Workshop on Virtual Reality Interaction and Physical Simulation VRIPHYS (2013)
J. Bender, J. Dequidt, C. Duriez, and G. Zachmann (Editors)

Physics-based Human Neck Simulation

Zhiping Luo[†] and Nicolas Pronost and Arjan Egges

Virtual Human Technology Lab
www.vhtlab.nl
Department of Information and Computing Sciences, Utrecht University

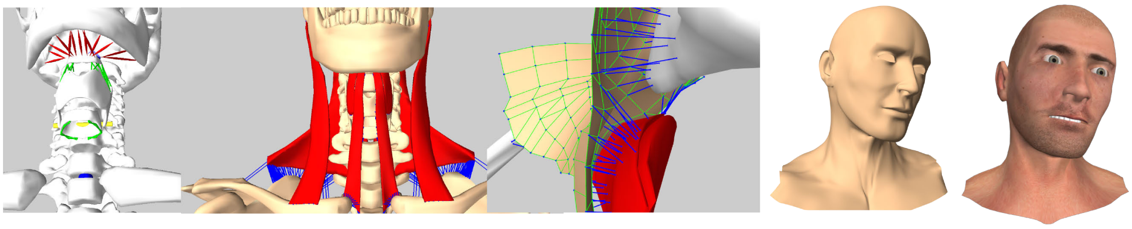

Figure 1: *We simulate the biomechanics of the human neck, combine it with a physics-based skinning approach, and we use output of the simulation to drive an animated character. From left to right: skeleton, skeleton-driven muscle deformation, skinning, simulation, animation.*

Abstract

In deformable character animation, the skin deformation of the neck is important to reproduce believable facial animation. The neck also plays an important role in supporting the head in balance while generating the controlled head movements that are essential to many aspects of human behavior. However, neck animation is largely overlooked both in computer graphics and animation due to the complexity of the cervical anatomy. This paper presents a physical human neck model based on biomechanical modeling. Relevant anatomical structures part of a 3D model of the human musculoskeletal system are modeled as deformable or linked rigid bodies. We couple the soft-hard bodies using soft constraints via elastic springs and form a Lagrangian dynamic system. The simulation of dynamic skin deformation is achieved by automatically binding the skin vertices to underlying bodies in an anatomically correct manner. Experimental results are provided and show the high level of realism that our model offers. In addition, the simulation runs at interactive rates on a modern computer.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

---

1. Introduction

Characters should be capable of expressing their physiological and emotional state convincingly in a 3D virtual environment. To achieve that, many efforts have been made to increase the realism of expressive animation systems, for example by animating the face [SNF05, Fra05, ZPS04], by simulating crying [TE09], by detailed skinning of a skeleton-based deformable body [KCvO07], or by simulating physiological phenomena such as breathing [ZCCD04].

As an aspect of non-verbal communication in character animation, neck animation should be considered as of importance. First, especially when viewing a character up close, displeasing distortions in neck motion are easily noticed, and may result in a drastic decrease of perceived realism. Sec-

---

[†] e-mail:z.luo@uu.nl

DOI: 10.2312/PE.vriphys.vriphys13.051-060

ond, the neck plays an important role in supporting the head in balance while generating the controlled head movements that are essential to many aspects of human behavior.

However, to our knowledge the neck animation has been largely overlooked both in computer graphics and animation despite [LT06] and [VLD98]. The latter research has investigated human neck simulations which visualizes motions that are generated kinematically and biomechanically. However, such models generally are not suitable for deforming skin. More detailed models accounting for the deformation of underlying soft tissues have been proposed for both the upper human body [LST09] and the lower body [ABT12]. However, these techniques are not applicable in interactive character animation due to the high computational cost.

Current approaches rely on approximations by using techniques such as skeleton-based skinning [KCvO07] for real-time neck animation. Even though such system may offer a level of realism suitable for visualization, they either lack physical accuracy, or they require considerable manual work, as in [LCF00]. In the case the propagation of muscle motion to the skin layer is visible, modeling of muscle deformation has been investigated based on dense motion capture data [PH08, NVH*13]. However, learning a model of muscle deformation only from captured skin motion cannot guarantee the accuracy of simulating the muscle dynamics. Attempts to alleviate this shortcoming have been proposed by [SNF05], by learning the facial muscle functions from skin motion based on a pre-defined physical face model, and [TTL12] approximated muscle activations in a fibre-driven soft body simulator from prescribed locomotion.

In this paper, we present a physical model for neck animation, where we directly investigate the biomechanics of the neck to solve the complexity of the cervical anatomy and then simulate the skin deformation based on the simulated dynamics underneath (see Figure 1). The neck anatomy was obtained from the Ultimate Human Model (UHM) data set(`http://www.cgcharacter.com/ultimatehuman.html`), which includes a complete and accurate human musculoskeletal system. We propose a novel approach to construct the musculoskeletal model that consists of deformable bodies and linked rigid bodies. We integrate the deformable bodies with the skeleton using a soft constraint concept, which leads to the skeleton driving the muscle deformation. In order to use the simulated dynamics of the underlying system for skin deformation, we bind each skin vertex to a muscle or a bone by an elastic spring, which is an automatic process. As a result, the skin deforms when the skeleton moves. We can simulate the neck at interactive rates because our modeling is based on linear elasticity (continuum) theory which is fast and easy to implement.

2. Related Work

We broadly classify the approaches proposed for character animation into the following categories. Our approach falls into the category of physics-based methods.

Geometric skinning: Because of its simplicity and efficiency, skinning of a skeleton-based deformable body is extensively used in real-time character animation, in particular using linear blend skinning. This skinning technique binds each skin vertex to one or more joints. Such a basic skinning technique will exhibit artifacts including skin collapsing effects, though some improvements to this technique have been proposed. For example, [KCvO07] can remove some artifacts yet they still fall short of delivering natural skin deformation, and producing realistic musculature or dynamic effects.

