Reconstructing Personalized Anatomical Models for Physics-based Body Animation

Petr Kadleček *
Charles University in Prague

Alexandru-Eugen Ichim *
EPFL

Tiantian Liu
University of Pennsylvania

Jaroslav Křivánek
Charles University in Prague

Ladislav Kavan
University of Utah

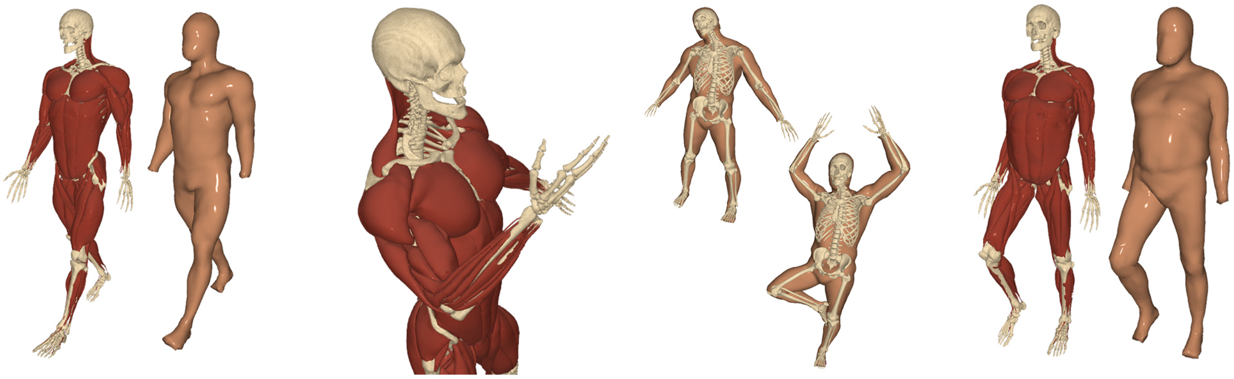


Figure 1: *We present a full-body reconstruction and animation system that can simulate physics-based volumetric effects such as self-collision and inertial effects. Our method uses a set of 3D surface scans to adapt an anatomically-inspired volumetric model to the user.*

Abstract

We present a method to create personalized anatomical models ready for physics-based animation, using only a set of 3D surface scans. We start by building a template anatomical model of an average male which supports deformations due to both 1) subject-specific variations: shapes and sizes of bones, muscles, and adipose tissues and 2) skeletal poses. Next, we capture a set of 3D scans of an actor in various poses. Our key contribution is formulating and solving a large-scale optimization problem where we compute both subject-specific and pose-dependent parameters such that our resulting anatomical model explains the captured 3D scans as closely as possible. Compared to data-driven body modeling techniques that focus only on the surface, our approach has the advantage of creating physics-based models, which provide realistic 3D geometry of the bones and muscles, and naturally supports effects such as inertia, gravity, and collisions according to Newtonian dynamics.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: 3D avatar creation, body animation, anatomical models, rigging

1 Introduction

The importance of human anatomy in visual arts was appreciated already by Renaissance masters such as Leonardo da Vinci. More recently, 3D anatomical models combined with physics-based simulation have been used to deliver unprecedented visual realism in modern computer generated movies. Unfortunately, the design of anatomically realistic characters is a labor intensive process even for experienced digital artists using professional modeling and simulation tools, such as those developed at Weta Digital and the ILM. Therefore, high-fidelity anatomical models are typically only affordable in high-budget production, e.g., in movies such as Avatar or The Lord of The Rings trilogy. Even though modeling of imaginary creatures such as dragons inherently relies on creativity of digital artists, when it comes to modeling humans, we believe we can substantially improve upon the state of the art.

In this paper we present an automatic method to create an anatomical, physics-based model of the body of a given human subject, e.g., an actor. We achieve this by capturing a set of full-body 3D scans in various poses and combining it with a template anatomical model. This template model represents the anatomy of an average human body, similar to traditional medical atlases. However, actual human bodies exhibit large variations in height, muscularity, adiposity, proportions of the limbs, etc. Our goal is to reshape and rescale the template anatomical model in order to fit the target scans as closely as possible, while accounting for shape changes due to both subject-specific variations (bone lengths, muscularity, adiposity, ...) as well as due to posing (changes of joint angles). The first type of deformations (subject-specific) are caused by long-term biological growth processes, while the pose-based deformations are induced by short-term voluntary muscle contractions and consequent joint motion. Our approach is summarized in Figure 2.

Data-driven modeling of animated human bodies has been a long standing topic in computer graphics. Systems such as SCAPE [Anguelov et al. 2005] or the more recent BlendSCAPE [Hirshberg et al. 2012] construct an articulated human body model from a set of input 3D scans. Similarly to artist-directed systems such as

[*] The first two authors contributed equally to this work.

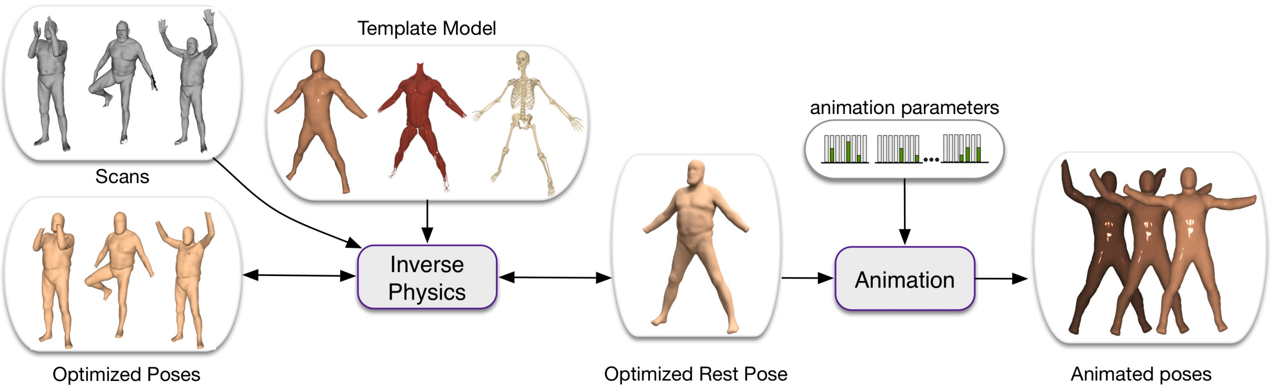


Figure 2: *Workflow of our method: We take as input a set of 3D scans of the same actor in different poses. Our method aims at reconstructing a complete volumetric, rigged, and physics-ready body model of the actor, by starting from an anatomical template model of an average male. This consists of extracting its exterior and interior shapes, as well as skeleton bone lengths. Finally, our models are ready to be animated using external skeletal and muscle activation data.*

Pose Space Deformation [Lewis et al. 2000], these methods build a data-driven model which predicts skin deformations based on the skeletal pose (i.e., joint rotations). However, these methods focus exclusively on the skin, i.e., outer boundary of the body. The skeleton is modeled as connected line segments, disregarding the volumetric nature of bones or muscles. While surface-based data-driven methods are effective at interpolating the input scans, they are oblivious to the fact that biological soft tissues are elastic solids subject to Newton's laws of motion. A notable exception is DYNA [Pons-Moll et al. 2015], which we will discuss in Related Work. To our knowledge, our method is the first to reconstruct a fully physics-based subject-specific anatomical model, naturally supporting effects such as inertia, collisions, and gravity. We found that already volumetric modeling of organs and their corresponding stiffness has interesting visual implications; e.g., the rigidity of the rib cage is clearly visible when animating upper trunk rotations, such as in Figure 8.

The problem of reconstructing anatomical models only from surface 3D scans is inherently ill-posed. Ground truth measurements of organs could be obtained using MRI or CT scans; however, these are expensive medical-grade devices designed to diagnose fine-scale pathologies such as bone fractures or tumors. Aside from the high costs, MRI or CT scanners are not suitable for computer animation purposes because they offer only a very limited workspace, i.e., the motion of the imaged human subject is highly constrained; furthermore, MRI machines require long scanning times and CT scanners expose the person to ionizing radiation. Fortunately, for computer graphics purposes we do not need high-fidelity medical imaging, because a rough estimate of the scale and shape of the bones, muscles, and subcutaneous adipose tissues is sufficient to produce high quality animations. Our anatomical model is designed for full-body animations and contains only the most visually significant muscles; we do not model the delicate muscles of the face, hands, and feet, as these body parts are often animated by specialized techniques. Our anatomical template also does not contain the nervous or circulatory systems or models of internal organs. However, our results can readily be combined with other computer graphics techniques such as displacement mapping in order to model, e.g., prominent veins or fine scale wrinkles.

By measuring only the 3D geometry of the skin, it seems impossible to determine what are the shapes and sizes of the underlying bones, muscles, and adipose tissues. However, bones and muscles do not grow arbitrarily in healthy human subjects (we do not consider pathologies in this work), because the musculoskeletal apparatus must be a functional mechanical system to allow locomotion. To quantify which shapes are more likely than others, we employ biomechanics-based growth models similar to Computational Bodybuilding [Saito et al. 2015]. While Computational Bodybuilding presented methods for the *forward* simulation of growth of bones, muscles, and adipose tissues, in this paper, we study the *inverse* problem, i.e., we formulate an optimization to recover the fitting parameters which best explain our input 3D scans. This problem is quite challenging because we have to account for 1) the fact that each 3D scan is in a different pose and 2) the organs do not grow independently, but influence each other due to internal action-reaction forces (when one bone/muscle grows, it pushes the adjacent organs).

Contributions. To our knowledge, the problem of reconstructing physics-based anatomical models from input 3D scans has not been tackled in previous work. Our main contribution is inverse body modeling (Section 5), i.e., formulating and solving a large optimization problem to find a subject-specific anatomical model which explains the input 3D scans as closely as possible. Most parts of our forward skinning model (Section 4) are derived from previous work; however, we devise a new elastic potential (which we call "symmetric as-rigid-as-possible" energy) in order to perform the subsequent inverse modeling, since classical as-rigid-as-possible models [Sorkine and Alexa 2007] do not work, as we discuss in Section 4. We hope that our approach will help to lower the costs of creating anatomical models of humans and make high-quality physics-based animation accessible not only to well-known VFX studios, but to a larger body of researchers and artists.

2 Related Work

Data-driven techniques. The most common approaches for modeling complex anatomical variation is by leveraging large amounts of data, usually in the form of 3D body scans or performance capture data. Anguelov et al. [2005] learn a statistical model for body shape variations as a function of body pose changes, which is applied on top of a statistical model of neutral-pose body shapes. As such, the same deformation model is used for all the people, while we have the advantage of constructing person-specific internal components which will behave differently in animations. This data-driven approach was extended and applied to sparse motion-capture animation by Loper et al. [2014], in order to obtain better quality motion reconstructions as compared to traditional skeleton-driven skinning approaches. Zuffi et al. [2015] propose a part-based model where each body component is a mesh associated to a statistical space, and connected together by pairwise stitching energies. Recently, Pons-Moll et al. [2015] introduced a data-driven technique that additionally encodes shape changes due to skin and limb velocity and acceleration, producing animations with compelling inertial effects without the need for a physics simulation. While these techniques are powerful interpolation tools, they are limited in their extrapolation capabilities, fixable only by collecting more and more data. In contrast, our method produces fully physics-based models, naturally supporting not only inertial effects, but also effects due to gravity, volumetric bones, and collisions.

