuals and within each individual over time than the absolute strength of a muscle.

Anthropometrical measurements have been carried out by Wagner<sup>43</sup>, who extensively measured size and joint mobility of the hands of pianists. He compared his results to studies about other musicians and non-musicians and found that in general piano players have greater mobility in their hands than the average.

## 2.2. Hand Models in Computer Graphics

In computer graphics, hand models have been developed for several typical applications. The most prominent application areas are model-based tracking (see for instance<sup>45</sup> for an overview), interactive grasping, and simulation systems used for e.g. surgery planning.

In<sup>33</sup>, a simple volume-based animatable hand model constructed from geometric primitives has been employed for tracking. The model includes anthropometrical and biomechanical constraints: the size of the palm is correlated to the length of the fingers and phalanges. Biomechanical laws determine the valid range and interdependencies of joint motion, thereby reducing the number of degrees of freedom of the model. Heap and Hogg<sup>15</sup> have built a statistical hand shape model from simplex meshes fitted to MRI data for their tracking system. For model-based finger motion capturing, Lin *et al.*<sup>28</sup> employ a learning approach for the hand configuration space to generate natural movement.

A parametric hand model has been designed for the semi-automatic grasping approach in<sup>29</sup>. In this model, skinning is based on joint-dependent local deformations, taking into account rounding at joints and bulging. Another approach to grasping is described in<sup>14</sup>. The system uses finite element simulation of the skin and the grasped object in order to simulate both skin and object deformations due to contact. In<sup>36</sup>, a simple hand model is described that likewise incorporates constraints on the movement range of joints. It was developed for the animation of semi-automatic knowledge-based grasping, where objects are approximated by primitives with individual grasping approach parameters. Another heuristic grasping system has been introduced in<sup>37</sup>. Objects are stored together with primitives associated with the graspable parts of the object. The final position of the hand is determined by inverse kinematics and collision detection. Huang *et al.*<sup>16</sup> extended the previous model. A multi-sensor approach for

collision detection has been added, where the sensors are constituted by spheres attached to the joint. Collision detection between hand and object is performed with these sensors to naturally place the hand around the object.

In<sup>21</sup>, artificial intelligence is used to position hand and wrist of a virtual violinist. Finger positions are determined by best-first search, while wrist position and orientation are decided by a neural network. Mulero *et al.*<sup>32</sup> present an anthropomorphic finger model with a tendon transmission system based on pulleys and a position controller. The controller is modeled by a neural network and transforms tendon pull into joint motion. The system can work in an agonist-antagonist fashion. A model of the hand and arms based on manifold mappings has been proposed by Kunii *et al.*<sup>23</sup>. They also consider inter-joint dependencies. Moccozzi *et al.*<sup>31</sup> use Dirichlet free-form deformations (DFFDs) to simulate the tissue and muscle layer between skin and bones. Muscles are not considered directly, but the use of DFFDs allows the authors to model wrinkles at joints and bulging of segments dependent on the angle of rotation of the respective proximal joint. Ip *et al.*<sup>18</sup> have built an anatomy-based hand model with muscles based on the work presented in<sup>4</sup>. The hand is modeled as a collection of hand segments connected by joints, where muscles are weightless expandable threads. Soft tissue, tendons, and ligaments are not modeled explicitly. Given the initial and final hand posture, the system is able to generate the in-between states. For describing the hand postures, the authors use the Hand Action Coding System<sup>17</sup>, a collection of muscle-based Hand Action Units that encode hand positions. Thompson *et al.*<sup>42</sup> presented a hand model capable of calculating relative muscle length, distance between pulley point / point of origin and transformed insertion point, moment arm, and moment potential for hand muscles during motion. A wireframe skeleton model is rendered together with the tendons, while the single parameters are displayed by bar graphs. Since the system was designed to aid medical doctors in planning tendon transfers, replacement of one tendon/muscle unit by another can be simulated. In<sup>30</sup>, the joint movements of a hand model composed of rigid bodies are constrained by biomechanical laws. The model was designed for use in animating American Sign Language. An approach for skinning a hand skeleton using eigendisplacements has been proposed in<sup>22</sup>. The resulting hand model can be animated in real-time using graphics hardware.

In addition to the literature on human hand models, several approaches for anatomical modeling and physics-based animation of human faces and bodies have been presented. In particular, the mass-spring system approaches in<sup>41, 26, 27</sup> and the muscle models proposed in<sup>38, 44, 19</sup> are of interest within the scope of this paper.

Concerning the use of feature points for model deformation, the work presented in<sup>40</sup> should be mentioned: anatomical models of articulated creatures equipped with feature

points can be morphed to obtain new models of similar shape. Based on input measurements, the structure hierarchy, bones, and muscles of the original model are deformed. From these deformed components, a new set of feature points is generated to deform the skin mesh using a local interpolation approach based on radial basis functions.

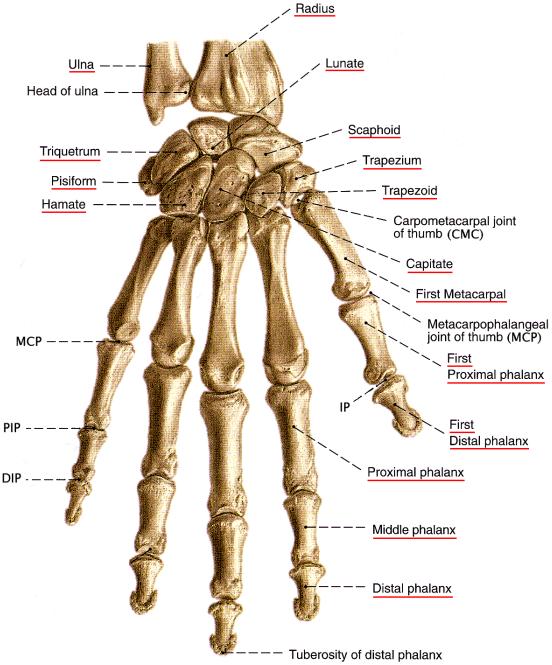
### 3. The Reference Hand Model

The central component of our system is a prototype hand model with anatomical structure, which is denoted as our *reference hand model* in the following. The building blocks of our reference hand model are:

- the *skin surface*, which is represented by a triangle mesh consisting of 3000 triangles;
- the *skeleton* of the hand, composed of 29 triangle meshes corresponding to the individual bones of the human hand and forearm (cf. Figure 1);
- a set of *virtual muscles*, which are embedded in between the skin surface and the skeleton;
- a *mass-spring system*, interlinking the skin, skeleton, and muscles;
- a *joint hierarchy*, which matches the structure of the skeleton, with an individually oriented coordinate system at each joint center defining valid axes of joint rotation.