Example-based techniques: Such techniques beforehand provide a number of input examples, and then synthesize the surface deformation using either direct interpolation between examples [LCF00], or more accurate, yet more complex example interpolation [RLN06]. A thrid approach works by fitting the linear parameters to match the provided examples [MG03]. Generally, the level of realism is limited by the number of provided examples. However, generating the examples can be costly, requiring a lot of memory to store them as well as a lot of manual work by an animator [LCF00].

Capturing real subjects: These methods either exploit a 3D scanning device to directly capture the skin deformation [ACP03], or complete the shape based on motion capture data of real people [PH08, ASK*05]. Furthermore, muscle deformations can also be captured [NVH*13]. While these approaches are highly accurate, they require expensive hardware and are subject-specific.

Physics-based methods: From the anatomical or physical view, it is logical to animate characters by simulating the underlying musculoskeletal structures. Generally, the skeleton is modeled as an articulated multibody dynamic system [LT06, SLPF11], muscles, fat and skin tissue can be modeled by either the finite element method [CGC*02, GLDW10, MDM*02] or a mass-spring system [ZCCD04, ZPS04, Fra05].

Assassi et al. [ABT12] developed a human lower body model consisting of finite element simulations, while Lee et al. [LST09] focused on the upper body. For visualization of the skin deformation, the former directly renders the model surface, whereas the latter embeds a high-resolution skin surface as the visualization geometry by means of barycentric interpolation of the surface nodes from the nodes of the tetrahedral simulation mesh. Physically based methods are known to have a high computational cost despite obtaining a high level of realism that includes visible muscle motion.

In this paper, we will present a physics-based neck simulation. We have developed a neck simulation that is anatomically accurate, but we take into account only the muscles that are most important for displaying a realistic skind deformation. Due to this optimization, our model can be simulated at interactive rates.

3. Modeling

This section presents the neck mechanical system of which each major component is described hereafter, namely the skeleton, musculoskeletal structure, skin model and the numeric simulation.

3.1. Skeletal Model

In character animation, generally head movements including flexion, extension, bending and rotation are generated directly by the configurations of joints in the neck spine instead of muscle actuation, and consequently result in deformation of the surrounding muscles. Consulting reference on the anatomy in the UHM data set, C1-C7 cervical vertebrae, hyoid, thyroid and cricoid are in the neck, and certain neck muscles span the bones including sternum, skull, clavicle scapula, costal cartilage and T1-T12 thoracic vertebrae. In our model, we will only model the muscles (see Section 3.2.1) which are the most relevant to skin deformation, therefore only the bones which one or more of these muscles span are incorporated into our skeletal model.

We model the skeleton as an articulated, multi-body dynamic system where the bones are rigid bodies. As shown in Figure 2, the skeletal model contains skull, hyoid, thyroid, cricoid, C1-C7, the base that is the combination of sternum, clavicle scapula, costal cartilage and T1-T3, and 3-DOF ball joints inserted between adjacent vertebrae, C1 and skull, base and C7 by carefully locating the pivot points as [LT06]. Additionally, we adopt the structure from [SLPF11] which consists of a revolution joint constraining thyroid and cricoid using the same way as locating ball joints, crossed elastic springs (Eqn. (8)) connecting cricoid and base, hyoid and thyroid, point-to-point muscle actuators (Section 3.1.1) coupling skull and hyoid.

Since Lee et al. [LST09] modeled the upper human body also from the UHM data set, hence we are motivated to use the same mass value assigned to each bone, and the details of the parameters are provided in [Lee08]. However, unlike their approximation of the inertial properties of the skeleton from the dense volumetric mesh, we approximate the inertial parameters of each bone directly from its density $d = m/v$ where m is the mass and v the volume.

3.1.1. Muscle Actuator

We model the muscle actuator as a linearized Hill-type muscle model [LT06]. The total muscle force is the sum of the forces from a contractile element (CE) and a parallel element (PE), whereas the length of the tendon is assumed be constant. The CE force is expressed as

$$f_c = \alpha f_o \mathbf{F}_l(l), \tag{1}$$

where $0 \leq \alpha \leq 1$ is the activation level of the muscle, and f_o the maximum isometric force of active muscle. The force-length curve $\mathbf{F}_l(l)$ is in the form

$$\mathbf{F}_l(l) = \max(0, 0.5(1+\cos(2\pi(\frac{l-l_o}{l_o})))), \tag{2}$$

where l is the length and l_o the optimal muscle length at which the maximum isometric force of active muscle is developed. The minimum and maximum length at which muscle can produce force is set to $0.5l_o$ and $1.5l_o$, respectively.

The PE force is expressed as

$$f_p = \gamma f_o \min(\frac{l-l_o}{l_{max}-l_o}, 1.0) + d_m\dot{e}, l \geq 0, \tag{3}$$

where l_{max} is the maximum stretched length of the muscle, d_m the damping coefficient, $\dot{e}$ the strain rate, γ the weighting factor of the passive tension in $f_m = f_c + f_p$.

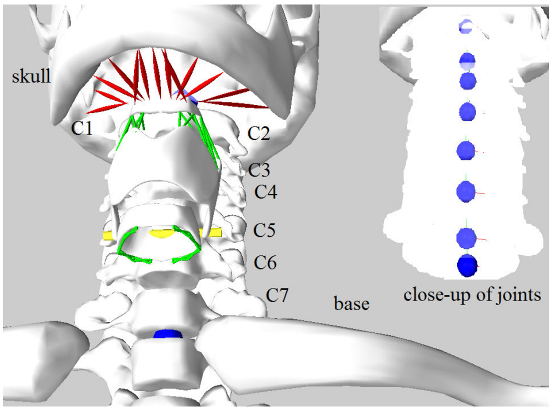

Figure 2: *The neck skeleton where red lines represent muscle actuators, green lines springs, yellow cylinder the revolution joint and blue dots the pivots of the eight cervical joints.*

3.2. Musculature and Skin Structure

3.2.1. Muscle modeling

The muscles close to the skin layer are chosen as the relevant underlying soft tissue which are trapezius, sternocleidomastoid, sternohyoid and thyrohyoid. We model them as deformable bodies with volume preservation introducing visual richness of a more detailed 3D muscular model which can facilitate the skin deformation. We use finite element analysis as the convenience in volume conservation. Therefore the simulation mesh that is a discrete representation of the muscle volume is generated beforehand based on the geometric data of polygon mesh in the UHM data set.

