EUROGRAPHICS 2017 / L. Barthe and B. Benes
(Guest Editors)

Volume 36 (2017), Number 2

Enriching Facial Blendshape Rigs with Physical Simulation

Yeara Kozlov[1,2] Derek Bradley[1] Moritz Bächer[1] Bernhard Thomaszewski[1] Thabo Beeler[1] Markus Gross[1,2]

[1]Disney Research [2]ETH Zurich

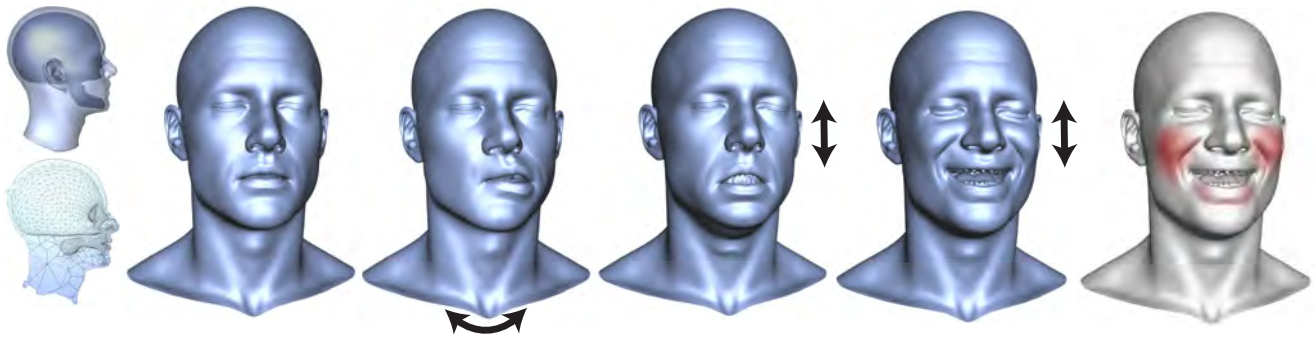

Figure 1: We add physical effects to traditional blendshape animation by enriching blendshape rigs with a simple volumetric tissue structure. This blendvolumes rig (left) allows to add, for example, secondary effects caused by head motion to a blendshape animation (center, with black arrows). To account for stiffening muscles during contraction, we further introduce blendmaterials (right) and show how both rest shape and material properties can be interpolated during simulation without introducing spurious artifacts. Shown here is the same neutral expression undergoing secondary dynamics caused by fast side-to-side head rotation and jumping. A tightened smile undergoing the same motion shows lessened dynamic effects.

Abstract

Oftentimes facial animation is created separately from overall body motion. Since convincing facial animation is challenging enough in itself, artists tend to create and edit the face motion in isolation. Or if the face animation is derived from motion capture, this is typically performed in a mo-cap booth while sitting relatively still. In either case, recombining the isolated face animation with body and head motion is non-trivial and often results in an uncanny result if the body dynamics are not properly reflected on the face (e.g. the bouncing of facial tissue when running).

We tackle this problem by introducing a simple and intuitive system that allows to add physics to facial blendshape animation. Unlike previous methods that try to add physics to face rigs, our method preserves the original facial animation as closely as possible. To this end, we present a novel simulation framework that uses the original animation as per-frame rest-poses without adding spurious forces. As a result, in the absence of any external forces or rigid head motion, the facial performance will exactly match the artist-created blendshape animation. In addition we propose the concept of blendmaterials to give artists an intuitive means to account for changing material properties due to muscle activation. This system allows to automatically combine facial animation and head motion such that they are consistent, while preserving the original animation as closely as possible. The system is easy to use and readily integrates with existing animation pipelines.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically Based Modeling

1. Introduction

Whether stunningly realistic or artistically stylized—high-quality facial animation is the pivotal element for compelling characters in contemporary filmmaking. A central requirement for a believable digital performance is the consistency between the primary motion of the head and the secondary motion of the facial tissue that it induces. While even subtle dynamics during speech can add greatly

to the realism of an animation, secondary motion simply cannot be neglected for performances involving running, jumping, and other forms of rapid motion.