The skin mesh of our reference hand model has been obtained by scanning a plaster cast of a human hand, see also the figure on the first page. The resulting triangle mesh has been reduced to a size of 3000 triangles to allow for real-time simulation of skin deformations. The triangle meshes of the individual bones have been taken from a publicly available skeleton model<sup>1</sup> and scaled to match the proportions of the skin mesh.

Using the hierarchy of coordinate systems, we can model the degrees of freedom (DOFs) for each joint easily. The only joints we do ignore are the joints between the individual wristbones (cf. Figure 1). This is justified, since their contribution to the overall movement is negligible. The PIP and DIP joints of the fingers and the IP joint of the thumb have one DOF each for flexion/extension, while the MCP joints of the fingers have a second DOF for adduction (towards the middle finger) and abduction (away from the middle finger). In addition, depending on the current amount of flexion or extension, the finger MCP joints exhibit some rotation around their long symmetry axis. Likewise, the CMC joint of the thumb is sometimes said to have three degrees of motion<sup>11</sup>. The impression of rotation around a third axis is evoked, because the two real axes are not completely perpendicular to each other (see<sup>8</sup>, p. 41). To overcome the restriction of two orthogonal DOFs, we model the MCP joints of the fingers and the thumb CMC joint as having three DOFs. The muscles must be designed to accommodate the dependencies between the flexion/extension and rotation axes: if a muscle flexes or extends the joint, it must also rotate it to



**Figure 1:** Bones of the human hand and forearm. Individual bone names are underlined in red. The metacarpal, proximal phalanx, and distal phalanx bones exist in each finger of the human hand, while the middle phalanx bones exist in all fingers but the thumb. Image taken from<sup>35</sup>.

some small degree. The CMC joints of the index and middle finger are fixed, while the ring and little finger CMC joints have two DOFs each with a very small range of motion.

Since muscles usually have greater strength and possible excursion than is required to move the limbs, it is also important to constrain the range of each DOF of the joints to avoid movement which is in reality prohibited by the form of the joints, by the joint capsules, and by ligaments. For each DOF, we set an upper and a lower limit according to<sup>28</sup>.

#### 3.1. Animation

Our reference hand model is animated exclusively through muscle contraction values given over time. These contraction values are specified in key frames with an arbitrary temporal distribution. During simulation, the contraction values are interpolated using a smooth spline function, which is evaluated at discrete points in time according to the desired rendering frame rate. At each point in time, the deformation of all muscles and the position of each bone is computed from the current contraction values. In turn, muscle and bone movements are used to update the positions of those nodes of the mass-spring system that attach to muscles and bones, respectively. In the final step of each simulation cycle, the

Lagrangian equations of motion are integrated through time for all nodes of the mass-spring system employing a Verlet leapfrog integration method. The resulting displacements of the nodes attached to the skin mesh represent the deformation of the skin surface. Details about the geometric muscle model, the mass-spring system, and the integration method can be found in<sup>19</sup>.

### 3.2. Rendering

Rendering is currently performed using plain OpenGL functionality. Conceptually, it would make no difference to output key frames for a more sophisticated rendering engine. However, we found the possibility to instantly view animations running in real-time worthwhile enough to accept the somewhat degraded rendering quality. Since the focus of this paper is on the geometry of our hand model and its deformation during animation, we have omitted textures that distract from the actual underlying geometry.

## 4. A Hybrid Muscle Model

Muscle mechanics of the human hand have evolved to a degree of complexity that is unique among mammals. This evolutionary process took place in order to allow us to perform fine motor manipulations and powerful manual work alike. Modeling and simulating all the subtle anatomical details of the muscles of the human hand is an impractical approach. In this section, we present a *hybrid muscle model*, which is flexible enough to cover the rich variety of muscle mechanics in the human hand and yet is easy to use.

Our hybrid muscle model comprises *pseudo muscles* and *geometric muscles*. Both of these muscle types are animated exclusively through muscle contraction values within the range [0, 1], where 0 means no contraction at all, and 1 means full contraction. Pseudo muscles directly control the rotation of the bones of the hand, while geometric muscles account for skin tissue deformation through physics-based simulation employing a mass-spring system that connects muscles, skin, and bones. Though each of these two muscle types can be used individually, we typically use a combination of a pseudo muscle and a geometric muscle to represent the effects of an anatomical muscle in the human hand. For instance, the *opponens pollicis* is implemented by a pseudo muscle that rotates the *proximal phalanx* of the thumb and by a geometric muscle (muscle (2) in Figure 2 left) that bulges the skin. Table 1 lists the pseudo muscles of our system together with their specific parameters (taken from<sup>8,2</sup>).

For each frame during animation, all pseudo muscles are evaluated to update the position of the bones. The segments of geometric muscles that are attached to bones are transformed correspondingly. Next, the geometric muscles' deformation due to contraction is computed. Finally, the mass-spring system is updated to evaluate the resulting skin deformation.

### 4.1. Pseudo Muscles

Pseudo muscles are virtual muscles that convert a given contraction value  $c \in [0, 1]$  into rotation angles  $\varphi_k$  for each DOF of each joint  $\mathcal{J}_k$  they affect. Our model for this conversion is based on anatomical data and mechanical laws. However, our implementation is only valid under two assumptions:

1. The bones that are rotated are long bones, which are represented as solid cylinders in our mathematical model. This is true for all bones of the human hand with the exception of the wristbones.
2. When rotating a hierarchy of bones, the number of levels in that hierarchy has to be less or equal to three. In our hand model, this is true for the fingers starting at the knuckles and for the thumb starting at the *trapezium*.

The second restriction is solely due to computational efficiency. In Section 4.1.2 we describe a technique to efficiently compute the rotation of chains of bones up to length three, which imposes the restriction above. To avoid this limitation, we are currently investigating the possibility to apply similarity transforms and the parallel axis (Steiner) theorem<sup>3</sup> for transforming inertia tensors from one coordinate frame to another. Our preliminary results show that this approach removes the limitation of the hierarchy depth at the cost of more expensive computations. In addition, the moment of inertia needs to be stored as a tensor to allow for the application of the parallel-axis theorem for non-parallel rotation axes. In the approach described below, we simplify our model by taking into account only the magnitude of torque and moment of inertia. Treating these variables as vector-valued tensors would render the computational costs of evaluating our model too high for real-time simulation.