For the particular task of breathing simulation, Tsoli et al. [2014] introduce a data-driven approach in which pose and shape variation is extracted from a set of registered 3D scans of people captured while breathing in different scripted ways. These priors are then used to generate varying types of respiration motions in novel characters. In our method we do not explain shape variations due to breathing, even though this would be an interesting direction for future work.

Anatomical models and physics. The motion of humans and interactions between the various anatomical elements have long been an important focus point for the biomechanics community. OpenSim [Delp et al. 2007] is an example of an open-source software framework for biomechanical modeling, simulation and analysis, extensively used in biomechanics and motor control science. However, OpenSIM does not support physics-based volumetric modeling of muscles or adipose tissues. There are also other specialized medically oriented frameworks, such as Sofa [Allard et al. 2007] or ArtiSynth [Lloyd et al. 2012].

The survey of Lee et al. [2010] offers a thorough overview of how the biomechanics and computer graphics communities model and simulate muscles, with most work being focused on skeletal muscles. Muscles are very complex structures that are not completely understood by modern medicine, and, as a result, various approximations have been proposed for making muscle simulation tractable for various medical or entertainment applications. Out of those, the physics-based and data-driven approaches are the ones of most interest for our work. Teran et al. [2003; 2005b; 2005a] introduced some of the first comprehensive approaches for biomechanical human body simulation in computer graphics. They construct a complete volumetric human body and a compatible FEM simulation by using solely data from the Visible Human Dataset.

Saito et al. [2015] propose a novel system for performing bodybuilding or weight loss simulations on human models. They model the muscles using synthetically computed muscle fibers. The growth of the muscles is discretized into the anisotropic stretch of individual muscle tetrahedra in the direction of the fibers, and computed efficiently using the projective dynamics solver [Bouaziz et al. 2014]. The key difference from our method is that Saito et al. [2015] require the bone/muscle/fat fitting parameters to be provided by the user.

Fan et al. [2014] propose a framework for simulating a dynamic volumetric musculoskeletal system using an Eulerian-on-Lagrangian discretization that can handle sliding elements in close contact, volume preservation and large deformations.

Anatomy Transfer [Dicko et al. 2013] is a method for transferring and editing the internal structure of human bodies. It uses a template human body model containing the skeleton and internal organs and registers it to a single surface-mesh humanoid model. The internal volume is adapted using harmonic deformation, driven by the registration of the exterior surface. The amount of fat tissue is controlled manually and the growth of the bones is constrained for more plausible results. In a similar vein, [Zhu et al. 2015] adapts the bone structure of upper and lower limbs given an RGB-D sequence of moving limbs. Comparisons and other differences are discussed in Section 7.

While a lot of research has gone into tackling the general problem of human body motion, there has been work targeting specific aspects. For example, Si et al. [2014] use an anatomical body model with muscle actuations in a complex fluid simulation in order to build a control system to simulate different styles of swimming. Similarly, Lee et al. [2006] focus on the biomechanical modeling and neuromuscular control of the neck region.

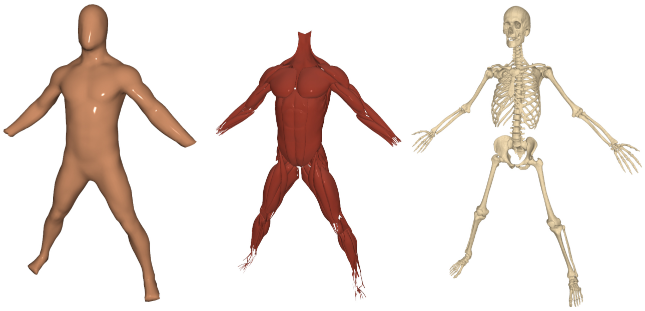


Figure 3: *Components of our anatomically-inspired volumetric template model. From left to right: skin and underlying generic soft tissue, muscles and tendons, skeleton.*

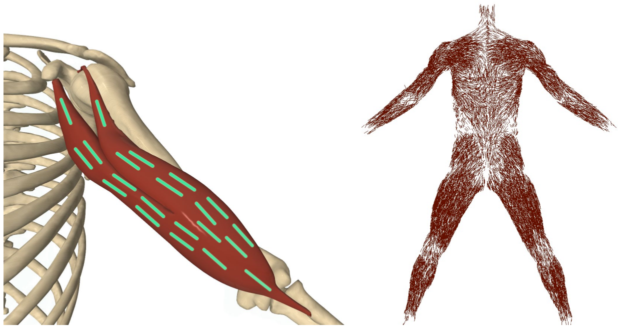


Figure 4: *Left: a close-up on the fibers on the right biceps muscle. Right: Visualization of the embedded muscle fibers in the template model.*

Combining simulation and data. A technique for modeling nonlinear material deformations from a set of captured examples is introduced by Bickel et al. [2009]. They used a scattered data interpolation technique in strain-space to simulate novel deformations of objects composed of the observed materials. Similarly, Wang et al. [2015] use off-the-shelf 3D sensors to track and model deformations of soft objects using physics-based probabilistic priors. Chen et al. [2014] propose a performant approach to reconstruct the zero-gravity rest pose shape of an object given multiple observations under various external forces such as gravity.

3 Template Body Model

The template model defines the topology of the fitted actors, and acts as a regularizer in the reconstruction process (see Fig. 3). It consists of a set of n vertices $\mathbf{X}^{\text{tmpl}} \in \mathbb{R}^{3n}$, connected in a tetrahedral mesh. We build the template similar in spirit to Saito et al. [2015] by starting from the commercially available Zygote anatomical model [Zygote 2016] with 111 muscles and 204 bones represented as meshes. The skin, muscles, and bones are uniformly remeshed with the Instant Meshes algorithm [Jakob et al. 2015] and then the surfaces are tetrahedralized using the approach of Jacobson et al. [2013].

In our work, we differentiate between four main types of materials: bones, tendons, muscles, and generic soft tissue. Each bone, tendon, and muscle is embedded into the template tetrahedral mesh in a nonconforming way; i.e., each tetrahedron might contain one or all of the materials in certain percentages. These percentages are computed as a pre-processing stage using a Monte Carlo sampling approach

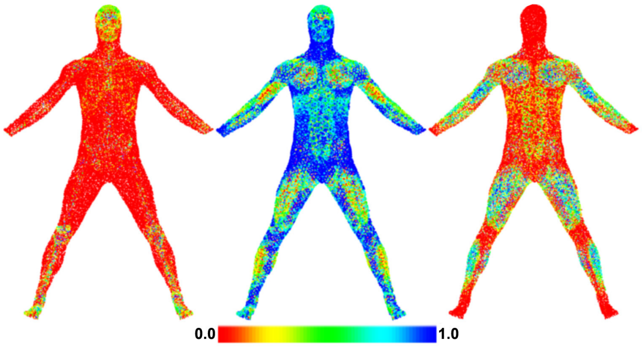


Figure 5: *The distribution of the material types inside the body. From left to right: bones, generic soft tissue, muscle.*

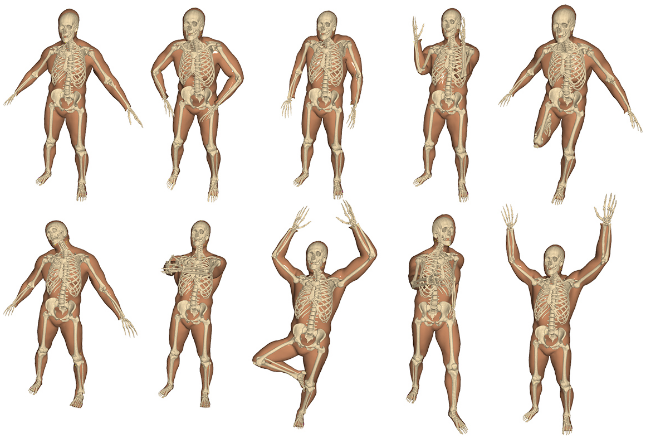


Figure 6: *Complex skeleton rig fitting on Faust dataset.*

to estimate the amount of overlap of each muscle/tendon/bone with each tetrahedron. For modeling the muscle atrophy and hypertrophy during subject-specific body fitting, as well as muscle activations during the animation stage (Section 6), the muscle fiber directions are required (see Figure 4). We compute the fiber directions in a similar way as Saito et al. [2015]. First, the tendon regions are selected manually and associated with Dirichlet boundary conditions. The non-tendon muscle boundaries are associated with Neumann boundary conditions. Next, we solve a Poisson equation for a scalar field using these boundary conditions. The resulting muscle fiber directions are aligned with gradients of this scalar field.

Our template anatomical model corresponds to a lean male. To be able to realistically model subjects with larger amounts of subcutaneous fat, we enhance our discretized volumetric template with a "muscle envelope," [Saito et al. 2015], i.e., a triangle mesh which wraps all of the muscles and separates them from the subcutaneous tissues. See Figure 5 for a visualization of the material distribution in the template model.

In addition to modeling soft tissue, we also use a realistic skeletal rig to parameterize the allowed motion of the bones. We built our rig using kinematic models established in biomechanics [Wu et al. 2002]. The final rig is sufficiently expressive to allow even for complex poses, as shown in Figure 6. Also, our rig describes not only pose-dependent variations (via the joint rotation angles $\boldsymbol{\theta}$), but also subject-specific variations (via scaling parameters $\boldsymbol{\pi}$). The scaling parameters $\boldsymbol{\pi}$ allow us to model different lengths and sizes of the bones between individuals. We shall denote $\text{Rig}(\boldsymbol{\theta}_i, \boldsymbol{\pi})$ as the function that describes the motion of the bones as a function of rig parameters. Specifically, the function $\text{Rig}(\boldsymbol{\theta}_i, \boldsymbol{\pi})$ returns posed (skinned) vertex samples, illustrated in Figure 9, in the current pose and scaling of the skeletal rig. These vertex samples will be used as boundary conditions for minimizing the elastic energies of the soft tissues, as described below.

4 Forward Skinning Model

Before diving into the *inverse* problem of body reconstruction, we first describe our *forward* physics-based character model. Our model is built by extending recent works, in particular Saito et al. [2015] and Zhu et al. [2015]. Saito et al. simulated growth only in the rest pose, without the use of a skeletal rig. Zhu et al. did create a skeletal rig, but only for the extremities (the arm and the leg) and the deformation model was based on direct skinning models.

In this paper, the body shape is implicitly defined as minimizer of a deformation energy (corresponding to elasticity of soft biological tissues) subject to Dirichlet boundary conditions (corresponding to the bones which are fixed in a given position in space). This process is known as *quasi-statics* [McAdams et al. 2011]: the bones are kinematically controlled, e.g., by an animator, and for each configuration of the bones, we compute a quasi-static equilibrium where the forces due to bone contacts cancel forces due to internal elasticity of the flesh (we use the term "flesh" as a shorthand for soft biological tissues). These two interpretations are equivalent because forces are negative derivatives of the elastic potential and therefore must be zero in a minimizer.