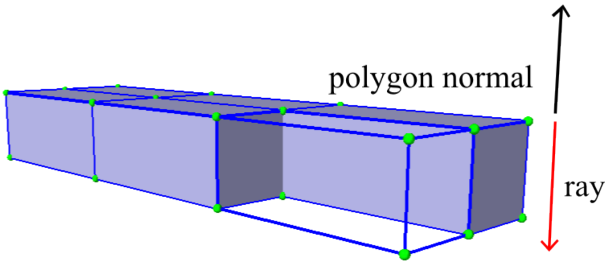

Figure 3: *Illustration of the meshing method. The polygon faces are extracted for which normals are nearly parallel, ray along the inverse normal direction of each vertex is cast, and then the inner node of a hexahedron is defined by picking a point on the 3D line of the ray according to our parameters (segments and offset).*

There are several meshing algorithms [LS07, MBTF03, Si06] handling complex geometry that can be applied. Lee et al. [LST09] used Monilo et al.s' [MBTF03] method for the meshing of polygon meshes from the UHM data set by first generating a Body-Centered-Cubic tetrahedral lattice completely covering the volume of the human body, and then use an algorithm as in [SDF07] to cut this lattice along the skin surface to obtain the volumetric mesh of the human body which resolves the muscle models. However, automatical meshing using the geometric data can produce a large amount of elements, hence make the simulation impractical. Despite we can generate coarser simulation meshes by increasing the element size as shown by Lee et al.s' approach [LST09]. Refinement is required in the regions with high-curvature feature due to higher aspect ratios the elements exhibited and so the large number of simulation elements is still there.

In order to reduce the number of simulation elements, we do not generate the volumetric meshes using the complete geometric data of the UHM data set. We start by a cleaning step which subjectively select the geometric data of each muscle polygon mesh as the input to the next step, meshing. Next we will explain how the pipeline is realized.

Based on an observation of the geometric complexity of the four selected muscles in the UHM data set, we individually take care of each polygon mesh. For sternocleidomastoid, sternohyoid and thyrohyoid particular faces are extracted for which adjacent vertex normals nearly parallel to each other as regular polygons are distributed on the surface. We cast rays along the inverse normals and take the points as new nodes on the line according a pre-defined offset (2.5 mm) and number of segments (1) (Figure 3). We generate 91 hexahedrons for sternocleidomastoid, 90 for sternohyoid and 152 for thyrohyoid. However, this method is not applicable to trapezius due to irregular faces and higher complexity in its geometry. Keeping the effort to reduce the model size, for trapezius, we only extract the neck part because, from an anatomical view, the skin patch at the back of the neck is mainly influenced. A tetrahedral mesh generator [Si06] is used, and using a maximum element volume of 1000 mm^3 and minimum radius-edge ratio of 2.0, the final simulation mesh for trapezius results in 3,497 tetrahedrons.

We use the finite simulation method [Coo94] to solve the governing partial differential equations for continuum behavior. On account of the fact that linear elasticity in constitutive models are both stable and computationally cheap, a method called *Stiffness Warping* proposed by Müller and Gross [MG04] can be integrated to handle the displeasing distortion in large rotational deformation, each simulation element is modeled as a co-rotated linear elastic material [MG04] instead of a hyperelastic Mooney-Rivlin material as in [LST09]. It is worth noting that the numerical stabilities of the corotational FEM can be further improved by the approaches proposed in [GW08].

In our FE models, the solid elements are isoparametric elements associated with a defined local coordinate system, the isometric coordinates. Within an element, the position vector in the global Cartesian coordinate system written as a function of the isoparametric coordinates is the linear interpolation of the spatial coordinates of the element nodes by the element shape functions, and the same parametric interpolation is also used for the displacement field. In this paper, we use the shape functions for hexahedral and tetrahedral element defined in FEBio [MEAW12].

Within each element e, the deformation map Ψ maps every point $\mathbf{X}$ in the undeformed body to point $\mathbf{x} = \mathbf{X} + \mathbf{u}(\mathbf{X})$ in the deformed body, and the total strain is obtained by integrating the strain tensor $\mathbf{F} = \partial\Psi/\partial\mathbf{X}$ at each point over the entire element in the form

$$\varepsilon = \frac{1}{2}(\mathbf{C} - \mathbf{I}), \tag{4}$$

where $\mathbf{C} = \mathbf{F}^T\mathbf{F}$ is the deviatoric Cauchy strain tensor. Given the Young's modulus and Poisson's ratio that are used to form a 6×6 matrix $\mathbf{E}$, the Cauchy stress tensor is calculated by the equation

$$\sigma = \mathbf{E} \cdot \varepsilon. \tag{5}$$

The elastic forces f_e exerted on the element nodes are derived from the total strain energy and turn out to be linearly dependent on the nodal displacement $\hat{x} = x - x_0$ by using the Cauchy strain:

$$f_e = -\frac{\partial \varepsilon}{\partial x} = \mathbf{K}_e \hat{x} \tag{6}$$

where $\mathbf{K}_e$ is the 12×12 stiffness matrix of the element and the calculation is described in [MDM*02].