Each pseudo muscle represents an anatomical muscle with a given maximum contraction force  $\vec{F}_{\max}$ . Relative values of  $\vec{F}_{\max}$  for all relevant hand muscles are listed in<sup>7</sup>, the direction of  $\vec{F}_{\max}$  has to be estimated from the layout and the attachment point of the muscle/tendon (see, e.g.,<sup>35</sup>). The contraction force of a muscle is not constant, but depends on the current fiber length  $\ell$  of the muscle: a muscle that is either (passively) stretched or (actively) contracted has a lower contraction force than a muscle at its fiber resting length  $\ell_0$ . The nonlinear relationship between the contraction force  $\vec{F}_{\text{contr}}$  and  $\ell$  is depicted in<sup>8</sup>. We fitted a quadratic curve to the diagrams shown there and obtained the relationship:

$$\vec{F}_{\text{contr}}(\ell) = [1 - 4 \cdot (\ell/\ell_0 - 1.1)^2] \cdot \vec{F}_{\max}.$$

In addition to the contraction force, each anatomical muscle exhibits a stretch force: a muscle that is (passively) stretched counteracts the stretch with a force  $\vec{F}_{\text{stretch}}$ , which depends on the muscle's current fiber length  $\ell$ . Obviously, the stretch force is equal to zero if  $\ell < \ell_0$ . Again, we fitted a curve to the diagrams shown in<sup>8</sup> and obtained:

$$\vec{F}_{\text{stretch}}(\ell) = \begin{cases} 2.77 \cdot (\ell/\ell_0 - 1)^2 \cdot \vec{F}_{\max}, & \ell \geq \ell_0 \\ 0, & \ell < \ell_0 \end{cases}.$$

According to <sup>8</sup>, the inequation  $0.6\ell_0 \leq \ell \leq 1.6\ell_0$  must hold, i.e. a muscle cannot become arbitrarily short or elongated. These upper and lower limits for  $\ell$  are — both in reality and in our model — usually never reached, since the corresponding joints are constrained in their rotations (cf. Section 3). The current fiber length  $\ell$  of a pseudo muscle is initialized to the resting length  $\ell_0$  and updated by the arc length of the rotation, see below.

#### 4.1.1. Rotation of a Single Bone

To see how our conversion model works, let us assume for now that there is exactly one (cylindrical) bone that is rotated about the joint's axis of rotation due to the contraction of one pseudo muscle. Given a contraction value  $c \in [0, 1]$ , the resulting force the muscle exerts on the bone is:

$$\vec{F} = c \cdot \vec{F}_{\text{contr}}(\ell) + \vec{F}_{\text{stretch}}(\ell). \quad (1)$$

Let  $\vec{r}$  denote the lever arm of the force working point. The amount of torque is computed as follows:

$$T = \text{sgn}(\langle \vec{a}, \vec{r} \times \vec{F} \rangle) \cdot \|\vec{r} \times \vec{F}\|, \quad (2)$$

where  $\langle \vec{a}, \vec{r} \times \vec{F} \rangle$  denotes the dot product of the rotation axis and the vector-valued torque. In addition, the following relationship between torque  $T$ , angular velocity  $\omega$ , and moment of inertia  $J$  holds:

$$T = J \cdot \frac{d\omega}{dt}. \quad (3)$$

Since the angular velocity  $\omega$  equals the first temporal derivative of the rotation angle  $\phi$ :

$$\omega = \frac{d\phi}{dt}, \quad (4)$$

we can discretize time and compute:

$$\Delta\omega \stackrel{(3)}{=} \Delta t \cdot J^{-1} \cdot T,$$

$$\omega_{\text{new}} = \omega_{\text{old}} + \Delta\omega,$$

$$\Delta\phi \stackrel{(4)}{=} \Delta t \cdot \omega_{\text{new}},$$

$$\ell \leftarrow \ell - \Delta\phi \cdot \|\vec{r}\|.$$

Using this approach, we can compute the increment  $\Delta\phi$  of the rotation angle from a contraction value  $c$ . The only unknown variable is the moment of inertia  $J$ . Although  $J$  is quite expensive to compute for an arbitrarily shaped body, it can be easily computed for a solid cylinder of length  $l$  and mass  $m$  that is rotated about an axis orthogonal to its length axis and passing through one of its ends <sup>13</sup>:

$$J = \frac{1}{3} \cdot m \cdot l^2. \quad (5)$$

In our case, the length  $l$  is the length of the bone rotated about the joint's axis. The mass  $m$ , however, is the mass of the bone plus the mass of the tissue surrounding the bone. Values for this bone-plus-tissue mass can be found in <sup>4</sup>.

The above formulas do not consider friction yet. This

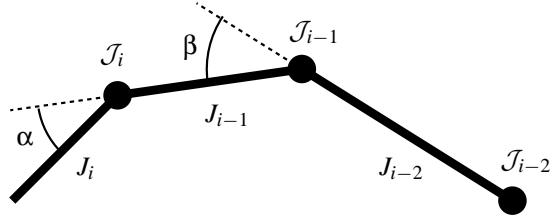


Figure 3: Moments of inertia for a chain of bones.

means that a rotation, once it has started due to muscle contraction, will not stop again. To take into account friction, we have to modify Equation (2) by subtracting the torque of friction:

$$T = \text{sgn}(\langle \vec{a}, \vec{r} \times \vec{F} \rangle) \cdot [\|\vec{r} \times \vec{F}\| - \mu \cdot |\omega_{\text{old}}|], \quad (2')$$

where  $\mu$  is the coefficient of friction. In accordance to medical literature, we use  $\mu = 0.015$ .

Finally, we extend our mathematical model to allow for an arbitrary number  $n$  of pseudo muscles that affect the rotation of the bone. Each pseudo muscle  $i$  ( $i = 1, \dots, n$ ) exerts the force:

$$\vec{F}_i = c_i \cdot \vec{F}_{\text{contr},i}(\ell_i) + \vec{F}_{\text{stretch},i}(\ell_i). \quad (1')$$

Since each muscle has its own lever arm  $\vec{r}_i$ , the total amount of torque is given by:

$$T = \text{sgn}(\langle \vec{a}, \vec{T} \rangle) \cdot [\|\vec{T}\| - \mu \cdot |\omega_{\text{old}}|], \quad (2'')$$