In equations, we can define the quasi-static solution as function:

$$\text{Skin}(\mathbf{X}^{\text{src}}, \boldsymbol{\theta}_i, \boldsymbol{\pi}) = \arg\min_{\mathbf{X}} E_{\text{skin}}(\mathbf{X}^{\text{src}}, \mathbf{X}, \boldsymbol{\theta}_i, \boldsymbol{\pi}), \tag{1}$$

where $E_{\text{skin}}(\mathbf{X}^{\text{src}}, \mathbf{X}, \boldsymbol{\theta}_i, \boldsymbol{\pi})$ is equal to the following sum:

$$\text{BoneFlesh}(\mathbf{X}, \boldsymbol{\theta}_i, \boldsymbol{\pi}) + E_{\text{def}}(\mathbf{X}^{\text{src}}, \mathbf{X}) + E_{\text{grav}}(\mathbf{X}) + E_{\text{col}}(\mathbf{X}). \tag{2}$$

Here $\boldsymbol{\theta}$ and $\boldsymbol{\pi}$ are joint orientations and bone scaling parameters as discussed in Section 3. The vector $\mathbf{X}^{\text{src}}$ describes positions of mesh vertices in a reference *rest pose*, while $\mathbf{X}$ corresponds to the *deformed pose*. The BoneFlesh function describes the connection between the deformable mesh representing the flesh and the fixed bones. $E_{\text{def}}(\mathbf{X}^{\text{src}}, \mathbf{X})$ is an elastic potential function which measures the amount of deformation between configurations $\mathbf{X}^{\text{src}}$ and $\mathbf{X}$ (both of which correspond to meshes with the same connectivity). $E_{\text{grav}}(\mathbf{X})$ is the gravity potential, i.e., a linear function which corresponds to the familiar mgh product (mass, gravity constant, height). The gravity potential allows us to simulate the interplay between inertial and gravity forces in a physically realistic way, which is important, e.g., in animating a fat man jumping. Finally, $E_{\text{col}}(\mathbf{X})$ is energy potential penalizing collisions, i.e., self-intersections of the mesh. The necessary condition for $\mathbf{X}$ being in quasi-static equilibrium is $\nabla_{\mathbf{X}} E_{\text{skin}} = 0$, i.e., sum of forces is zero. More details on the above mentioned terms follow.

BoneFlesh. The BoneFlesh term models coupling between kinematically controlled bones and physically simulated flesh. Anatomically, this term can be related to connective tissues which hold the musculoskeletal system together. Mathematically, we define:

$$\text{BoneFlesh}(\mathbf{X}, \boldsymbol{\theta}_i, \boldsymbol{\pi}) = w_{\text{bone}} \left\| \mathbf{S}^{\text{bone}}\mathbf{X} - \text{Rig}(\boldsymbol{\theta}_i, \boldsymbol{\pi}) \right\|^2, \tag{3}$$

where $\mathbf{S}^{\text{bone}}$ is a binary selector matrix which extracts vertices corresponding to the bone vertices kinematically controlled by the Rig function, see Figure 9. These vertices are chosen to approximately uniformly sample the surface of the bones and are explicitly present in the tet-mesh associated with $\mathbf{X}$ (conforming embedding). In theory, barycentric (non-conforming) embedding of bone vertices should be sufficient; however, we observed occasional numerical

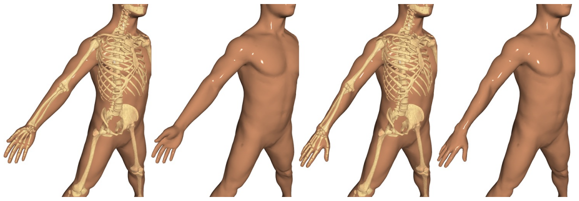


Figure 7: *Complex pronation-supination motion is handled well by our physics skinning.*

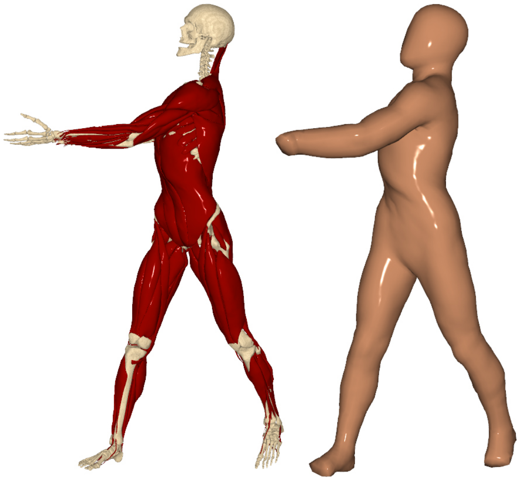


Figure 8: *Anatomically correct bones produce more realistic body shapes e.g. during upper trunk rotation, where the rib cage retains its shape.*

stability issues when nearly co-linear or co-planar vertex samples shared the same tetrahedron. Switching to conforming embedding of bone-samples successfully prevents these issues. For that we use TetGen with a switch to insert additional points [Si 2015]. The weighting w_{bone} controls the stiffness of the bone-flesh attachments and is chosen sufficiently high to avoid excessive sliding of the flesh (we note that some sliding is natural because biological connective tissues are compliant). This model is sufficient even for large deformations of the flesh such as pronation/supination (Figure 7) or upper trunk rotation (Figure 8).

Rig. Our kinematic skeleton is modeled by the function $\text{Rig}(\boldsymbol{\theta}, \boldsymbol{\pi})$, which takes joint angle orientations $\boldsymbol{\theta}$ and bone scaling parameters $\boldsymbol{\pi}$ as input, and produces world-space coordinates of vertices sampling the surfaces of the bones, as shown in Figure 9. The Rig function performs two main tasks: 1) it geometrically deforms the bones according to the scaling parameters, allowing us to model individuals with various lengths and shapes of the bones; 2) it implements standard forward kinematics, i.e., hierarchical composition of rotations of individual joints. We currently support only rotational joints, but more complicated joint types (e.g. spline joints [Lee and Terzopoulos 2008]) could be added to improve the accuracy of the kinematic modeling.

When changing the lengths and shapes of the bones, it is important not to distort the shape of the bone heads, because adjacent bone heads are often in close sliding contact. We achieve this in a similar way as Zhu et al. [2015]. Specifically, each bone is deformed using linear blend skinning with bounded biharmonic weights [Jacobson et al. 2011] with handles located in the center of each of the bone heads, see Figure 10 for an example of a long bone elongation. The handles of adjacent bones (i.e., forming a joint) are constrained to be transformed by the same matrix which contains only translation

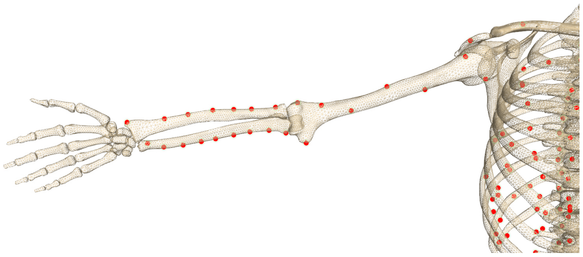


Figure 9: *Sampled bone vertices corresponding to the selector matrix* $\mathbf{S}^{bone}$ *used in BoneFlesh function.*

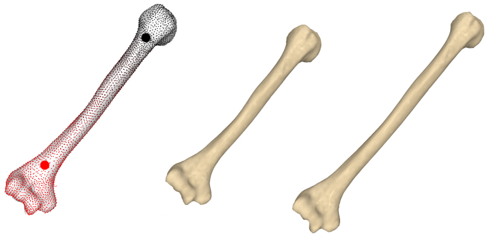


Figure 10: *Example of the humerus bone elongation preserving shape of bone heads using two deformation handles and precomputed bounded biharmonic weights.*

and uniform scale. This guarantees that the structure of the joint will be preserved. Bones with more complex shapes, such as the ribcage and spine, are treated in a similar fashion. We allow uniform scaling and elongation in the direction of the spinal cord to preserve vertebrae connections. Similarly, we allow elongation of the Scapula in the direction from the Sternum to the Humerus. For every bone, we choose an elongation direction, precompute weight for each sampled bone vertex and set bounds of the parameterization based on anatomical limits. We found that this parameterization was expressive enough to fit all tested scans well. Formally, we can express this deformation using the $\text{BoneFit}(\boldsymbol{\pi})$ function which depends only on the fitting parameters $\boldsymbol{\pi}$ and produces the modified rest pose bone vertex samples as a weighted linear combination of scaling and elongation transformations $\mathbf{T}^{\text{elng}}$:

$$\text{BoneFit}(\boldsymbol{\pi}) = \pi_{scale}\mathbf{S}^{\text{bone}}\mathbf{X}^{\text{src}} + \boldsymbol{\pi}_{elng}\mathbf{W}_{\text{BBW}}\mathbf{T}^{\text{elng}}\mathbf{S}^{\text{bone}}\mathbf{X}^{\text{src}} \quad (4)$$

where $\mathbf{W}_{\text{BBW}}$ are precomputed weights, π_{scale} is a global scale parameter and $\boldsymbol{\pi}_{elng}$ represents rest of the fit parameters.

The next step is standard forward kinematics, i.e., hierarchical composition of transformations which correspond to the rotations of individual joints (appearing as components of $\boldsymbol{\theta}$) and coordinate transformations between the individual joints. This is analogous to traditional forward kinematics models used in robotics [Murray et al. 1994], with the only difference that in our model, the lengths of the bones can change according to the $\boldsymbol{\pi}$ parameters. If we denote the resulting transformation from the rest pose to the world space as $\text{FK}(\boldsymbol{\theta}, \boldsymbol{\pi})$, the entire rig function can be written as composition:

$$\text{Rig}(\boldsymbol{\theta}, \boldsymbol{\pi}) = \text{FK}(\boldsymbol{\theta}, \boldsymbol{\pi})\text{BoneFit}(\boldsymbol{\pi}), \quad (5)$$

where we assume the FK function returns a stack of homogeneous matrices which are applied to each of the rescaled rest pose bone samples returned by BoneFit.

Elastic potential E_{def}. Elastic models of biological soft tissues have received considerable attention both in the biomechanics [Weiss et al. 1996; Fung 2013] as well as computer graphics communities [Teran et al. 2005a; Teran et al. 2005b; Sifakis et al. 2005; Lee et al. 2009; Sifakis and Barbic 2012]. Neohookean hyper-elastic materials have been found to function well in recent work [Bickel et al. 2012;

Skouras et al. 2014; Skouras et al. 2013]. Their advantage is realistic modeling of large compression – when an element degenerates, the Neohookean energy approaches infinity, as such configuration is not physically realistic. However, for applications in computer graphics, this behavior can be problematic, because as shown by Irving et al. [2004], inverted tetrahedra may be necessary to capture large deformations without resorting to remeshing. Increasing the mesh resolution can avoid these problems, but the resolution required to avoid all inversions would be prohibitively high; consider, e.g., the narrow space between cartilages of two bones connected by a joint. One possible solution is the popular corotated elastic model, which penalizes inverted elements by finite energies, i.e., allowing elements to invert if they are forced to do so. In the core of corotated elasticity is the following term: $\|\mathbf{D}_S\mathbf{D}_M^{-1} - \mathbf{R}\|_F^2$, where $\mathbf{D}_M$ and $\mathbf{D}_S$ are edge direction matrices in the material (i.e., reference) space and the deformed space (this notation is consistent with the tutorial of Sifakis and Barbic [2012]). The matrix $\mathbf{R} \in SO(3)$ is found by projecting $\mathbf{D}_S\mathbf{D}_M^{-1}$ onto the closest rotation.