The artifacts arising from large rotational deformation that happens in neck movements where moment is generated, are removed using the warped stiffness concept on element scale [MG04]. The rotated stiffness matrix of an element is calculated by $\mathbf{K}_e^{'} = \mathbf{R}_e \mathbf{K}_e \mathbf{R}_e^{-1}$ where the rotational component $\mathbf{R}_e$ of the element is calculated using the method described in [MDM*02]. Therefore f_e is modified as

$$f_e^{'} = \mathbf{K}_e^{'} \hat{x} = \mathbf{R}_e \mathbf{K}_e \mathbf{R}_e^{-1} x - \mathbf{R}_e \mathbf{K}_e x_0, \tag{7}$$

and the whole stiffness matrix $\mathbf{K}^{'}$ of the entire mesh is the sum of all element's rotated stiffness matrix.

3.2.2. Integration with Skeleton

In our model, the muscles deform when the skeleton moves. In this section, we describe the method to integrate the skeleton with the FE models.

A widely used solution consists in constraining any node of the simulation mesh that lies inside or near a bone to a fixed position within the local coordinate frame of that bone. For example the FE simulation software FEBio [MEAW12] provides a tool where we can subjectively select such nodes by area, surface, or volume selection function. Lee et al. [LST09] proposed a method to automatically find the nodes to be constrained, and its feasibility lies on the fact that they created one simulation mesh resolving all tissues except the bones of the human body and the mesh overlaps with the skeleton tightly. Unlike their modeling, we model part of the superficial muscles individually and only some nodes (mainly in the attachment areas where the muscles connect to the bones) of their simulation meshes lie inside or near bones, hence it is difficult to automatically find these nodes accurately. Eventually we opted for the selection functions provided by FEBio.

Some parts of the simulation mesh are very close to the skin surface in our model, especially of sternocleidomastoid. It will lead to odd looking patches of the skin if the attached nodes move rigidly with the bones since we will propagate the simulated nodal motion to the skin layer and consequently deform the skin (see Section 3.2.3). We address this issue by implementing the soft constraint concept proposed in [LST09]. We developed an interface to FEBio so the simulation mesh can be imported into its environment. Nodes are selected based on the observation on the anatomy available in the UHM data set. In more details, the nodes which are the closest to the bones underneath are selected. Despite it requires manual labour, in practice, this process is fast and it is tolerable, as a one-time modeling cost. Next we project the node to the surface of the bone by a fast closest point projection method (Section 3.2.3) and the projection point is attached to the bone. Finally, we connect the node and its projection point using an elastic spring which applies traction forces on the node as the bone moves. Figure 4 depicts how the FE models are coupled with the bones underneath.

3.2.3. Skin Model

Simulating the dynamic skin deformation based on the simulation of the underlying musculoskeletal model has been investigated. [ABT12] and [LST09] generate the simulation mesh resolving the skin tissue in their lower and upper body, respectively. Nonetheless, the former directly renders the model surface so the visualization largely depends on the element size while the latter barycentrically embeds an high-resolution skin surface as the visualization geometry into the mesh. While it experimentally shows high degree of realism in skin deformation, we cannot afford such simulation considering our performance objective.

In the torso model in [ZCCD04], skin simulation is decoupled from the simulation of the underlying model by first recording the trajectories of pre-selected points attached to the model that are the control vertices of a NURBS (Non-uniform rational B-spline) surface. The surface shape is then updated to show skin deformation. The shape of the visualization geometry is implicitly defined by how the control vertices are selected.

We use the mass-spring approach which is popular in real-time simulators like facial animation [Fra05,ZPS04] because it is simple to implement and meets our performance requirements. The human skin is experimentally shown as a multi-layered elastic material with non-linear stress-strain relationship based on the study on the rabbit abdominal skin [Fra05], and the epidermal layer is stiffer than inner layers so its spring stiffness are set to make it moderately resistant to deformation [ZPS04].

We use linear springs embedding a damper where the spring force magnitude is a function expressed as

$$f_s = k_s(l - l_0) + d_s \dot{e}, \tag{8}$$

where k_s, ds are elastic and damping coefficients, l, l_0 are its length and slack length, $\dot{e} = \dot{l}/l_o$ is the strain rate.

The geometry data of the skin mesh serves as the basis to construct a mass-spring network representing the epidermal layer. We do not model the inner layers, yet we use springs as the medium through which the underlying simulated motion is propagated to the upper mass-spring network, and consequently result in the dynamic skin deformation.

From an anatomical view, a patch of the skin surface deforms mainly from the anatomically closest tissue underneath. On account of this fact, the propagation medium should be located between them. To do that, we connect the nodes in the epidermal layer with the points that are attached to the anatomically closest bone or muscle. These points are automatically found based on closest point projection method as follows.

First, to reduce the computation time of finding closest points, we build the oriented bounding box (OBB) trees of the polygon mesh that combines all surface meshes of the underlying bodies using an implementation of Goatishly et al.'s algorithm [GLM96]. Each skin vertex is projected onto the closest OBB in world coordinates. Secondly, given a skin vertex we decide on which body surface its projection ex-

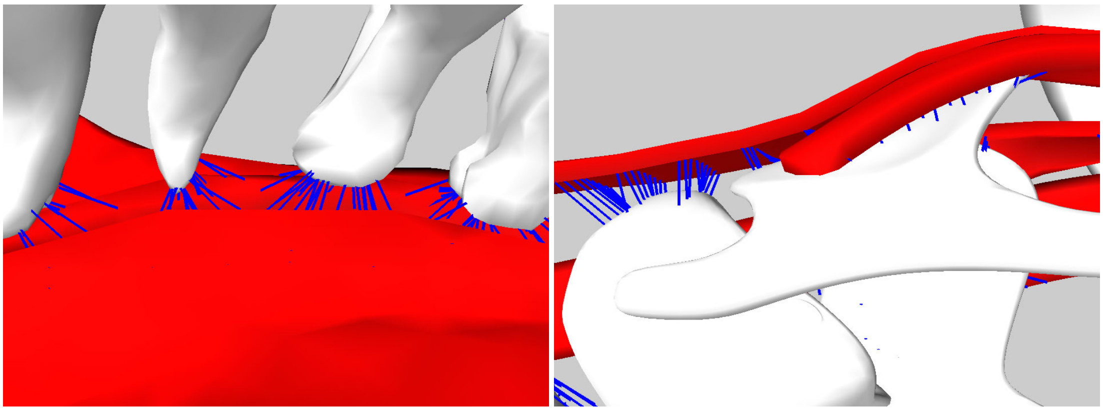

Figure 4: *Illustration of the soft constraint on FE models, the blue lines represent constraint springs.*