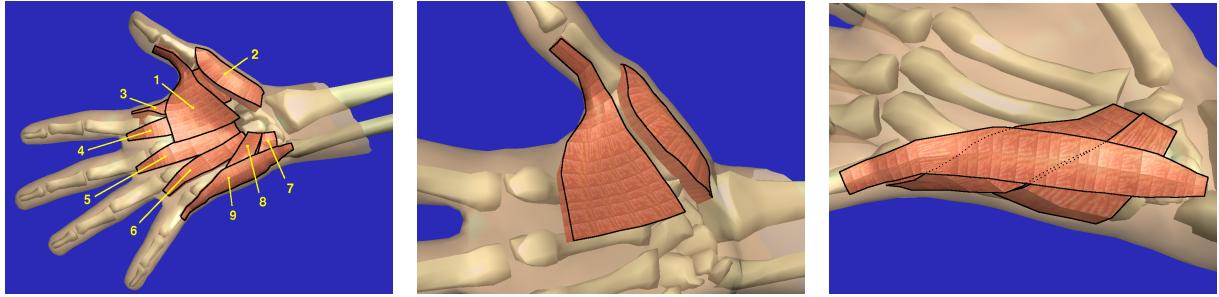
with

$$\vec{T} = \sum_{i=1}^n \vec{r}_i \times \vec{F}_i.$$

#### 4.1.2. Rotation of Chains of Bones

For the derivation of the conversion formulas in the previous paragraph we assumed that the rotated bone is an end segment, i.e. one of the *distal phalanges*. If, however, we want to rotate a chain of bones, for instance the three *phalanges* of a finger, the moment of inertia  $J$  depends on the position of all bones in that chain.

Figure 3 depicts this situation: when rotating about joint  $J_i$ , the moment of inertia  $J_i$  of the rotated bone is constant and can be computed according to Equation (5). The total moment of inertia for a rotation about joint  $J_{i-1}$  is composed of  $J_{i-1}$  and the moment of inertia of the end segment. The latter, however, is not simply  $J_i$  in this case: the axis of rotation does not pass through the end of the rotated end segment as required for Equation (5). Thus the position of the end segment has to be transformed into the coordinate system of  $J_{i-1}$  and the moment of inertia  $J_i^*$  is computed by summing up the squared distances of the transformed bone mesh vertices to the rotation axis multiplied by the mass of



**Figure 2:** Geometric muscles of our hand model. Left: adductor pollicis (1), opponens pollicis (2), 1st dorsal interosseus (3), 1st palmar interosseus (4), 2nd palmar interosseus (5), 3rd palmar interosseus (6), opponens digiti minimi (7), flexor digiti minimi brevis (8), and abductor digiti minimi (9). Middle: the complex shape of muscles can be observed in this close-up view of (1) and (2) with all other muscles removed. Right: different muscle layers are set up automatically. The vertical muscles (7) and (8) slide freely below the horizontal muscle (9).

the bone. This computation becomes more and more costly when longer chains of bones are rotated.

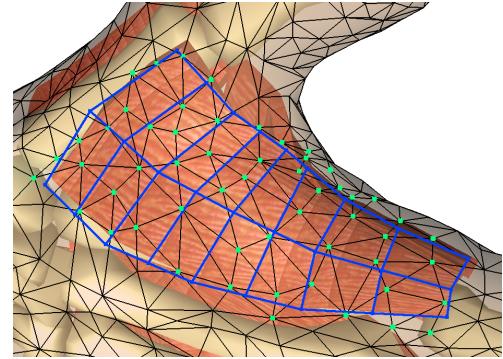
Fortunately, the moment of inertia  $J_i^*$  of the transformed bone depends only on the rotation angle  $\alpha$ . Thus we precompute  $J_i^*(\alpha)$  for a discrete set of angles (typically in steps of five degrees) and store the array  $J_i^*[\alpha]$  in the joint  $\mathcal{J}_i$  for further look-up. The total moment of inertia  $J$  for a rotation about  $\mathcal{J}_{i-1}$  can thus be simply computed as  $J = J_{i-1} + J_i^*[\alpha]$ . Similarly, the total moment of inertia for a rotation about  $\mathcal{J}_{i-2}$  is given by the sum  $J_{i-2} + J_{i-1}^*[\beta] + J_i^*[\alpha][\beta]$ .

Precomputing the moments of inertia for chains of bones with more than three segments would require storing arrays of dimension three and more in the joints. To avoid this exhaustive memory consumption, we restrict the computation of the moments of inertia to hierarchies with at most three levels. For the rotation of the complete hand about the wrist we assume a constant moment of inertia of the hand.

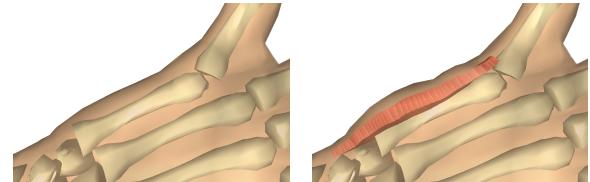
#### 4.2. Geometric Muscles

In our system, geometric muscles are embedded in between the skin surface and the underlying bone structure. Geometric muscles have an actual geometric shape assigned to them, which deforms and bulges during contraction. Springs are used to connect the surface of the muscle's geometry to skin and bones. We have adopted the approach presented in<sup>19</sup> for the embedding of muscles into a mass-spring system. However, some modifications of that approach were necessary to allow for a more complex muscle layout. In particular, we have introduced the following changes:

- Each individual muscle has its own minimum and maximum thickness. Rather thick muscles, e.g. the *opponens pollicis* (2), can thus be created as well as thin sheet muscles, e.g. the *adductor pollicis* (1), see Figure 2 (middle).
- The distance between the skin surface and the surface of the muscle can be set individually for each muscle to al-



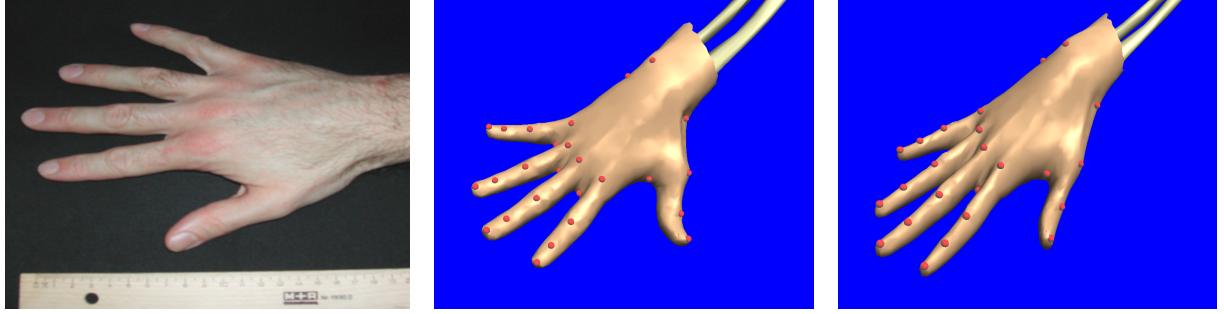
**Figure 4:** Geometric muscles are created automatically from a muscle grid (shown in blue) painted onto the skin surface. The green dots mark the vertices of the skin mesh (shown as a wireframe), which are influenced by the muscle.