Even though the classical corotated model is robust enough for use in a production environment [McAdams et al. 2011], it has a significant problem for our *inverse* problem, where we are optimizing also over the rest pose; i.e., in our setting, the matrices $\mathbf{D}_M$ are no longer constant. Unfortunately, we found that the inversion of the $\mathbf{D}_M$ matrices poses serious numerical problems when rest pose tetrahedra become close to degenerate; i.e., the $\mathbf{D}_M$ matrices become close to singular. This is problematic even if there is just a single degenerate tetrahedron present in the entire optimization.

To avoid these numerical difficulties, we use the following energy:

$$E_{\text{def}}(\mathbf{X}^{\text{src}}, \mathbf{X}) = \sum_i k_i \|\mathbf{D}_{S,i} - \mathbf{R}_i\mathbf{D}_{M,i}\|_F^2, \tag{6}$$

where the index i goes over all tets and $k_i \geq 0$ is stiffness of the i-th tet. Note that $\mathbf{D}_{M,i}$ depends linearly on $\mathbf{X}_i^{\text{src}}$, $\mathbf{D}_{S,i}$ depends linearly on $\mathbf{X}_i$ and $\mathbf{R}_i$ are rotation matrices minimizing the value of $E_{\text{def}}(\mathbf{X}^{\text{src}}, \mathbf{X})$. This optimal $\mathbf{R}_i$ can be computed by forming the signed SVD of $\mathbf{D}_{S,i}\mathbf{D}_{M,i}^{\mathsf{T}}$ and replacing the matrix of singular values with an identity matrix. We call this energy "symmetric as-rigid-as-possible" because $\|\mathbf{D}_{S,i} - \mathbf{R}_i\mathbf{D}_{M,i}\|_F = \|\mathbf{R}_i^{\mathsf{T}}\mathbf{D}_{S,i} - \mathbf{D}_{M,i}\|_F$, i.e., the rest pose and the deformed pose can be interchanged without changing the value of the energy. Perhaps more importantly, there is no need to invert the rest pose edge matrices $\mathbf{D}_{M,i}$, avoiding the numerical difficulties of the classical corotated model. Another advantage to the corotated model is that we do not need any volume weighting term such as $\frac{1}{6}|\det(\mathbf{D}_{M,i})|$ [Sifakis and Barbic 2012], because our units do not cancel as in the $\mathbf{D}_S\mathbf{D}_M^{-1}$ term; i.e., larger tets automatically contribute more to the total energy than smaller ones. See the appendix for more details.

The *stiffness* k_i of each tetrahedron is computed as a weighted average of materials overlapping this tetrahedron. Note that even though our tet-mesh conforms to bone sample vertices, it does *not* conform to the full polygonal boundaries of the bones or muscles (which would require prohibitively high-resolution tet-meshes). Similarly to Lee et al. [2009], we define the stiffness of each tetrahedron as $(\sum_t V_t k_t)/(\sum_t V_t)$, where t indexes individual material types (bones, tendons, muscles, generic soft tissues), $k_t > 0$ represents stiffness of each of the materials and V_t is the volume of a tetrahedron occupied by each component (bone, tendon, muscle, and generic soft tissues account for the remaining volume). We estimate V_t using Monte Carlo sampling (high accuracy is not necessary). See Section 9 for more details.

Handling Collisions We treat collisions in a fashion similar to McAdams et al. [2011]. We detect tet-tet collisions using a fast bounding box sequence intersection algorithm [Zomorodian and Edelsbrunner 2000]. For efficiency reasons, only selected regions near the joints are considered for collision processing, as these are the most common places where self-intersections occur. For example, our system does not try to detect or resolve pose-induced collisions such as hand touching the belly. The detected collisions are handled by instantiating temporary anisotropic springs that project the colliding vertices $\mathbf{X}$ out of the collision, to the surface of the tetrahedral mesh:

$$E_{\text{col}}(\mathbf{X}) = \left(\mathbf{n}_{\boldsymbol{\Pi}(\mathbf{X})}{}^{\mathsf{T}}(\mathbf{X} - \boldsymbol{\Pi}(\mathbf{X}))\right)^2, \tag{7}$$

where $\boldsymbol{\Pi}(\mathbf{X})$ is the projection of $\mathbf{X}$ onto the surface of the tetrahedral mesh, encoded by the barycenters of the closest surface triangle, and $\mathbf{n}_{\boldsymbol{\Pi}(\mathbf{X})}$ is the normal at the projected surface triangle. This anisotropy is helpful by allowing for sliding along the tangent plane at the projected surface point [McAdams et al. 2011]. The E_{col} energy potential is removed once the corresponding vertices are no longer in contact.

Muscle growth. Our symmetric as-rigid-as-possible (ARAP) elastic model can be extended to account for muscle growth [Saito et al. 2015]. We accomplish this by replacing E_{def} with the following energy for the tetrahedra containing muscles:

$$E_{\text{muscle}}(\mathbf{X}^{\text{src}}, \mathbf{X}, \boldsymbol{\alpha}) = \sum_i \|\mathbf{D}_{S,i} - \mathbf{R}_i\mathbf{B}_i\mathbf{S}(\alpha_i)\mathbf{B}_i^{\mathsf{T}}\mathbf{D}_{M,i}\|_F^2, \tag{8}$$

which differs from the symmetric ARAP model by the term $\mathbf{B}_i\mathbf{S}(\alpha_i)\mathbf{B}_i^{\mathsf{T}}$ accounting for muscle growth. Specifically, the orthonormal matrix $\mathbf{B}_i$ is a change of coordinates which transforms the x-axis to align with the fiber directions of the i^{th} tetrahedron (Figure 4). The matrix $\mathbf{S}(\alpha_i)$ is a scaling matrix in the y and z-axes, which allows for simulating muscle shape changes due to atrophy or hypertrophy:

$$\mathbf{S}(\alpha_i) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \alpha_i & 0 \\ 0 & 0 & \alpha_i \end{pmatrix}. \tag{9}$$

5 Inverse Body Modeling

The input of our algorithm is a set of scans corresponding to various poses of a given human subject (see Figure 2). First, the input scans are registered against the skin surface mesh of our template body model $\mathbf{X}^{\text{tmpl}}$, i.e., deforming $\mathbf{X}^{\text{tmpl}}$ until it is in close correspondence with the target scans. We use a non-rigid ICP procedure [Rusinkiewicz and Levoy 2001], explained in more detail in Section 5.1. We denote the resulting registered meshes as $\mathbf{T}_k$, where $k = 1 \ldots \text{numScans}$. The goal of inverse body modeling is to recover the subject-specific body shape in the rest pose $\mathbf{X}^{\text{pers}}$. Note that this configuration is devoid of the effects of gravity (as if in zero-gravity environment), because the gravity forces are added during the quasi-static solve in the forward skinning process (Eq. 1). In addition to determining $\mathbf{X}^{\text{pers}}$, we also have to solve for the bone fitting parameters $\boldsymbol{\pi}$ and joint angles $\boldsymbol{\theta}_k$, where k indexes individual poses, $k = 1 \ldots \text{numScans}$. The fitting parameters $\boldsymbol{\pi}$ are fixed for a given human being, but the joint angles $\boldsymbol{\theta}_k$ vary from pose to pose. We need to find the values of $\mathbf{X}^{\text{pers}}$, $\boldsymbol{\pi}$, and $\boldsymbol{\theta}_k$ such that the forward skinning function $\text{Skin}(\mathbf{X}^{\text{pers}}, \boldsymbol{\theta}_k, \boldsymbol{\pi})$ produces shapes as close as possible to $\mathbf{T}_k$. Because Skin is a complicated implicitly defined non-linear function, we introduce auxiliary variables $\mathbf{X}_k^{\text{arti}}$ for the personalized and articulated (posed) body shapes. When the inverse body modeling process is complete, we will have $\mathbf{X}_k^{\text{arti}} = \text{Skin}(\mathbf{X}^{\text{pers}}, \boldsymbol{\theta}_k, \boldsymbol{\pi})$; however, this equality does not have to hold in the intermediate steps of our optimization pipeline.

Targeting term. We formalize the requirement of $\mathbf{X}_k^{\text{arti}}$ aligning as closely as possible with $\mathbf{T}_k$ using the following "targeting term", which is the main objective of our optimization:

$$E_{\text{targ}}(\mathbf{X}_k^{\text{arti}}) = \sum_k \|\mathbf{N}_k^{\mathsf{T}}(\mathbf{S}^{\text{skin}}\mathbf{X}_k^{\text{arti}} - \mathbf{S}_k^{\text{corsp}}\mathbf{T}_k)\|^2, \qquad (10)$$

where $\mathbf{N}_k$ is a matrix of stacked scan normals, $\mathbf{S}^{\text{skin}}$ is a binary selector matrix of surface vertices, and $\mathbf{S}_k^{\text{corsp}}$ is a matrix of barycentric coordinates that allows us to depart from the initial registration in order to account for imperfections in the initial correspondences. This is also why we use this "point-to-plane" objective which allows for sliding of the skin vertices of $\mathbf{X}_k^{\text{arti}}$ along their corresponding tangent planes at $\mathbf{T}_k$. The matrix $\mathbf{S}_k^{\text{corsp}}$ is initialized to the identity (i.e., trusting the initial registration as described in Section 5.1) and after each iteration of the optimization process, we search for new correspondences. Specifically, for every skin vertex of $\mathbf{X}_k^{\text{arti}}$, we search for closest point of $\mathbf{T}_k$, rejecting pairs further than 5 cm away or with normals differing by more than 30 degrees [Rusinkiewicz and Levoy 2001].

Reconstruction. Inverse body modeling can be formulated as the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{X}^{\text{pers}},\mathbf{X}_k^{\text{arti}},\boldsymbol{\pi},\boldsymbol{\theta}_k} E_{\text{targ}}(\mathbf{X}_k^{\text{arti}}) + E_{\text{reg}}(\mathbf{X}^{\text{pers}},\boldsymbol{\pi}) \\ &\text{subject to } \nabla_{\mathbf{X}_k^{\text{arti}}} E_{\text{skin}}(\mathbf{X}^{\text{pers}},\mathbf{X}_k^{\text{arti}},\boldsymbol{\theta}_k,\boldsymbol{\pi}) = 0 \end{aligned} \qquad (11)$$

where $k = 1 \ldots \text{numScans}$ as before. The equality constraints require the posed shapes $\mathbf{X}_k^{\text{arti}}$ to be exactly in quasi-static equilibrium; however, these constraints will be relaxed during our numerical solution procedure described below.