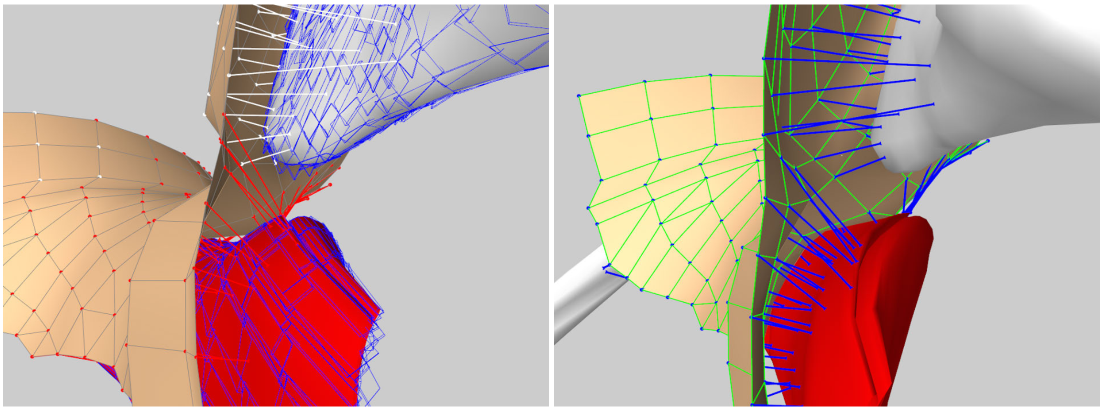

Figure 5: *Illustration of the skin modeling. Left: White and red lines depict the projection maps and blue rectangles are the volume bounding boxes. Right: Blue lines represent connecting springs and green lines the epidermal springs.*

actly is. We test the projection against each triangulated body surface. If its barycentric coordinate $(1-\lambda_1-\lambda_2, \lambda_1, \lambda_2)$ with respect to a triangle satisfies $0 \leq \lambda_1 \leq 1, 0 \leq \lambda_2 \leq 1$ and $0 \leq \lambda_1 + \lambda_2 \leq 1$ then the projection is in this triangle belonging to the body surface. If the body is a FE model, the projection is barycentrically embedded by the nodes of the element containing it, otherwise is attached to the coordinate frame of the rigid body. Finally springs connecting the skin vertices and their projection are added into the spring network (see Figure 5).

3.3. Numerical Simulation

Let $\mathbf{q}$ be the positions, and $\mathbf{u}$ the velocities of all the dynamical components of the mechanical system, with $\dot{\mathbf{q}}$ related to $\mathbf{u}$ by $\dot{\mathbf{q}} = \mathbf{Q}\mathbf{u}$. Let $\mathbf{f}(\mathbf{q},\mathbf{u},t)$ be the force produced by all the force effector components, let $\mathbf{M}$ be the (block-diagonal) composite mass matrix. We can ensure that $\mathbf{M}$ is constant by representing rigid-body velocity and acceleration in body coordinates. The following governing equation describes the Lagrangian dynamics of the system according to Newton's second law

$$\mathbf{M}\dot{\mathbf{u}} = \mathbf{f}(\mathbf{q},\mathbf{u},t), \tag{9}$$

and it is constrained due to the presence of bilateral (joints and point-surface constraints) and unilateral constraints (joint limits) in the form

$$\mathbf{G}(\mathbf{q}) \geq \mathbf{0}, \quad \mathbf{N}(\mathbf{q})\mathbf{u} \geq \mathbf{0}. \tag{10}$$

For FE simulation, a lumped mass model [SLPF11] is used to ensure that $\mathbf{M}$ is block-diagonal.

As the presence of the FE models, the system is stiff and therefore we need an implicity time integrator for efficient performance. We opt for the simulation framework of Stavness et al. [SLPF11], where the Newmark integrator with $\lambda = \frac{1}{2}$ and $\beta = \frac{1}{4}$ is used and hence the step-based updating rule is formulated as a mixed linear complementarity problem which is solved by Pardiso solver [SG04].

4. Experiments

In the experiments, we use the parameters of the muscle actuators (maximum muscle forces) and crossed springs (stiffness) from [SLPF11]. The FE models have a Young's modulus of 50 MPa and a Poisson's ratio of 0.4. We set stiffness to $1.2\times10^3 N\cdot m^{-1}$ to epidermal springs, connecting springs to $0.5\times10^3 N\cdot m^{-1}$, and constraint springs to $2.0\times10^3 N\cdot m^{-1}$. All simulations are run on a laptop with a Intel Core i7 2.4Ghz processor and 6 GB memory.

The head movements mainly include flexion, extension, rotation and lateral flexion. They are:

- **flexion**: moving the head forward at the joint just below the skull.
- **extension**: moving the head backward at the joint just below the skull.
- **rotation**: turning the head to the side (right or left) at a joint below the skull.
- **lateral flexion**: moving the head toward the shoulder (left or right) at a joint below the skull.

In facial animation, a notable visualization is the neck skin deforms when the jaw opens. Therefore, to show the level of realism of the neck animation our model can offer, we demonstrate the result of the four head movements and the jaw openning.

We extract frames under extreme postures for each testing movements from the simulation of the musculoskeletal system and present them in Figure 6.