**Figure 5:** Bulging of geometric muscles. Left: pseudo muscles are used to move the bones. Right: combining pseudo and geometric muscles results in additional skin deformation due to bulging.

low for several layers of muscles (e.g. superficial and deep layer) to be created automatically, see Figure 2 (right).

- Muscles are allowed to attach to bones on both muscle ends. Such types of muscles do not exist among the facial muscles (with the exception of the *masseter*, which was not present in<sup>19</sup>), but are prevalent in the human hand.



**Figure 6:** Deformation of the reference hand model. Left: photograph of an individual’s hand including a ruler. Middle: position of the 26 source feature points on the reference hand model. Right: resulting hand model after applying the warping function.

- Muscles may be assigned to several individual bones. Thus, individual segments of large or long muscles move with the bones they are assigned to. The *abductor digiti minimi* (9), for instance, is assigned to the *carpal bones* (wristbone), and the *metacarpal* and *proximal phalanx* of the little finger (cf. Figures 1 and 2 (left)).

Geometric muscles are created by interactively painting *muscle grids* onto the skin surface (see Figure 4). From the shape of a muscle grid, the corresponding muscle is created automatically to fit in between skin and bone surfaces, taking into account the individual muscle thickness and skin distance. This fitting process is similar to the one described in<sup>19</sup>, with the obvious extensions for individual muscle parameters.

During contraction of a geometric muscle, the assigned geometric shape deforms and bulges, see Figure 5. On the muscle surface, the attachment points of the springs connecting the muscle to skin and bones move accordingly. The displacement of these nodes exerts force on the mass-spring mesh, which is updated to compute the corresponding skin deformation (cf. Section 3.1).

## 5. New Hand Models from Photographs

We employ a deformation technique based on feature points to warp the complete reference hand model to an individual hand model. Our approach is similar in spirit to the technique proposed by Kähler et al.<sup>20</sup> for deformation of human head models. However, we do not require a 3D target hand model to be obtained in a time-consuming scanning process. Instead, we use a simple photograph of the individual hand to be modeled. The photograph merely needs to show a simple ruler as depicted in Figure 6 (left). Since there are no other prerequisites for the photograph, low-cost consumer cameras can be used for the acquisition.

First, we identify a small set of feature points in the input photograph. Our reference hand model is already tagged with the same feature points by default. Section 5.1 presents details about this step. Next, the complete structure of the

reference hand model is deformed to match the shape of the individual hand from the photograph. The warp function is set up using correspondence of feature points, see Section 5.2 for the details.

### 5.1. Feature Points

We use a small set of feature points on both the 3D reference hand model and in the 2D photograph. These feature points can be easily identified without anatomical knowledge. In the following, the feature points on the reference hand model are denoted as the *source feature points*, whereas those in the input photograph are called *target feature points*.

The reference hand model is equipped with 26 source feature points by default (T = thumb, I = index finger, M = middle f., R = ring f., L = little f.; cf. Figures 1 and 6 (middle)):

# location

---

5	tip of {T,I,M,R,L}
4	distal interphalangeal joint (DIP) of {I,M,R,L}
1	interphalangeal joint (IP) of T
4	proximal interphalangeal joint (PIP) of {I,M,R,L}
5	metacarpophalangeal joint (MCP) of {T,I,M,R,L}
4	interdigital skin (between each pair of adjacent fingers)
2	radial and ulnar wrist (inner and outer side of the wrist)
1	head of ulna

---

Upon loading the input photograph into our system, the user is asked to identify as many target feature points from the above set as possible in the photograph. Feature points whose positions are not clearly visible in the photograph can be omitted. In our simulations, we obtained reasonable results using a subset of only 16 feature points.

In addition to selecting the target feature points, a calibration process is carried out to measure the size of the hand

in the input photograph. To this end, the user performs two mouse clicks at a known distance on the division scale of the ruler shown in the photograph, for instance at the points “0 cm” and “20 cm”. From their real world distance, the system automatically computes the scale of the photograph and uses this information to transform the positions of the target feature points into the coordinate system of the reference hand model.

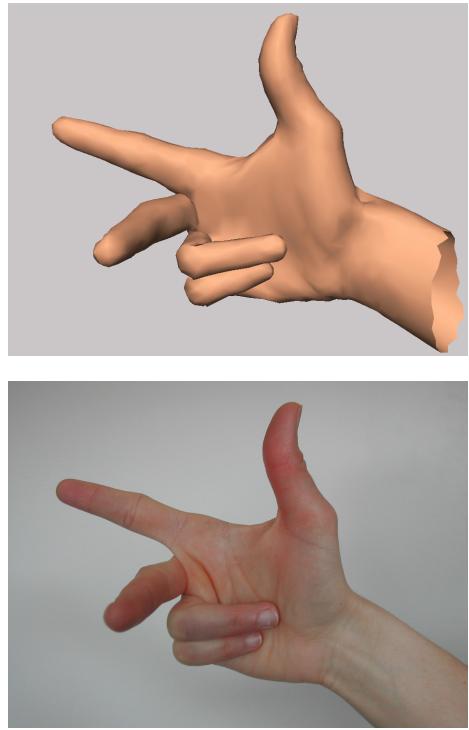
## 5.2. Warping the Reference Hand Model

Given two sets of  $N$  corresponding source and target feature points, we are looking for a function  $\mathbf{f}$  that maps the source feature points  $\mathbf{s}_i$  to the target feature points  $\mathbf{t}_i$ , ( $i = 1, \dots, N$ ). A natural solution to this interpolation problem is to employ a radial basis function (RBF), see for instance<sup>10</sup> for mathematical details. We use biharmonic basis functions  $\Phi_i(\mathbf{x}) := \|\mathbf{x} - \mathbf{s}_i\|_2$ , which minimize bending energy<sup>12</sup>. This choice is in consonance with Bookstein’s suggestion to use thin-plate splines for the deformation of biological tissues<sup>5,6</sup>.