But first, let us explain the regularization term $E_{\text{reg}}(\mathbf{X}^{\text{pers}},\boldsymbol{\pi})$. Reconstructing anatomical models from surface scans only is an ill-posed problem, because we lack direct measurements from the *inside* of the human body. Instead, we rely on anatomical priors to rule out unlikely or even unnatural anatomies. We use

$$\begin{aligned} E_{\text{reg}}(\mathbf{X}^{\text{pers}},\boldsymbol{\pi}) = \text{BoneFlesh}(\mathbf{X}^{\text{pers}},\boldsymbol{\theta}_0,\boldsymbol{\pi}) + E_{\text{def}}(\mathbf{X}^{\text{tmpl}},\mathbf{X}^{\text{pers}}) \\ + w_{\text{muscle}} E_{\text{muscle}}(\mathbf{X}^{\text{tmpl}},\mathbf{X}^{\text{pers}}). \end{aligned} \qquad (12)$$

Even though the sum of the BoneFlesh and E_{def} terms is reminiscent of the forward skinning function, here these terms have a somewhat different function: they serve to explain deformations between individual human subjects, as opposed to poses of a single individual. The $\boldsymbol{\theta}_0$ vector of joint angles corresponds to the rest pose and the term $\text{BoneFlesh}(\mathbf{X}^{\text{pers}},\boldsymbol{\theta}_0,\boldsymbol{\pi})$ requires the personalized rest pose $\mathbf{X}^{\text{pers}}$ to align with the skeleton grown according to skeletal fitting parameters $\boldsymbol{\pi}$. The $E_{\text{def}}(\mathbf{X}^{\text{tmpl}},\mathbf{X}^{\text{pers}})$ term states that the deformation between $\mathbf{X}^{\text{tmpl}}$ and $\mathbf{X}^{\text{pers}}$ should be minimized. In other words, the personalized mesh needs to stretch or shrink according to the resized skeleton, but the shape should not depart too much from the initial template.

Finally, the $E_{\text{muscle}}(\mathbf{X}^{\text{tmpl}},\mathbf{X}^{\text{pers}})$ term penalizes shape changes which cannot be explained by muscle growth (the $\boldsymbol{\alpha}$ parameters are free). Our approach tries to explain as much shape variation as possible with biologically-inspired muscle growth. After that, fat growth is applied in the subcutaneous layer to match the surface shape of the target. The assumption is that muscle growth gives rise to different shapes than fat growth. The parameter $w_{\text{muscle}} \geq 0$ controls our confidence in this assumption and can be tuned by the user or based on external measurements; e.g., it could be obtained by the assessment of body fat percentage done by measuring the skin fold thickness, or standard body mass index (BMI) approximations. This worked well for both the overweight and muscular man (Fig. 12 third and first row). The effect of constrained muscle growth is also clearly visible in Fig. 16. When the muscle modeling can no longer explain target shape variations, the thin subcutaneous fat layer of tetrahedra (called "muscle envelope") grows to match the target. Note that there is no gravitational potential acting on $\mathbf{X}^{\text{pers}}$; it only acts on the final articulated shapes $\mathbf{X}_k^{\text{arti}}$. In other words, our $\mathbf{X}^{\text{pers}}$ shape corresponds to the rest-pose body in a zero gravity environment [Chen et al. 2014].

Penalty method. Equation 11 represents a non-convex constrained optimization problem that can be written in a general form as $\min f(\mathbf{x})$ subject to $\mathbf{c}(\mathbf{x}) = 0$, where f is the objective and $\mathbf{c}$ a vector function of constraints. We solve this optimization problem by converting it into a sequence of unconstrained optimization problems using the penalty method [Nocedal and Wright 2006]. Each unconstrained subproblem has the following form: $\min f(\mathbf{x}) + \gamma\|\mathbf{c}(\mathbf{x})\|^2$, where γ is the penalty weight. The γ parameter is progressively increased from 0 to γ_{max}.

Each γ-subproblem is solved using Newton's method with Hessian modification (Algorithm 3.2 in [Nocedal and Wright 2006]). In particular, evaluating the exact Hessian matrix would be complicated because it contains third derivative terms (note that the constraints $\mathbf{c}$ already contain first derivatives of the E_{skin} potential). Similarly to Bickel et al. [2012], we drop these third derivative terms. The approximate Hessian is further modified by adding scalar multiple of the identity matrix to ensure positive definiteness. Having determined the descent direction, we calculate appropriate step size using backtracking line search. We note that alternative numerical solution procedures are possible, e.g., the Augmented Lagrangian Method, however, we found that our quasi-Newton penalty method converges rapidly in our experiments.

5.1 Registration

In this section we describe our method to obtain the initial registration between our template model $\mathbf{X}^{\text{tmpl}}$ and the input scans $\tilde{\mathbf{T}}_1, \ldots, \tilde{\mathbf{T}}_{\text{numScans}}$, which are unstructured triangle meshes with noise, holes, or other imperfections. We use a non-rigid ICP procedure which deforms $\mathbf{X}^{\text{tmpl}}$ into $\mathbf{T}_1, \ldots, \mathbf{T}_{\text{numScans}}$ such that each $\mathbf{T}_k$ is well aligned with its corresponding scan $\tilde{\mathbf{T}}_k$. We initialize the process with approximately 15 landmark points, interactively selected by the user in our GUI. We use the tet-mesh associated with $\mathbf{X}^{\text{tmpl}}$ to define a regularization energy for non-rigid ICP. Specifically, we use our symmetric ARAP energy (Eq. 6) with uniform stiffness k_i for all tets. We do not even account for the rigidity of the bones, i.e., we treat the entire template tet-mesh as a jellyfish. This approximation is sufficient to establish good initial correspondences, which will be refined in subsequent iterations of our optimization process.

5.2 Reconstruction pipeline

Since the inverse problem formulated in Equation 11 is highly nonlinear and the search space is large (e.g. up to 200k variables for our experiments), it can be challenging to find a correct solution and avoid numerical difficulties. Therefore, we split the optimization into several sub-problems to obtain well-defined initial values that approximate the solution. We found this step important to speedup the convergence and avoid local minima due to the non-linearity of some terms.

After registration, we start the reconstruction process by approximating skeleton parameters. We first optimize for global translation, rotation and scaling of the skeleton for each scan. Next, we enable the articulation parameters $\boldsymbol{\theta}$ to fit scan poses. After that, we optimize the rest of the skeleton variables including $\boldsymbol{\pi}$. This concludes the initial skeleton fitting and we move onto a next phase, which we call "forward initialization". In this phase we disable minimization

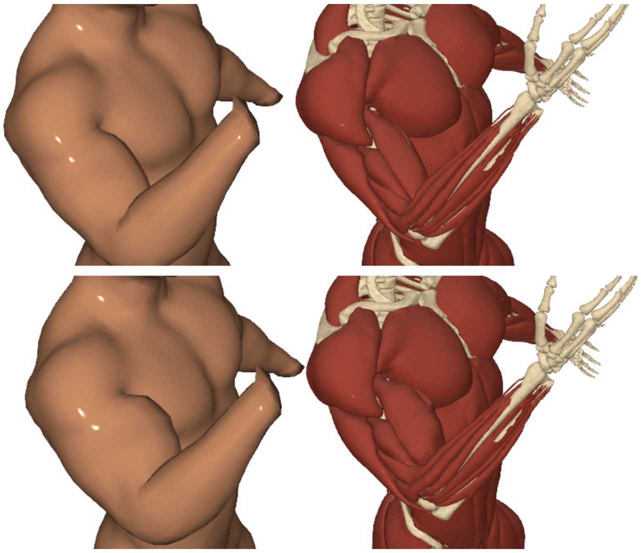


Figure 11: *Our physics-based animation approach allows for animating pose-specific muscle shape changes due to muscle contractions. The upper images show the shape of the arm and muscles in a flexing pose, and the lower images show the effect of contracting the biceps muscle in the same pose.*

over the $\mathbf{X}^{\text{pers}}$ variable in Eq. 11 to fit the scans without changing the original rest pose. Finally, we optimize over all variables (including $\mathbf{X}^{\text{pers}}$), taking advantage of good initial estimates computed in the previous steps. Our symmetric ARAP energy is particularly important to avoid numerical difficulties in evaluating the E_{def} and E_{muscle} terms. More details about the symmetric ARAP behavior can be found in the appendix.

6 Animation

The resulting personalized body model $(\mathbf{X}^{\text{pers}}, \boldsymbol{\pi})$ is ready for physics-based animation. As input, we provide a time-varying sequence of joint angles $\boldsymbol{\theta}_j$, where the index j samples discrete time intervals (corresponding, e.g., to a constant time step such as $1/30s$). The animated joint angles can come from various sources such as keyframe animation or from retargeted motion capture data. This is particularly easy to achieve by using a subset of the functionality of our optimization framework.

But first, let us explain how to introduce dynamic effects, such as flesh jiggling. In our physics-based framework, this can be naturally achieved by switching from quasi-statics to full dynamics simulation. Assuming the widely used Implicit Euler time integration, this is as simple as adding an extra convex quadratic term to the energy terms in the $E_{\text{skin}}(\mathbf{X}^{\text{src}}, \mathbf{X}, \boldsymbol{\theta}_k, \boldsymbol{\pi})$ function (Eq. 4). This "inertial" term introduces history dependence, i.e., accounts for Newton's first law (which is ignored in quasi-statics). Specifically, let us denote the animated body shape as $\mathbf{X}_j^{\text{anim}}$, where j again indexes discrete time steps. We assume that $\mathbf{X}_0^{\text{anim}}$ and $\mathbf{X}_1^{\text{anim}}$ are provided as initial conditions (typically starting with zero velocities, i.e., $\mathbf{X}_0^{\text{anim}} = \mathbf{X}_1^{\text{anim}}$). The inertial term can be defined as:

$$E_{\text{inert}}(\mathbf{X}) = \frac{1}{2h^2} \|\mathbf{M}^{1/2}(\mathbf{X} - 2\mathbf{X}_j^{\text{anim}} + \mathbf{X}_{j-1}^{\text{anim}})\|^2 \tag{13}$$

where $\mathbf{M}$ is a diagonal mass matrix and h is the time step. This term can be derived from the Implicit Euler integration rules, which can be found e.g. in [Bouaziz et al. 2014].

In addition to the inertial term, we also add the collision avoidance potential E_{col} discussed in Section 4. Gravity potential is also accounted for as described already in Eq. 4.

The physics-based animation framework is quite versatile and in addition to supporting the effects of inertia, collisions, and gravity, we can also add muscle contraction forces. We assume that time-varying muscle activation signals are provided by the user. These can be e.g., keyframed, which is common in professional VFX animation systems [WETA digital 2013], or calculated using inverse dynamics models [Lee et al. 2009]. Let us denote the muscle activation signals as $\boldsymbol{\beta}_j$, where j indexes discrete time steps as before. The muscle contraction potential is similar to the muscle growth potential (Eq. 8), however, instead of the rest-pose growth matrix $S(\alpha_i)$ for each tetrahedron i (Eq. 9) we use the following matrix:

$$\mathbf{S}(\beta_{i,j}) = \begin{pmatrix} \beta_{i,j}^{-1} & 0 & 0 \\ 0 & \sqrt{\beta_{i,j}} & 0 \\ 0 & 0 & \sqrt{\beta_{i,j}} \end{pmatrix} \tag{14}$$

which accounts for the volume preserving nature of muscle contraction due to high water content in soft biological tissues [Weiss et al. 1996]. Mathematically, this is modeled by the fact that the determinant of matrix $\mathbf{S}(\beta_{i,j})$ is one, resulting in the characteristic bulging behavior of contracting muscles (see Figure 11 for an example). Note that the muscle growth scaling matrix $\mathbf{S}(\alpha_i)$ (Eq. 9) does not have determinant one because it accounts for *growth*, which is of course not volume conserving.