A comparison with the popular technique, namely the linear blend skinning, is conducted. We compare the animation at a pose, and by referring to a static photo of a human neck at the same pose, to show that the simulation deliver more realistic results. The comparison is shown in Figure 8. Muscles are modeled symmetrically from the left and right, therefore we only show the simulation frames of one-side rotation and lateral flexion.

Our model based on biomechanical modeling also can reproduce a wide range of lifelike animation. It is interesting to control the model by exciting the muscle acttutors and rotate the cervical joints, generating realistic animations of the head. We show some frames from such animations in Figure 7.

Recall that, due to performance constraints, we do not model all the superficial muscles available in the UHM data set, therefore the simulated musculoskeletal dynamics is not complete enough to conform with the real neck motion while can guarantee the high-fidelity emulation (as shown in Figure 8). We also bind the vertices to underlying bodies according to the anatomy we incorporated, hence the less anatomy we use, the more coarse the vertex binding is.

Modeling even more anatomical structures, or a nonlinear multilayered skin modeling like [ZPS04, Fra05] will definitely increase the memory consumption in order to store the states during the simulation. Additional constraints (attaching proper finite nodes to bones) would be required and therefore would increase the size of influenced matrices in the numerical simulation, hence resulting in higher time complexity. Based on these facts, we trade the speed for accuracy, yet as shown in the figures and accompanying video, our model still offers realistic animations.

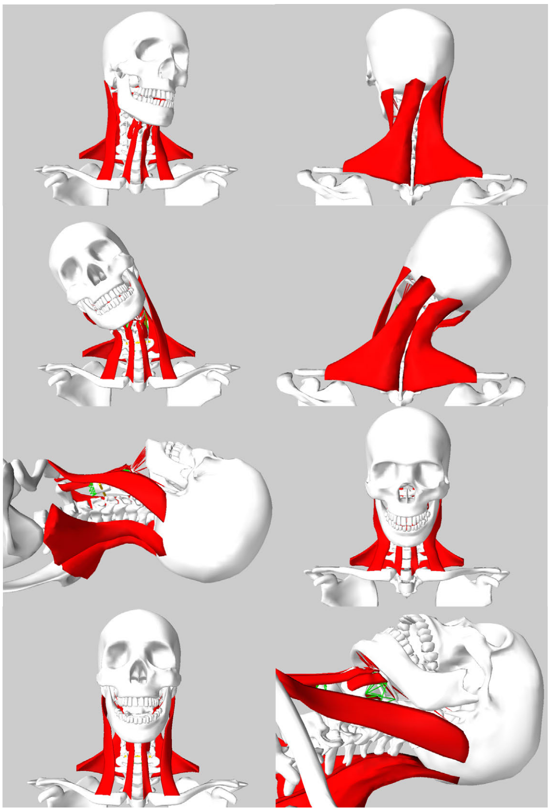

Figure 6: *Frames from the simulation of the underlying components (muscles, skeleton, actuators and springs.).*

5. Conclusion and Future Work

We have described a physics-based model of the human neck. The simulations were performed at interactive rates (average 20 FPS) and results showed the realism that the model can deliver. Considering the computational cost, we only modeled the most relevant anatomical structures available in a 3D digital model and form a Lagrangian dynamic system which is solved by a semi-implicity time integrator.

Our model experimentally demonstrated that physically based modeling can produce skin deformation reflecting the muscular and dynamic effects, and showed that the linear elasticity constitutive models are both stable and computationally cheap, therefore is suitable for soft tissue simulation in neck animation. Our method of skin modeling highlights

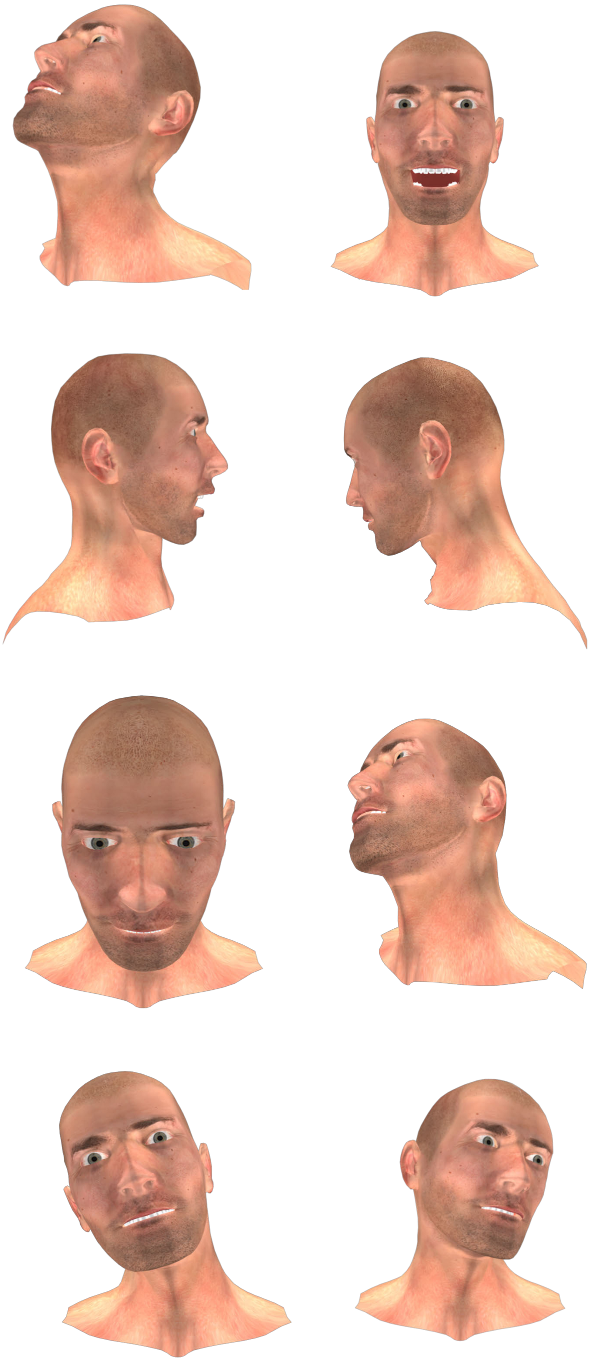

Figure 7: *Some frames of animations we generated using the simulation of our physical neck model.*

the automatical binding of skin vertices to underlying muscles and bones via linear elastic springs in an anatomical manner, and it is quite easy to implement and fast to simulate.