Before setting up the function  $\mathbf{f}$ , we need to transform the target feature points  $\mathbf{t}_i$  into the coordinate system of the  $\mathbf{s}_i$ . This is necessary to make the transformed hand model appear in approximately the same place as the initial reference hand model. We transform the 2D points  $\mathbf{t}_i$  into the fitting plane of the 3D points  $\mathbf{s}_i$  using a rigid body transformation. The (uniform) scaling factor of this transformation is taken from the calibration step described in the previous section. Within the fitting plane of the  $\mathbf{s}_i$ , the target feature points are rotated to align the largest diameter of their set to the largest diameter of the set  $\{\mathbf{s}_i\}$ . Next, we assign to each target feature point the height of its corresponding source feature point above/below the fitting plane. These heights are additionally scaled by a (uniform) scaling factor obtained from the ratio of the largest diameters of the source and target feature points, respectively. Converting the 2D target feature points into 3D points is essential to ensure that the transformed hand model will not be flattened but possesses a thickness proportional to its overall size. Finally, we can set up our radial basis warping function as described in standard literature<sup>34,10</sup>.

After setting up the warping function  $\mathbf{f}$ , we transform the complete structure of our reference hand model as follows:

1. Skin and bone meshes are transformed by applying the function  $\mathbf{f}$  to each vertex of the meshes. The connectivity of the meshes is not changed.
2. Joint positions and positions of feature points are transformed in the same way by direct application of the warping function.
3. Geometric muscles are defined by muscle grids (cf. Section 4.2). To warp the muscles, only the control points of the muscle grids have to be transformed. The shape of the warped muscles is computed automatically to fit in between the transformed skin and bone meshes.



**Figure 7:** A typical computer graphics scientist’s hand pose. Side-by-side comparison of our hand model (top) with a photograph of the hand that was scanned to build the model (bottom). To allow for a better comparison of the actual geometries, textures and sophisticated shaders have been omitted from the hand model.

4. Pseudo muscles do not have any spatial parameters that need to be transformed.

## 6. Results

We created several animation sequences to verify the efficiency and correctness of our hybrid muscle model. In spite of some simplifications, we found our pseudo muscle model to work very well and yield plausible results. Figure 7 shows a side-by-side comparison of our hand model with a photograph of the (female) hand that has been scanned to build the reference hand model. Clearly, our hybrid muscle model is able to correctly reproduce the pose of the hand. When creating a new animation, however, estimating the right muscle contraction values (i.e. those that result in the desired movement of the fingers) is not always a straightforward process. We are currently working on an optimization process that computes minimal energy muscle contractions for a given target position of the fingers to eliminate the process of specifying individual muscle contraction values. Creating the geometric muscles is simple and fast: the complete set of geo-

metric muscles shown in Figure 2 (left) has been created in less than an hour.

Using our hybrid muscle model, animations were running at interactive frame rates of 5–10 fps on a 1.7GHz Pentium Xeon machine with a GeForce3 graphics board. The main bottleneck is the integration of the equations of motion for the mass-spring network, which comprises approximately 1500 nodes and 4500 springs. Yet our integration technique runs stable for a reasonable choice of stiffness parameters for the skin model. In all our tests, the skin mesh never lost integrity after mass-spring simulation. In particular, our method does not break the skin while moving from pose to pose, since the connectivity of the skin mesh is never changed.

Our deformation technique works reliably and is easy to use. We warped our reference hand model, which has been built from scan data of a female, to match the size and proportions of a male's and a child's hand. The interactive specification of the target feature points in the photograph takes about a minute. Since all components of the hand model are transformed, the resulting hand model is instantly animatable. However, the animation parameters (i.e. the muscle contraction values) have to be adapted to the warped hand model: different proportions of reference and target hand model result in different torques, moments of inertia, and finally different rotation angles.

## 7. Conclusion and Future Work

We presented an approach for the construction and animation of human hand models with underlying anatomical structure. Our system is built around a reference hand model, which is animated using muscle contraction values. We introduced a hybrid muscle model that comprises pseudo muscles and geometric muscles. While pseudo muscles control the rotation of bones based on anatomical data and mechanical laws, the deformation of geometric muscles causes realistic bulging of the skin tissue. As a result, the created animations automatically exhibit anatomically and physically correct behavior. In addition, we proposed a deformation technique based on feature points to create individual hand models from photographs. Warping the complete structure of the reference hand model results in deformed hand models that are instantly animatable.

Although our system is working reliably and rather efficiently, there are further ways of improvement. Ideally, geometric muscles should move the bones. This, however, involves modeling tendons as well as setting up a mass-spring system where rigid objects (such as the bones) can be moved due to spring forces. In addition, gravity should be included into our hybrid muscle model. While the effect of gravity is probably negligible for the deformation of skin tissue in the human hand, it plays an important role for the computation of bone positions from given muscle contraction values.

To facilitate the creation of animation sequences, we are working on an optimization process that computes minimal energy muscle contractions for a given target position of the fingers. Moreover, it would be helpful to include collision detection among the parts of the hand. Given these two add-ons (optimization process and collision detection), grasping of external objects would be rather easy to implement.

Finally, we would like to automatically generate textures from the photographs that are used to create individual hand models. However, the parameterization of the skin mesh is not trivial, if the texture is to be used for OpenGL rendering.

## Acknowledgments

The authors would like to thank Kolja Kähler for his valuable comments regarding the geometric muscle model and the mass-spring system.

## References

1. 3D cafe. Arm skeleton. <http://www.3dcafe.com/asp/anatomy.asp>.
2. K. An, Y. Ueba, E. Chao, W. Cooney, and R. Linscheid. Tendon excursion and moment arm of index finger muscles. *J. Biomechanics*, 16(6):419–425, 1983.
3. H. Baruh. *Analytical Dynamics*. McGraw-Hill, New York, 1998.
4. E. Biryukova and V. Yourovskaya. A Model of Hand Dynamics. In F. Schuind, K. An, W. Cooney, and M. Garcia Elias, editors, *Advances in the Biomechanics of Hand and Wrist*, pages 107–122. Plenum Press, New York, 1994.
5. F. L. Bookstein. *Morphometric Tools for Landmark Data*. Cambridge University Press, 1997.
6. F. L. Bookstein. Shape and the Information in Medical Images: A Decade of the Morphometric Synthesis. *Computer Vision and Image Understanding*, 66(2):97–118, 1997.
7. P. W. Brand, R. B. Beach, and D. E. Thompson. Relative tension and potential excursion of muscles in the forearm and hand. *J. Hand Surgery*, 6(3):209–219, 1981.
8. P. W. Brand and A. M. Hollister. *Clinical Mechanics of the Hand*. Mosby - Year Book, Inc., St. Louis, MO, 3rd edition, 1999.
9. B. Buchholz and T. Armstrong. A kinematic model of the human hand to evaluate its prehensile capabilities. *J. Biomechanics*, 25(2):149–162, 1992.
10. J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and Representation of 3D Objects With Radial Basis Functions. In Eugene Fiume, editor, *Computer Graphics (SIGGRAPH '01 Conf. Proc.)*, pages 67–76. ACM SIGGRAPH, August 2001.
11. R. Chase. Examination of the Hand and Relevant Anatomy. In *Plastic Surgery*, volume 7, chapter 89, pages 4247–4284. 1990.
12. J. Duchon. Spline minimizing rotation-invariant semi-norms in Sobolev spaces. In W. Schempp and K. Zeller, editors, *Constructive Theory of Functions of Several Variables*, volume 571 of *Lecture Notes in Mathematics*, pages 85–100, 1977.