7 Results

We performed our experiments on 3D surface scans with diverse quality and resolution. Specifically, we tested our reconstructions on publicly available good quality 3D surface scans obtained from the FAUST dataset [Bogo et al. 2014] and database of Hasler et al. [2009], and on high quality commercially available scans [TEN 24 2016]. Additionally, we also experimented with low resolution scans captured using the Microsoft Kinect with the Skanect Pro registration software.

Reconstruction accuracy. We have successfully reconstructed targets with various body types and skeletal variations including a muscular bodybuilder, subjects with apparent subcutaneous fat, as well as a slim actor, see Figure 12. We used between 2 to 5 scans for each subject depending on the quality of scans and diversity of the poses. Although it would be possible to use only a single scan in our method (similarly to Dicko et al. [2013]), this would mean the underlying anatomical model would be less well determined. In particular, we observed ambiguities when optimizing for subject specific variations in bone lengths. For example, given one 3D surface scan with the actor with straight limbs, it is very difficult to accurately determine the locations of the joints. Jointly optimizing over scans of multiple poses, e.g. adding a scan with bent limbs, helps to eliminate this uncertainty, as the optimization algorithm places the joint in the most appropriate location. In Figure 12 we demonstrate the accuracy of our approach in terms of matching the input 3D scans. Our results show that our physics-based model can reproduce high quality body shapes with a close visual similarity to the scans. Moreover, in Figure 17 we show how the fitting accuracy improves when increasing the number of input scans.

Gravitational effects. Another advantage of using multiple scans is reducing the ambiguity due to gravitational effects and self-collisions of the skin. In Figure 13 we show the effect of taking gravity into consideration during our inverse body modeling process. We aim to reconstruct the rest pose in zero gravity, because gravity will be added in the forward simulation process. Note that this is a challenging problem in its own right [Chen et al. 2014].

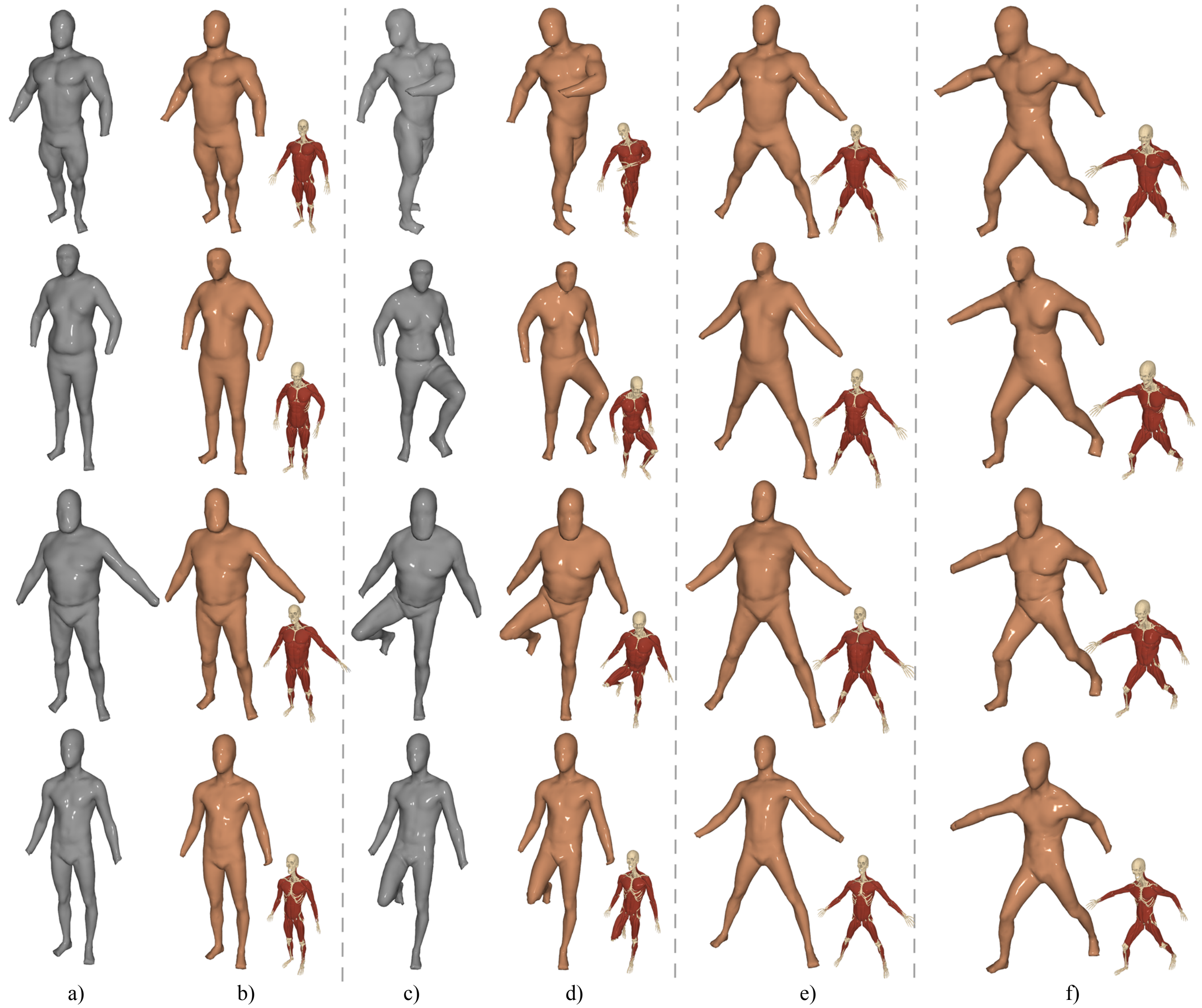


Figure 12: *Registered 3D surface scans of our test subjects in two different poses (a, c) and corresponding reconstructions using our anatomical physics-based model (b, d). Note that the shapes are quite similar. We also show our optimized rest pose* $\mathbf{X}^{pers}$ *(e) and a novel, unseen pose synthesized using our forward skinning model (f).*

Collisions. An example of collision handling during the forward animation phase is shown in Figure 14. Equally important is collision handling during inverse body modeling. When the input 3D scan contains body parts in contact, it means the measured shape was influenced by action-reaction forces preventing the flesh from inter-penetrating. Our E_{col} term estimates these contact forces and compensates for them during our inverse body modeling process. This results in recovering more accurate rest poses, as shown in Figure 15.

Comparison to Anatomy Transfer. Our approach has several key advantages over Anatomy Transfer [Dicko et al. 2013] and its extensions [Zhu et al. 2015]. First, our approach can take advantage of multiple scans in different poses, which leads to high reconstruction accuracy, as discussed above and shown in Figure 17. Second, Anatomy Transfer as well as its extensions [Zhu et al. 2015] use only an approximate deformation model of biological soft tissues. In our method, we use more realistic growth models for the bones and muscles, which allows us to estimate the underlying anatomy more accurately. Specifically, our material-aware deformation can reconstruct subcutaneous fat and the shape of skeletal muscles more realistically as shown in Figure 16. Furthermore, our reconstruction process is fully automatic, without needing artistic input. This is particularly important for animations including inertial effects and secondary motions of soft tissues. Our method is also able to reconstruct the whole skeleton while preserving realistic bone shapes, while Anatomy Transfer deforms skeleton using general affine transformations which results in non-realistic bone deformations.

8 Limitations and Future Work

We focus on capturing the physics of large- and medium-scale anatomical details, but we do not reconstruct faces, hands or toes. We believe that these are research topics on their own which require specialized approaches. However, such techniques already exist and could be integrated in our body modeling framework.

In the visualizations of our experiments we noticed that the bones sometimes protrude through the muscles, which is most visible in the chest region. This is due to the soft non-conformal embedding of the bones in the tetrahedral mesh of the body, as well as due to the multi-material property of each body tetrahedron. These problems could be

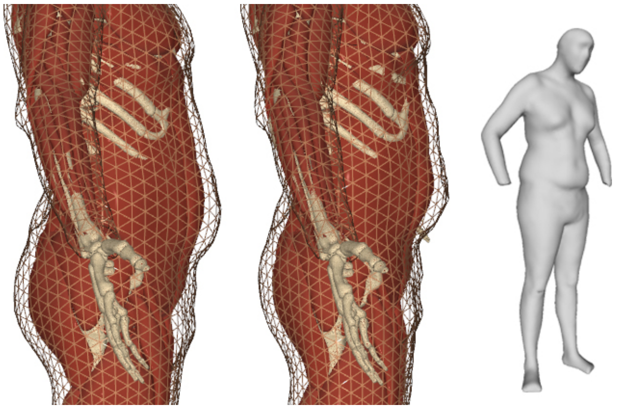


Figure 13: *Example of the effect of gravity on the rest pose reconstruction process. The figure on the left shows the result of the reconstruction without taking gravity into account. In the middle, gravity is taken into account and eliminated from the rest pose – note the belly "floating" as if the body was submersed in water. This "zero gravity" rest pose matches the input scan (right) closely because gravity is added during the forward simulation process.*

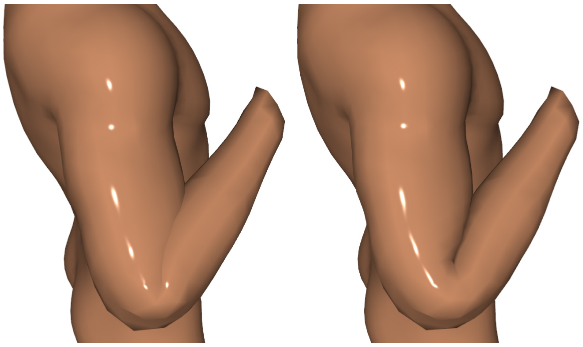


Figure 14: *Forward simulation collision handling example.*

alleviated by increasing the resolution of the template model, which may lead to the necessity of applying more memory-efficient and performant optimization techniques. Another solution is to use hard constraints for the BoneFlesh term which would require a different class of algorithms to solve the constrained optimization problem, e.g. Augmented Lagrangian Method as discussed in Section 5.

We do not consider muscle shape changes in the posed scans, assuming all the muscles are in a relaxed stage or that they are not contracted significantly. While this holds true for most of the scans we used in our experiment, one can think of poses and situations in which correctly capturing the shape variation of muscles due to contractions becomes important. For example, using a scan of the bodybuilder flexing his arm muscles together with scans in which he was relaxed created issues in our optimization. However, once reconstructed, our anatomical models allow for simulating muscle contraction in the forward animation stage. A venue of future research would be to automatically extract muscle activations given the pose of the subject, and to normalize the shape changes due to contractions in the rest pose reconstruction problem.