Animating secondary motion is generally a challenging and tedious task if done manually. Wherever possible, this process is therefore automated using physical simulation. For facial animation, one approach is to use a detailed anatomical model of the human face as a basis for simulation [SNF05]. While this approach can faithfully reproduce facial dynamics, two limitations hinder its adoption in production environments: first, creating the anatomical model with its numerous muscles, tendons, and other tissue components is a manual and exceedingly labor-intensive process. Second, anatomically-based simulation rigs are controlled through muscle activations. Compared to the commonly used blendshape rigs, this type of control is indirect, unintuitive, and difficult to integrate into existing animation workflows.

An alternative approach is to start from an artist-created blendshape animation, which is then used as a target for physics-based simulation. Recent work by Barrielle et al. [BSC16] has demonstrated that, by interpolating forces instead of geometry, it is possible to extend the expressive range of blendshape rigs and, at the same time, automatically handle lip contacts. However, while not a focus of their work, the underlying shell-based simulation model is too crude an approximation to faithfully reproduce secondary motion in soft tissue.

In summary, the difficulty of combining artist-directed animation with physics-based simulation has so far prevented the widespread use of simulation for facial animation. In order to bridge this gap, we propose a method that combines the benefits of physics-based simulation for creating secondary motion while maintaining the ease of control offered by blendshape rigs. Rather than a detailed anatomical model, our method relies on a simple volume mesh for simulation. In contrast to conventional simulation, however, we adapt the rest state mesh according to the artist-provided blendshape animation. Doing so ensures that, in the absence of external forces such as head motion, the equilibrium states of the simulation coincide with the artist-defined blendshape poses. In order to achieve natural dynamics in the presence of head motion, our method allows artists to define per-pose material distributions for the simulation mesh. In this way, our approach captures two central properties: first, owing to the anatomical complexity of the face, different regions exhibit largely different material properties, ranging from soft fat tissue to quasi-rigid bones; second, the material properties change as a function of deformations as, e.g., muscle tissues stiffens when contracted.

Combining physics-based simulation with time-varying rest shapes and per-pose material properties makes for a powerful tool for creating natural and controllable facial dynamics. As the technical core of our approach, we propose a method for changing the rest shape of the simulation without inducing spurious dynamic effects. As a result, our method only adds dynamic motion when commanded by external forces or head motion, but in the absence of such effects, the output will correspond exactly to the input animation.

As we show in our results, our method can enrich blendshape animations with physics-based dynamic effects for a broad range of characters, including digital human actors as well as fantasy creatures. Moreover, our experiments indicate that it is essential to control material properties both spatially and temporally in order to faithfully capture facial dynamics with a simplified simulation model. Thanks to its simplicity, setting up our simulation rigs is a lightweight process that requires only a few additional steps compared to standard animation workflows. In particular, per-pose material properties can simply be painted using existing tools for shape or texture control.

2. Related Work

In order to put our work into perspective with existing methods, we start by reviewing the basic approaches for facial animation, paying particular attention to physics-based variants. We also discuss more general but related animation techniques for controlling physical simulations.

Facial Animation Techniques. There are several approaches for creating facial animation, including performance capture, blendshape animation, and anatomical models that rely on physics. Performance capture methods [BHPS10, BHB*11, KH12, FJA*14] can directly capture the animation including all the physical effects that occur, but they do not allow easy editing of the animation. Blendshape animation [LAR*14] is the most common technique, and by now is integrated into every professional animation software and is widely used throughout the entertainment industry. However, the drawbacks of blendshape animation are well-known, for example, the simple linear interpolation does not handle rotational deformations, self-collisions, physical interaction with external objects or secondary motion caused by dynamics. With the goal of improving flexibility and increasing realism, anatomical approaches that directly model facial muscles have been proposed [TW90, KHS01, SNF05, BJTM08]. These methods can simulate all the desired dynamic effects, but are extremely difficult and time-consuming to set up. This is why some researchers have focused on only small parts of the face, such as the forehead [WM15] or cheek region [MHSH10, RP07]. With a similar motivation, other works have investigated automatic initialization [LTW95] and anatomy transfer [CBJ*15]. However, even these rigs are not capable of creating production quality results and require further artistic refinement as presented in Cong et al. [CBF16].