13. H. Goldstein. *Classical Mechanics*. Prentice Hall, 3rd edition, 2002.
14. J.-P. Gourret, N. Magnenat-Thalmann, and D. Thalmann. Simulation of Object and Human Skin Deformations in a Grasping Task. In Jeffrey Lane, editor, *Computer Graphics (SIGGRAPH '89 Conf. Proc.)*, volume 23, pages 21–30. ACM SIGGRAPH, 1989.
15. T. Heap and D. Hogg. 3D deformable hand models. In *Proc. Gesture Workshop '96*, pages 131–139, 1986.
16. Z. Huang, R. Boulic, N. Magnenat-Thalmann, and D. Thalmann. A Multi-sensor Approach for Grasping and 3D Interaction. In *Proc. CGI '95*, pages 235–254, 1995.
17. H. Ip, S. Chan, and M. Lam. HACS: Hand Action Coding System for Anatomy-Based Synthesis of Hand Gestures. In *Proc. Int'l. Conf. Systems, Man, and Cybernetics 1998*, pages 1307–1312, 1997.
18. H. Ip, S. Chan, and M. Lam. Hand Gesture Animation from Static Postures Using an Anatomy-Based Model. In *Proc. Computer Graphics International (CGI 2000)*, pages 29–36, 2000.
19. K. Kähler, J. Haber, and H.-P. Seidel. Geometry-based Muscle Modeling for Facial Animation. In *Proc. Graphics Interface 2001*, pages 37–46, June 2001.
20. K. Kähler, J. Haber, H. Yamauchi, and H.-P. Seidel. Head shop: Generating animated head models with anatomical structure. In *Proc. ACM SIGGRAPH Symposium on Computer Animation (SCA '02)*, pages 55–64, July 2002.
21. J. Kim, F. Cordier, and N. Magnenat-Thalmann. Neural Network-based Violonist's Hand Animation. In *Proc. Computer Graphics International (CGI 2000)*, pages 37–41, 2000.
22. P. G. Kry, D. L. James, and D. K. Pai. EigenSkin: Real Time Large Deformation Character Skinning in Hardware. In *Proc. ACM SIGGRAPH Symposium on Computer Animation (SCA '02)*, pages 153–159, July 2002.
23. T. Kunii, Y. Tsuchida, H. Matsuda, M. Shirahama, and S. Miura. A model of the hands and arms based on manifold mappings. In *Proc. Computer Graphics International (CGI '93)*, pages 381–398, 1993.
24. J. Landsmeer. Studies in the anatomy of articulation. *Acta morphologica Neerlando-Scandinavia*, 3:287–303, 1961.
25. K.-H. Lee and K. Kroemer. A Finger Model with Constant Tendon Moment Arms. In *Proc. Human Factors and Ergonomics Society 37th Annual Meeting*, pages 710–714, 1993.
26. Y. Lee, D. Terzopoulos, and K. Waters. Constructing Physics-based Facial Models of Individuals. In *Proc. Graphics Interface '93*, pages 1–8, May 1993.
27. Y. Lee, D. Terzopoulos, and K. Waters. Realistic Modeling for Facial Animations. In Robert Cook, editor, *Computer Graphics (SIGGRAPH '95 Conf. Proc.)*, pages 55–62. ACM SIGGRAPH, August 1995.
28. J. Lin, Y. Wu, and T. Huang. Modeling the Constraints of Human Hand Motion. In *Proc. Workshop on Human Motion*, pages 121–126, 2000.
29. N. Magnenat-Thalmann, R. Lemperrière, and D. Thalmann. Joint-dependent local deformations for hand animation and object grasping. In *Proc. Graphics Interface '88*, pages 26–33, 1988.
30. J. McDonald, J. Toro, K. Alkoby, A. Berthiaume, R. Carter, P. Chomwong, J. Christopher, M. Davidson, J. Furst, B. Konie, G. Lancaster, L. Roychoudhuri, E. Sedgewick, N. Tomuro, and R. Wolfe. An improved articulated model of the human hand. *The Visual Computer*, 17(3):158–166, 2001.
31. L. Moccozet and N. Magnenat-Thalmann. Dirichlet Free-Form Deformations and their Application to Hand Simulation. In *Proc. Computer Animation '97*, pages 93–102, 1997.
32. J. I. Mulero, J. Feliú Batlle, and J. López Coronado. Parametric Neurocontroller for Positioning of an Anthropomorphic Finger Based on an Opponent-Driven Tendon Transmission System. In *Proc. IWANN '01*, pages 47–54, 2001.
33. H. Ouhaddi and P. Horain. Conception et ajustement d'un modèle 3D articulé de la main. In *Actes des 6èmes journées du Groupe de Travail Réalité Virtuelle*, volume 12/13, pages 83–90, 1998.
34. F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. H. Salesin. Synthesizing Realistic Facial Expressions from Photographs. In Michael F. Cohen, editor, *Computer Graphics (SIGGRAPH '98 Conf. Proc.)*, pages 75–84. ACM SIGGRAPH, July 1998.
35. R. Putz and R. Pabst, editors. *Atlas of Human Anatomy — Volume 1: Head, Neck, Upper Limb*. Lippincott Williams & Wilkins, Philadelphia, 13th edition, 2001.
36. H. Rijpkema and M. Girard. Computer Animation of Knowledge-Based Human Grasping. In Thomas W. Sederberg, editor, *Computer Graphics (SIGGRAPH '91 Conf. Proc.)*, volume 25, pages 339–348. ACM SIGGRAPH, 1991.
37. R. Mas Sanso and D. Thalmann. A Hand Control and Automatic Grasping System for Synthetic Actors. *Computer Graphics Forum*, 13(3):167–177, 1994.
38. F. Scheepers, R. E. Parent, W. E. Carlson, and S. F. May. Anatomy-Based Modeling of the Human Musculature. In Turner Whitted, editor, *Computer Graphics (SIGGRAPH '97 Conf. Proc.)*, pages 163–172. ACM SIGGRAPH, August 1997.
39. L. Sibille, M. Teschner, S. Srivastava, and J.-C. Latombe. Interactive Simulation of the Human Hand. In *CARS '02*, pages 7–12, 2002.
40. M. Simmons, J. Wilhelms, and A. Van Gelder. Model-based Reconstruction for Creature Animation. In *Proc. ACM SIGGRAPH Symposium on Computer Animation (SCA '02)*, pages 139–146, 2002.
41. D. Terzopoulos and K. Waters. Physically-based Facial Modelling, Analysis, and Animation. *J. Visualization and Computer Animation*, 1(2):73–80, 1990.
42. D. Thompson, W. Buford, L. Myers, D. Giurintano, and J. Brewer III. A Hand Biomechanics Workstation. In John Dill, editor, *Computer Graphics (SIGGRAPH '88 Conf. Proc.)*, volume 22, pages 335–343. ACM SIGGRAPH, 1988.
43. Ch. Wagner. The pianist's hand: Anthropometry and biomechanics. *Ergonomics*, 31(1):97–131, 1988.
44. J. Wilhelms and A. Van Gelder. Anatomically Based Modeling. In Turner Whitted, editor, *Computer Graphics (SIGGRAPH '97 Conf. Proc.)*, pages 173–180. ACM SIGGRAPH, August 1997.
45. Y. Wu and T. Huang. Hand Modeling, Analysis, and Recognition. *IEEE Signal Processing Magazine*, 18(3):51–60, 2001.