In our current model, the muscle actuators are only used for the interconnection between jaw and hyoid, we plan to implement a full muscle actuation model using them so that the simulation of the muscle tensioning can be achieved which is difficult in skeleton-driven methods, especially for strong emotional expression in the face. We also plan to compare our method with methods of capturing real humans, to investigate the applicability of our model in character animation.

Acknowledgments

This work is supported by the Dutch research project COMMIT-Virtual Worlds for Well-Being.

References

[ABT12] Assassi L., Becker M., Thalmann N. M.: Dynamic skin deformation based on biomechanical modeling. In *Proceedings of the 25th Annual Conference on Computer Animation and Social Agents* (2012). 2, 5

[ACP03] Allen B., Curless B., Popović Z.: The space of human body shapes: reconstruction and parameterization from range scans. *ACM Trans. Graph. 22*, 3 (2003), 587–594. 2

[ASK*05] Anguelov D., Srinivasan P., Koller D., Thrun S., Rodgers J., Davis J.: Scape: shape completion and animation of people. *ACM Trans. Graph. 24*, 3 (2005), 408–416. 2

[CGC*02] Capell S., Green S., Curless B., Duchamp T., Popović Z.: Interactive skeleton-driven dynamic deformations. *ACM Trans. Graph. 21*, 3 (July 2002), 586–593. 2

[Coo94] Cook R. D.: *Finite element modeling for stress analysis*. John Wiley & Sons, Inc., 1994. 4

[Fra05] Fratarcangeli M.: Physically based synthesis of animatable face models. In *VRIPHYS* (2005), pp. 32–39. 1, 2, 5, 7

[GLDW10] Georgii J., Lagler D., Dick C., Westermann R.: Interactive deformations with multigrid skeletal constraints. In *VRIPHYS* (2010), pp. 39–47. 2

[GLM96] Gottschalk S., Lin M. C., Manocha D.: Obbtree: a hierarchical structure for rapid interference detection. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (1996), SIGGRAPH '96, pp. 171–180. 5

[GW08] Georgii J., Westermann R.: Corotated finite elements made fast and stable. In *Workshop in Virtual Reality Interactions and Physical Simulation* (2008), The Eurographics Association, pp. 11–19. 4

[KCvO07] Kavan L., Collins S., Žára J., O'Sullivan C.: Skinning with dual quaternions. In *Proceedings of the 2007 symposium on Interactive 3D graphics and games* (2007), I3D '07, pp. 39–46. 1, 2

[LCF00] Lewis J. P., Cordner M., Fong N.: Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), pp. 165–172. 2

[Lee08] Lee S. H.: *Biomechanical modeling and control of the human body for computer animation*. PhD thesis, University of California at Los Angeles, 2008. 3

[LS07] Labelle F., Shewchuk J. R.: Isosurface stuffing: fast tetrahedral meshes with good dihedral angles. *ACM Trans. Graph. 26*, 3 (July 2007). 4

[LST09] Lee S.-H., Sifakis E., Terzopoulos D.: Comprehensive biomechanical modeling and simulation of the upper body. *ACM Trans. Graph. 28*, 4 (Sept. 2009), 99:1–99:17. 2, 3, 4, 5

[LT06] Lee S.-H., Terzopoulos D.: Heads up!: biomechanical modeling and neuromuscular control of the neck. *ACM Trans. Graph. 25*, 3 (July 2006), 1188–1198. 2, 3

[MBTF03] Molino N., Bridson R., Teran J., Fedkiw R.: A crystalline, red green strategy for meshing highly deformable objects with tetrahedra. In *In 12th Int. Meshing Roundtable* (2003), pp. 103–114. 4

[MDM*02] Müller M., Dorsey J., McMillan L., Jagnow R., Cutler B.: Stable real-time deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2002), SCA '02, pp. 49–54. 2, 4, 5

[MEAW12] Maas S. A., Ellis B. J., Ateshian G. A., Weiss J. A.: FEBio: finite elements for biomechanics. *Journal of biomechanical engineering 134*, 1 (2012). 4, 5

[MG03] Mohr A., Gleicher M.: Building efficient, accurate character skins from examples. *ACM Trans. Graph. 22*, 3 (July 2003), 562–568. 2

[MG04] Müller M., Gross M.: Interactive virtual materials. In *Proceedings of Graphics Interface 2004* (2004), GI '04, pp. 239–246. 4

[NVH*13] Neumann T., Varanasi K., Hasler N., Wacker M., Magnor M., Theobalt C.: Capture and statistical modeling of arm-muscle deformations. *Computer Graphics Forum (Proc. of Eurographics)* (2013). 2

[PH08] Park S. I., Hodgins J. K.: Data-driven modeling of skin and muscle deformation. *ACM Trans. Graph. 27*, 3 (2008), 96:1–96:6. 2

[RLN06] Rhee T., Lewis J. P., Neumann U.: Real-time weighted pose-space deformation on the gpu. *Comput. Graph. Forum 25*, 3 (2006), 439–448. 2

[SDF07] Sifakis E., Der K. G., Fedkiw R.: Arbitrary cutting of deformable tetrahedralized objects. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2007), SCA '07, pp. 73–80. 4

[SG04] Schenk O., Gärtner K.: Solving unsymmetric sparse systems of linear equations with pardiso. *Future Generation Computer Systems 20*, 3 (2004), 475–487. 6