In this work, we wish to extend blendshape animation by adding a layer of physical simulation to account for secondary dynamics, self-collision, and interaction with external objects. Our method is transparent to artists, in the sense that it only requires conventional blendshape animation sequences in order to automatically add physical effects. We note that the combination of blendshape animation and physics-based simulation has, to various extents, been investigated in previous work. For example, Ma et al. [MWF*12] use a mass-spring system on top of blendshapes, with the goal of allowing interaction between external objects and better handling of rotational deformations. You et al. [YSZ09] add a physical model of skin deformation to blendshape animation and propose to blend forces rather than shapes, such that the resulting facial animation can more closely follow the underlying physical properties. Similar in spirit, Barrielle et al. [BSC16] optimize for *blendforces* in order to achieve animations that go beyond the linear span of the

blendshape basis and automatically incorporate deformations due to, e.g., lip contacts. Similar to these methods, our approach is able to automatically prevent self-collisions and respond to contact. However, our primary focus is on reproducing secondary dynamics in the facial tissue in a natural, yet controllable way. We therefore lay aside surface-based models and instead turn to finite element simulation on volumetric meshes. Most similar to our approach is the volumetric blendshape method of Ichim et al. [IKNDP16], who combine finite element simulation with facial blendshapes in order to preserve volume and avoid self-collisions during animation, as well as respond to external forces. While we also propose a representation of *blendvolumes*, our approach is fundamentally different in that we take the artist-created blendshape animation as per-frame rest shapes for the simulation, in order to ensure that we match the input exactly in the absence of any head motion or external forces. Furthermore, we propose shape-dependent material parameters, which we call *blendmaterials* in order to account for time-varying muscle stiffness during animation. Most importantly, these contributions are coupled with a novel method for dynamic rebalancing of inertial forces to ensure physical simulation without spurious forces induced by the shape and material changes, which can cause unwanted animation artifacts.

Art-Directed Physics-Based Animation. The ability to control physical simulations according to the artistic vision is crucial in production environments. One general approach is to attract the simulation toward artist-defined target configurations by virtue of control forces. In the context of blendshape animation, one appealing option is the tracking solver by Bergou et al. [BMWG07], which adds physics-based detail while guaranteeing that the resulting simulation will match the low-frequency components of the input animation. Without modifications, however, this approach would induce equal amounts of oscillations and wrinkles everywhere on the surface. Furthermore, the locality of the tracking constraints has a very significant impact on the results, as it determines how closely the input animation will be tracked. However, adjusting this locality is a technically-involved process that can be very cumbersome and frustrating.

Instead of applying arbitrary, non-physical control forces, another line of work has explored the use of muscle-like control forces that arise due to a change in rest state. For example, Kondo et al. [KKA05] propose to keyframe rest state changes in order to guide the simulation of a deformable object. In contrast, Coros et al. [CMT*12] automatically compute time-varying rest state changes in order to control the motion of deformable objects. Instead of active control, Martin et al. [MTGG11] propose an example-based approach for designing elastic materials with desired passive deformation behavior. Galoppo et al. [GOM*09] present another approach for passive control. They propose to specify pose-specific dynamic targets and use them to interpolate elastic forces that direct the simulation towards the desirable behavior. Li et al. [LXB16] share a similar goal of adding physical effects to triangle-mesh animations. However, an important difference is that their method relies on a single rest-state to drive the simulation for the entire animation sequence. In contrast, our method changes the rest-state of the face in order to conform with the artist-provided blendshape animation. This has the crucial advantage that our method's quality does not degrade if the input animation exhibits severe deformations (as is often the case for fantastic characters). Note that a key component of our work is to compensate for the forces induced by this rest-state change, so that all dynamic effects in the facial tissue are induced by head motion or other external forces.