anatomical name	$\ell_0$	joint(s) / DOF	$  \vec{r}  $	anatomical name	$\ell_0$	joint(s) / DOF	$  \vec{r}  $
flexor carpi radialis	52	wrist flex. wrist abd.	17.5 10.5	abductor digiti minimi	40	CMC pinky opp. MCP pinky abd.	6 4
palmaris longus	50	wrist flex. wrist abd.	21 1.5			PIP pinky ext. DIP pinky ext.	2.5 2
flexor digitorum superf. index	72	wrist flex. wrist add. MCP index flex. MCP index add. PIP index flex.	15 3 11.9 3 6.2	flexor digiti minimi brevis	34	CMC pinky opp. MCP index flex. MCP index abd.	6 4 4
flexor digitorum superf. middle (analogously: flexor digitorum superf. ring flexor digitorum superf. pinky)	70	wrist flex. wrist add. MCP middle flex. MCP middle add. PIP middle flex.	15 3.0 11.9 1.7 6.2	opponens digiti minimi	34	CMC pinky opp.	6
flexor carpi ulnaris	42	wrist flex. wrist add.	18.5 15	abductor pollicis brevis	37	CMC thumb opp. CMC thumb abd.	3.5 7.5
flexor digitorum prof. index (analogously: flexor digitorum prof. middle flexor digitorum prof. ring flexor digitorum prof. pinky)	66	wrist flex. wrist add. MCP index flex. MCP index add. PIP index flex. DIP index flex.	6 13 11.1 6 7.9 4.1	flexor pollicis brevis	36	CMC thumb opp. CMC thumb add.	9 1
flexor pollicis longus	59	wrist flex. wrist abd. CMC thumb add. CMC thumb opp. MCP thumb flex. IP thumb flex.	5 13 10 10 7.5 5.5	opponens pollicis	24	CMC thumb opp. CMC thumb add.	4 8.5
extensor carpi radialis longus	93	wrist ext. wrist abd.	10 21	adductor pollicis	36	CMC thumb opp. CMC thumb add.	4.5 9
extensor carpi radialis brevis	61	wrist ext. wrist abd.	13 24	lumbrical I	55	MCP index flex. MCP index radial abd.	9.3 4.8
extensor digitorum index (analogously: extensor digitorum middle extensor digitorum ring extensor digitorum pinky)	55	wrist ext. wrist add. MCP index ext. MCP index abd. PIP index ext. DIP index ext.	13 7.5 8.6 0.2 2.8 2.2	lumbrical II	66	PIP index ext. DIP index ext.	1.8 0.7
extensor digiti minimi	59	wrist ext. wrist add. MCP pinky ext. PIP pinky ext. DIP pinky ext.	13 7.5 8.6 2.6 1.9	lumbrical III	60	MCP ring flex. MCP ring radial abd.	5 4.8
extensor carpi ulnaris	45	wrist ext. wrist add.	6 25	lumbrical IV	49	PIP ring ext. DIP ring ext.	1.8 0.7
extensor pollicis longus	57	wrist ext. wrist abd. CMC thumb ext. CMC thumb add. MCP thumb ext. IP thumb ext.	9 10.5 5 10 2.5 2	palmar interosseus I	15	MCP index flex. MCP index add.	6.6 5.8
extensor indicis	55	wrist flex. wrist abd. MCP index ext. MCP index add. PIP index ext. DIP index ext.	1.4 0.4 9 1.3 2.6 1.9	palmar interosseus II	15	DIP index ext. PIP index ext.	2.6 1.6
abductor pollicis longus	46	wrist flex. wrist abd. CMC thumb ext.	7.4 24 0.5	palmar interosseus III	15	MCP pinky flex. MCP pinky add.	6.6 5.8
extensor pollicis brevis	43	wrist flex. wrist abd. CMC thumb ext. CMC thumb abd. MCP thumb ext.	3.2 23 4.5 3 3	dorsal interosseus I	25	DIP pinky ext. PIP pinky ext.	2.6 1.6
				dorsal interosseus II	25	MCP index flex. MCP index abd.	3.7 6.1
				dorsal interosseus III	25	PIP index ext. DIP middle ext.	2.6 1.6
				dorsal interosseus VI	25	MCP middle flex. MCP middle radial add.	3.7 6.1
						PIP middle ext. DIP middle ext.	2.6 1.6
						MCP pinky flex. MCP pinky abd.	3.7 6.1
						PIP pinky ext. DIP pinky ext.	2.6 1.6

**Table 1:** List of the pseudo muscles of our system with their fibre resting length  $\ell_0$  [mm], the affected joints, and the lever arms  $||\vec{r}||$  [mm]. Abbreviations used in this table: superf. = superficialis, prof. = profundus, flex. = flexion, ext. = extension, abd. = abduction, add. = adduction, opp. = opposition.