The scans used in our experiments are static poses, in which the actor was in equilibrium. The reconstruction problem becomes much more complex when dynamics is added to the scans, e.g., by capturing a continuous stream of point clouds from an actor's performance. Our algorithms do not make any male-specific assumptions (we even removed the genitals) and should therefore work for females. The only issue is preparation of a template female body which would require non-trivial 3D modeling efforts. Stylized or imaginary characters such as Incredible Hulk are beyond the scope of this project, because we cannot capture their ground truth scans and also, their musculoskeletal structure may differ from human anatomy. This opens opportunities for another research project with emphasis on user interaction and anatomical modeling.

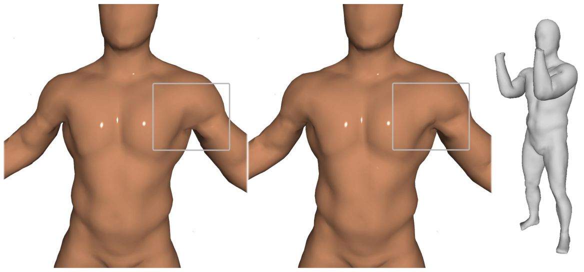
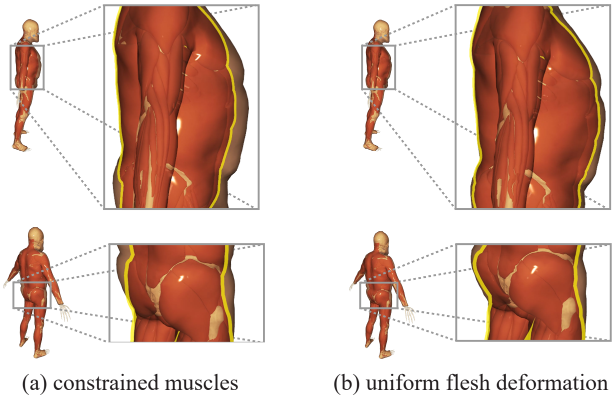


Figure 15: *Example of collision handling during inverse body modeling. In this example, a single scan was used (shown in gray) in which the actor was pressing his arms against his body. Notice that the rest pose reconstruction on the left has the shape of the arm imprinted on the chest; the reconstruction on the right does take the collision forces into account and reaches a more realistic rest shape.*



(a) constrained muscles

(b) uniform flesh deformation

Figure 16: *Example of material-aware deformations during inverse body modeling. In this comparison, muscle modeling is constrained by muscle fiber directions (a). The yellow outline shows that muscles are modeled more accurately and do not extend into regions that clearly should be fat in contrast to simple uniform flesh deformation used in (b). Material modeling is important for realistic simulation of inertial effects of soft tissues, such as subcutaneous fat.*

9 Conclusion

We presented an automatic method to reconstruct an anatomical, physics-based model of the body of a given human subject. To our knowledge, our system is the first to reconstruct personalized fully volumetric physics-based human body models, which are suitable for computer animation including effects such as inertia, collisions, and gravity. We believe that our method will serve as a tool for reducing the costs of person-specific modeling, and may inspire applications even beyond the traditional realms of computer graphics.

Acknowledgments. Our special thanks belong to Mark Pauly for the useful discussions and ideas, as well as Sanchit Garg for helping with the skeleton rig modeling. We also thank the anonymous reviewers for their valuable comments. This research was supported by NSF awards IIS-1617172, IIS-1622360, the grant SVV 2016-260332 and a gift from Activision. We thank TEN24 and their 3D Scan Store for the sets of detailed 3D scans used in our experiments.

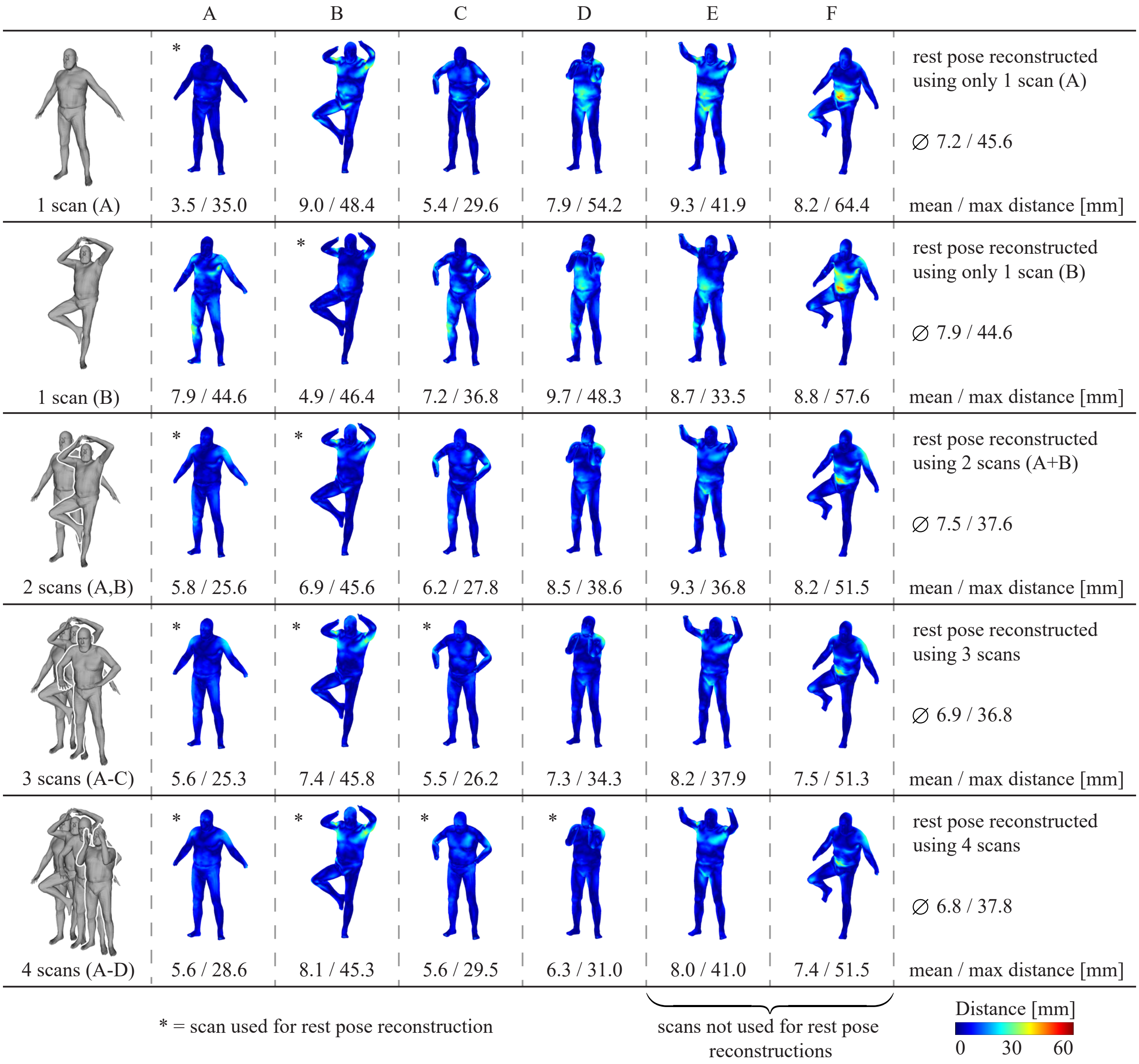


Figure 17: *Evaluation experiment showing the effects of using different input scans (rows 1 and 2), as well as varying the number of input scans (rows 3, 4, and 5) on the rest pose reconstruction process. The mean and maximum point-to-plane distances between the input scans and our fits (columns 2, 3, 4, 5 marked with *), as well as between our posed reconstr. and not-seen-before scans (columns 6 and 7) are shown.*

References

ALLARD, J., COTIN, S., FAURE, F., BENSOUSSAN, P.-J., POYER, F., DURIEZ, C., DELINGETTE, H., AND GRISONI, L. 2007. Sofa-an open source framework for medical simulation. In *MMVR 15-Medicine Meets Virtual Reality*, vol. 125, IOP Press, 13–18.

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., RODGERS, J., AND DAVIS, J. 2005. Scape: shape completion and animation of people. In *ACM Trans. Graph.*, vol. 24, 408–416.

BICKEL, B., BÄCHER, M., OTADUY, M. A., MATUSIK, W., PFISTER, H., AND GROSS, M. 2009. Capture and modeling of non-linear heterogeneous soft tissue. *ACM Trans. Graph. 28*, 3, 89.

BICKEL, B., KAUFMANN, P., SKOURAS, M., THOMASZEWSKI, B., BRADLEY, D., THE, T., JACKSON, P., MARSCHNER, S., MATUSIK, W., AND GROSS, M. 2012. Physical face cloning. *ACM Trans. Graph. 31*, 4, 118.

BOGO, F., ROMERO, J., LOPER, M., AND BLACK, M. J. 2014. FAUST: Dataset and evaluation for 3D mesh registration. In *Computer Vision and Pattern Recognition*.

BOUAZIZ, S., MARTIN, S., LIU, T., KAVAN, L., AND PAULY, M. 2014. Projective dynamics: fusing constraint projections for fast simulation. *ACM Trans. Graph. 33*, 4, 154.

Chen, X., Zheng, C., Xu, W., and Zhou, K. 2014. An asymptotic numerical method for inverse elastic shape design. *ACM Trans. Graph. 33*, 4, 95.

Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman, E., and Thelen, D. G. 2007. Opensim: open-source software to create and analyze dynamic simulations of movement. *Biomedical Engineering, IEEE Transactions on 54*, 11, 1940–1950.

Dicko, A.-H., Liu, T., Gilles, B., Kavan, L., Faure, F., Palombi, O., and Cani, M.-P. 2013. Anatomy transfer. *ACM Trans. Graph. 32*, 6, 188.

Fan, Y., Litven, J., and Pai, D. K. 2014. Active volumetric musculoskeletal systems. *ACM Trans. Graph. 33*, 4, 152.

Fung, Y.-c. 2013. *Biomechanics: mechanical properties of living tissues*. Springer Science & Business Media.

Guennebaud, G., Jacob, B., et al., 2010. Eigen v3. http://eigen.tuxfamily.org.

Hasler, N., Stoll, C., Sunkel, M., Rosenhahn, B., and Seidel, H.-P. 2009. A statistical model of human pose and body shape. In *Comp. Graph. Forum*, vol. 28, Wiley Online Library, 337–346.

Hirshberg, D. A., Loper, M., Rachlin, E., and Black, M. J. 2012. Coregistration: Simultaneous alignment and modeling of articulated 3d shape. In *Computer Vision–ECCV 2012*. Springer, 242–255.

Irving, G., Teran, J., and Fedkiw, R. 2004. Invertible finite elements for robust simulation of large deformation. In *Proc. of the EG/SIGGRAPH Symposium on Comp. Anim.*, 131–140.

Jacobson, A., Baran, I., Popovic, J., and Sorkine, O. 2011. Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph. 30*, 4, 78.

Jacobson, A., Kavan, L., and Sorkine-Hornung, O. 2013. Robust inside-outside segmentation using generalized winding numbers. *ACM Trans. Graph. 32*, 4, 33.

Jakob, W., Tarini, M., Panozzo, D., and Sorkine-Hornung, O. 2015. Instant field-aligned meshes. *ACM Trans. Graph. 34*, 6, 189:1–189:15.