Perhaps closer to our problem setting, several works have investigated ways of using physics-based simulation in the context of character skinning and animation. One example is the physics-based skinning method by McAdams et al. [MZS*11], who drive a volumetric simulation mesh with an animated skeleton. While this approach extends the expressive range of a given rig using simulation, Rig-space physics [HMT*12, HTC*13] enables simulation within the subspace of a rig. This approach ensures that all of the automatically computed secondary motion is expressed in terms of rig controls, but it can only produce motion that lies within the space of the rig. Dyna [PMRMB15] learns a statistical model of soft tissue deformation for changing body poses, and uses it to generate plausible secondary motion effects for human bodies, but does not account for other physical interactions. Xu and Barbič [XB16] enrich character animations by computing the inertia induced by changes in the character's pose and use it to induce elastic forces. In contrast, we consider these forces undesired and introduce a means to remove such spurious forces and hence preserve the inertial energy in the system when blending poses or materials. This is of central importance, for example, in speech sequences where spurious forces can lead to visibly out-of-sync animations.

3. Overview

Our goal is to enhance traditional blendshape animation with physical effects, such as secondary motion caused by dynamics. To this end, we propose to add a layer of physical simulation on top of blendshape animation. We wish to avoid a complex muscle-based physical face model and instead we create a simple volume of tissue between the skin surface and the skull and jaw, which consists of a small number of elements. This volume is created to maintain direct shape correspondence with the blendshape subspace, defining a *blendvolume* rig. A central component of our approach is to allow spatially-varying per-expression material parameters, accomplished with a *blendmaterial* rig that can account for physical phenomena such as tissue stiffening when muscles contract. A description of this physical face model is given in Section 4.

Given an input blendshape animation, our method can enrich the motion by adding physical effects such as secondary dynamics. This is accomplished by performing a dynamic simulation of the element volume, using the shape-dependent material parameters, and a novel per-frame rest shape update approach that allows for large shape changes without inducing unwanted dynamic effects. The details of this blendshape-driven simulation approach are given in Section 5.

Finally, we describe the steps necessary to initialize our face model, including the creation of the blendvolume and blendmaterial rigs, as described in Section 6, and we highlight important implementation details in Section 7. Section 8 demonstrates our method on several animation sequences of both humans and fantasy creatures. The overview of our approach is illustrated in Fig. 2.

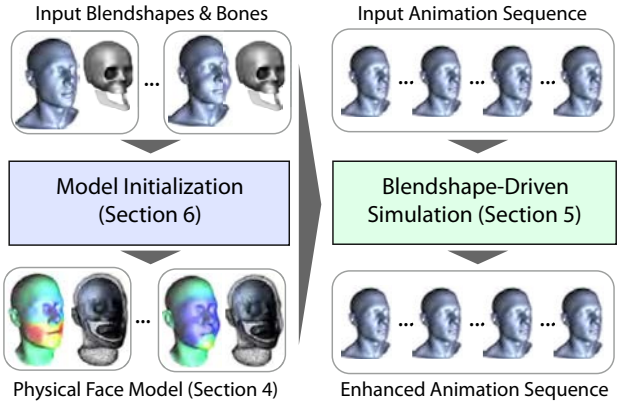

Figure 2: Overview – our goal is to add physical simulation to blendshape animations. First we build a *blendvolume* rig from a set of input blendshapes and corresponding skull and jaw bones (Sec. 6). This physical face model is then used to augment an input animation sequence with physical effects, such as secondary dynamics (Sec. 5).

4. Physical Face Model

Our method relies on finite-element-based simulation to compute secondary dynamics in facial tissue. As our goal is to avoid the geometric complexity of anatomical models and the corresponding setup times, we use a connected volume mesh for all of the facial soft tissue. However, in order to capture the salient properties of muscles and other soft tissue with this simplified model, we allow for time-varying rest states and spatio-temporally varying material properties. Aside from being convenient in terms of setup and artistic control, our model draws on the same insights from anatomy used in previous work (see, e.g., Fan et al. [FLP14]): changes in rest shape approximate the effect of muscle activation, while changes in material properties account for the stiffening of living tissue with increasing deformation. In the following, we introduce our physical face model which, using blendshape rigs as a basis, interpolates both rest state geometry and material properties.