[Si06] Si H.: Tetgen: A quality tetrahedral mesh generator and three-dimensional delaunay triangulator. Weierstrass Institute for Applied Analysis and Stochastic, Berlin, Germany, Jan 2006. 4

[SLPF11] Stavness I., Lloyd J. E., Payan Y., Fels S.: Coupled hard–soft tissue simulation with contact and constraints applied to jaw–tongue–hyoid dynamics. *International Journal for Numerical Methods in Biomedical Engineering 27*, 3 (2011), 367–390. 2, 3, 6, 7

[SNF05] Sifakis E., Neverov I., Fedkiw R.: Automatic determination of facial muscle activations from sparse motion capture marker data. *ACM Trans. Graph. 24*, 3 (July 2005), 417–425. 1, 2

[TE09] Tol W., Egges A.: Real-time crying simulation. In *Proceedings of the 9th International Conference on Intelligent Virtual Agents* (2009), IVA '09, pp. 215–228. 1

[TTL12] Tan J., Turk G., Liu C. K.: Soft body locomotion. *ACM Trans. Graph. 31*, 4 (2012), 26:1–26:11. 2

[VLD98] Vasavada A. N., Li S., Delp S. L.: Influence of muscle morphometry and moment arms on the moment-generating capacity of human neck muscles. *Spine 23*, 4 (1998), 412–422. 2

[ZCCD04] Zordan V. B., Celly B., Chiu B., DiLorenzo P. C.: Breathe easy: model and control of simulated respiration for animation. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2004), Eurographics Association, pp. 29–37. 1, 2, 5

[ZPS04] Zhang Y., Prakash E. C., Sung E.: A new physical model with multilayer architecture for facial expression animation using dynamic adaptive mesh. *IEEE Transactions on Visualization and Computer Graphics 10*, 3 (May 2004), 339–352. 1, 2, 5, 7

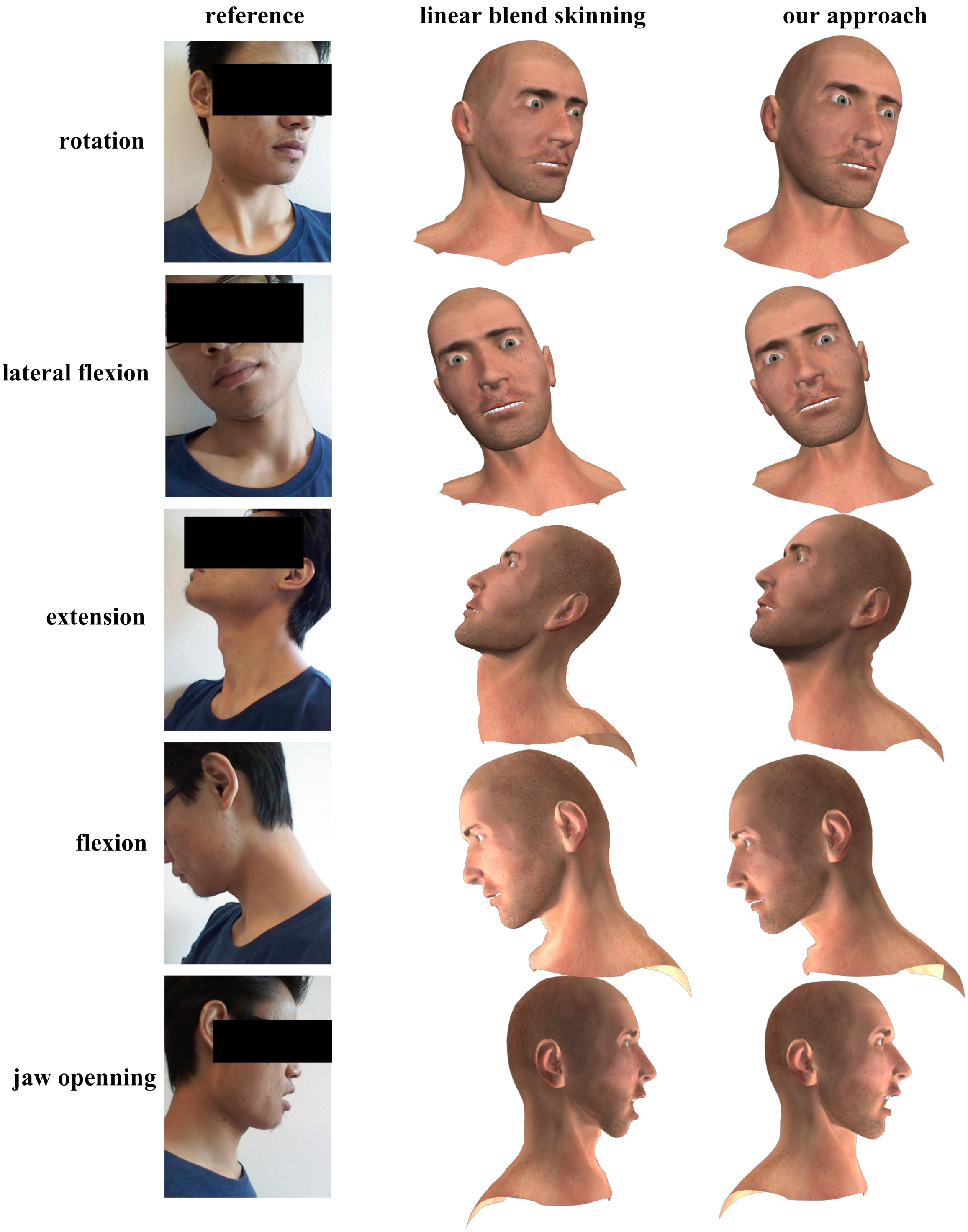

Figure 8: *Referring to the corresponding static photos, linear blend skinning exhibits artifacts and cannot generate muscular and dynamic effects reflected in the skin, while our model simulates lifelike head poses and neck skin deformation where the visible muscle motion is reflected.*