Lee, S.-H., and Terzopoulos, D. 2006. Heads up!: biomechanical modeling and neuromuscular control of the neck. In *ACM Trans. Graph.*, vol. 25, 1188–1198.

Lee, S.-H., and Terzopoulos, D. 2008. Spline joints for multibody dynamics. In *ACM Trans. Graph.*, vol. 27, 22.

Lee, S.-H., Sifakis, E., and Terzopoulos, D. 2009. Comprehensive biomechanical modeling and simulation of the upper body. *ACM Trans. Graph. 28*, 4, 99.

Lee, D., Glueck, M., Khan, A., Fiume, E., and Jackson, K. 2010. A survey of modeling and simulation of skeletal muscle. *ACM Trans. Graph. 28*, 4, 1–13.

Lewis, J. P., Cordner, M., and Fong, N. 2000. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proc. of the 27th annual conf. on Comp. graph. and inter. tech.*, ACM Press/Addison-Wesley Publishing Co., 165–172.

Lloyd, J. E., Stavness, I., and Fels, S. 2012. Artisynth: A fast interactive biomechanical modeling toolkit combining multibody and finite element simulation. In *Soft tissue biomechanical modeling for computer assisted surgery*. Springer, 355–394.

Loper, M., Mahmood, N., and Black, M. J. 2014. Mosh: Motion and shape capture from sparse markers. *ACM Trans. Graph. 33*, 6, 220.

McAdams, A., Zhu, Y., Selle, A., Empey, M., Tamstorf, R., Teran, J., and Sifakis, E. 2011. Efficient elasticity for character skinning with contact and collisions. In *ACM Trans. Graph.*, vol. 30, 37.

Muja, M., and Lowe, D. G. 2014. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Trans. on Pattern Analysis and Machine Intelligence 36*.

Murray, R. M., Li, Z., Sastry, S. S., and Sastry, S. S. 1994. *A mathematical introduction to robotic manipulation*. CRC press.

Nocedal, J., and Wright, S. 2006. *Numerical optimization*. Springer Science & Business Media.

Pons-Moll, G., Romero, J., Mahmood, N., and Black, M. J. 2015. Dyna: A model of dynamic human shape in motion. *ACM Trans. Graph. 34*, 4, 120.

Rusinkiewicz, S., and Levoy, M. 2001. Efficient variants of the icp algorithm. In *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*, IEEE, 145–152.

Saito, S., Zhou, Z.-Y., and Kavan, L. 2015. Computational bodybuilding: Anatomically-based modeling of human bodies. *ACM Trans. Graph. 34*, 4.

Si, W., Lee, S.-H., Sifakis, E., and Terzopoulos, D. 2014. Realistic biomechanical simulation and control of human swimming. *ACM Trans. Graph. 34*, 1, 10.

Si, H. 2015. Tetgen, a delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software (TOMS) 41*, 2, 11.

Sifakis, E., and Barbic, J. 2012. Fem simulation of 3d deformable solids: a practitioner's guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses*, 20.

Sifakis, E., Neverov, I., and Fedkiw, R. 2005. Automatic determination of facial muscle activations from sparse motion capture marker data. In *ACM Trans. Graph.*, vol. 24, 417–425.

Skouras, M., Thomaszewski, B., Coros, S., Bickel, B., and Gross, M. 2013. Computational design of actuated deformable characters. *ACM Trans. Graph. 32*, 4, 82.

Skouras, M., Thomaszewski, B., Kaufmann, P., Garg, A., Bickel, B., Grinspun, E., and Gross, M. 2014. Designing inflatable structures. *ACM Trans. Graph. 33*, 4, 63.

Sorkine, O., and Alexa, M. 2007. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, vol. 4.

TEN 24, 2016. 3d scan store. http://www.3dscanstore.com.

Teran, J., Blemker, S., Hing, V., and Fedkiw, R. 2003. Finite volume methods for the simulation of skeletal muscle. In *Proc. of the EG/SIGGRAPH Symposium on Comp. Anim.*, Eurographics Association, 68–74.

Teran, J., Sifakis, E., Blemker, S. S., Ng-Thow-Hing, V., Lau, C., and Fedkiw, R. 2005. Creating and simulating skeletal muscle from the visible human data set. *Visualization and Computer Graphics, IEEE Transactions on 11*, 3, 317–328.

TERAN, J., SIFAKIS, E., IRVING, G., AND FEDKIW, R. 2005. Robust quasistatic finite elements and flesh simulation. In *Proc. of the EG/SIGGRAPH Symposium on Comp. Anim.*, ACM, 181–190.

THE CGAL PROJECT. 2016. *CGAL User and Reference Manual*, 4.8 ed. CGAL Editorial Board.

TSOLI, A., MAHMOOD, N., AND BLACK, M. J. 2014. Breathing life into shape: capturing, modeling and animating 3d human breathing. *ACM Trans. Graph. 33*, 4, 52.

WANG, B., WU, L., YIN, K., ASCHER, U., LIU, L., AND HUANG, H. 2015. Deformation capture and modeling of soft objects. *ACM Trans. Graph. 34*, 4, 94.

WEISS, J. A., MAKER, B. N., AND GOVINDJEE, S. 1996. Finite element implementation of incompressible, transversely isotropic hyperelasticity. *Computer methods in applied mechanics and engineering 135*, 1, 107–128.

WETA DIGITAL, 2013. Tissue system. http://www.fxguide.com/fxguidetv/fxguidetv-166-weta-digitals-tissue-system/.

WU, G., SIEGLER, S., ALLARD, P., KIRTLEY, C., LEARDINI, A., ROSENBAUM, D., WHITTLE, M., D DLIMA, D., CRISTOFOLINI, L., WITTE, H., ET AL. 2002. Isb recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motionpart i: ankle, hip, and spine. *Journal of biomechanics 35*, 4, 543–548.

ZHU, L., HU, X., AND KAVAN, L. 2015. Adaptable anatomical models for realistic bone motion reconstruction. *Comp. Graph. Forum 34*, 2.

ZOMORODIAN, A., AND EDELSBRUNNER, H. 2000. Fast software for box intersections. In *Computer Graphics Forum (Proc. of the EG/SIGGRAPH Symposium on Geom. Proc.)*, ACM, 129–138.

ZUFFI, S., AND BLACK, M. J. 2015. The stitched puppet: A graphical model of 3d human shape and pose. In *Computer Vision and Pattern Recognition*, 3537–3546.

ZYGOTE, 2016. Zygote body. [Online; accessed 9-May-2016].

Appendix

Implementation details. The geometric search data structures and algorithms used for the scan registration and collision detection are based on CGAL [The CGAL Project 2016] and nanoflann [Muja and Lowe 2014]. Numerical linear algebra is implemented using Eigen [Guennebaud et al. 2010]. We benchmarked the performance on a consumer laptop with a 3.1 GHz Intel Core i7 processor and 32GB of main memory. For a complete rest pose optimization using 4 scans, we needed a total number of about 15 Newton iterations until convergence, with about 120s of computation time per iteration. The template model used for the results presented in this paper has 12977 vertices, out of which 4901 are surface vertices, and 64164 tetrahedra. The skeleton used for rigging has 67 joints with a total of 52 articulation and 38 sizing parameters. There are 111 muscles in the template model.

In order to compute the contribution of each material to each body tetrahedron, we use a Monte Carlo sampling approach. For each muscle/tendon/bone tetrahedron T_m, we generate one sample for each mm^3 of the volume of T_m. Specifically, we generate random samples using a uniform distribution around the centroid of T_m until the desired number of samples is reached. Using those locations, we perform look-ups in the AABB tree of the body tetrahedrons T_b and count the contributions of those samples inside the body. In the forward simulation for the animation stage, we use a time step $h = 1/30s$, and we build the mass matrix $\mathbf{M}$ assuming uniform density of the material in the body, meaning that the per-vertex mass is proportional to the sum of the volumes of the tetrahedra in which that vertex is present. Our approach proved robust, and excessive parameter tuning was not needed. The material parameters we used to generate results are: $k_{\text{bone}} = 10^{-1}$, $k_{\text{def\_bone}} = 7 \times 10^{-4}$, $k_{\text{def\_tendon}} = 3 \times 10^{-4}$, $k_{\text{def\_muscle}} = 2 \times 10^{-4}$, $k_{\text{def\_soft\_tissue}} = 10^{-4}$, $k_{\text{muscle}} = 10^{-3}$. The penalty term γ was increased from 0 to $\gamma_{max} = 10^7$ by factors of 10. In our experiments, increasing γ further did not produce any visible differences.

Discussion on the symmetric as-rigid-as-possible energy. Our reconstruction algorithm is formulated as an inverse problem which caused many issues in our experiments in terms of stability and numerical accuracy. At first, we implemented the optimization using standard ARAP elastic potential which proved robust for the forward simulation. However, for the rest pose optimization, we found that the inverse of the reference shape matrix D_M (which was no longer constant) caused problems during the reconstruction in cases of degeneration or inversion of tetrahedra. To illustrate the problem, we formulate a simple 1D ARAP-like deformation energy:

$$E_{sdef}(l_P, l_R) = k \left|\left| \frac{l_P}{l_R} - 1 \right|\right|^2 \tag{15}$$

where k is the stiffness, l_R is a rest-pose and l_P is a deformed element length. The goal of the inverse problem is to find a rest-pose that minimizes the deformation energy $l_R = \arg\min_{l_R} E_{sdef}(l_P, l_R)$. The problem appears when the current configuration has element l_P inverted ($l_P = -1$) but undeformed in the rest-pose ($l_R = 1$). A correct solution is $l_R = -1$; however, the singularity of E_{sdef} at $l_R = 0$ causes Newton's method to iterate towards wrong result as shown in Fig. 18. Fig. 19 shows comparison of forward simulation of ARAP and symmetric ARAP energies.

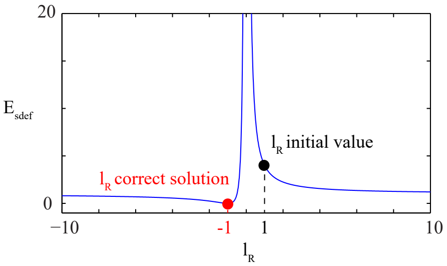
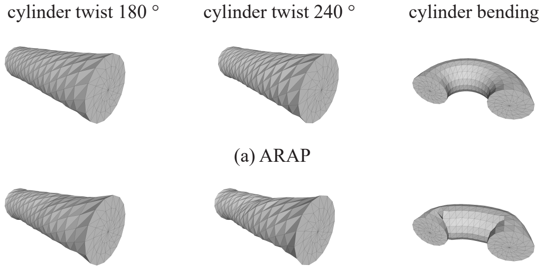


Figure 18: *Illustration of the rest pose optimization instabilities of the ARAP-like energy in 1D caused by inverted element. Newton's method does not converge to the correct solution of the minimization problem $l_R = -1$ (red dot) when initialized as $l_R = 1$ (black dot).*



(a) ARAP

(b) symmetric ARAP

Figure 19: *We compare results of the forward simulation of a cylinder deformation using the ARAP (a) and our symmetric ARAP energy (b). Although the symmetric ARAP energy converges to slightly less smooth results, it is much more robust in inverse body modeling. Neither simulation includes a volume preservation term which could be used to improve the visual quality of the results.*