4.1. Geometric Model for Animation

We follow the traditional approach for blendshape animation, where poses $\mathbf{S}$ are created from linear combinations of base shapes $\mathbf{B_S}$ which encode the relative displacements from the neutral shape $\mathbf{S_0}$ as

$$\mathbf{S} = \mathbf{S}_0 + \mathbf{B_S}\mathbf{w}, \tag{1}$$

where $\mathbf{w}$ are blendweights. For simulation, we augment the surface shape $\mathbf{S}_0$ with a volumetric mesh $\mathbf{X}_0$, representing the soft tissue underneath. Analogously to the blendshape basis $\mathbf{B_S}$ that encodes the shape variation of the surface, we can describe the volume variation of the soft tissue via a *blendvolumes* basis $\mathbf{B_X}$, which encodes the nodal displacements relative to $\mathbf{X}_0$. In this way, we define a pose-dependent rest state $\mathbf{X}$ of the soft tissue as

$$\mathbf{X} = \mathbf{X}_0 + \mathbf{B_X}\mathbf{w}. \tag{2}$$

Details on how to construct the blendvolumes basis $\mathbf{B_X}$ are given in Section 6. In addition to the soft tissue, our method also expects the hard tissue, i.e., skull and jaw, to be given as input. Fig. 3 shows the individual components of our geometric model.

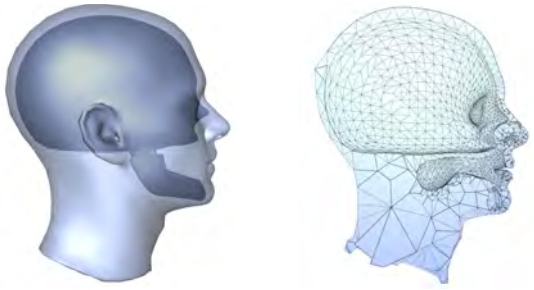

Figure 3: Geometric Model – neutral shape $\mathbf{S}_0$ with embedded skull and jaw meshes (left) and cross-sectional view of the corresponding tetrahedral mesh $\mathbf{X}_0$, passing through the mouth cavity (right).

4.2. Material Model for Animation

When facial muscles are activated, the soft tissue beneath the skin stiffens. In order to reproduce this effect in our simplified model, we augment each blendshape with a *blendmaterial*

$$\mathbf{P} = \mathbf{P}_0 + \mathbf{B_P}\mathbf{w_P}, \tag{3}$$

where $\mathbf{P}_0$ defines the per-element default material parameters, $\mathbf{B_P}$ encodes the expression-specific parameter offsets, and $\mathbf{w_P}$ is a corresponding weight vector. Although not a requirement, simplifying the formulation by assuming that the material changes linearly with shape, i.e., $\mathbf{w_P} = \mathbf{w}$, produced good results in all our experiments.

The material properties can be painted by an artist either as a texture or directly on the mesh. This is very appealing to artists as they are used to work with shape dependent maps, such as albedo textures or normal maps. These maps are then propagated to the volumetric structure as described in Section 6. An illustration of shape-dependent material properties is shown in Fig. 4.

The combination of temporally blending volume and material properties in the same way as blending surface shape allows artists to add simulation to existing blendshape animations with minimal overhead. In the next section we will discuss how to correctly blend these properties without adversely affecting simulation or animation.

5. Blendshape-Driven Simulation

Using the physical face model described in the previous section, we now turn to the question of how to enrich blendshape animations with secondary motion and other physical effects. We begin

Figure 4: Blendmaterials – we propose to use spatio-temporally varying materials as a means of modeling changes in physical properties caused, for example, by muscle activation. Material properties can be painted per blendshape on the surface of the mesh. Here we show areas of higher stiffness in red.

by describing our simulation model, then explain how to incorporate spatio-temporal material and rest shape changes without inducing spurious dynamics.

5.1. Simulation

Elastic Potential. We use a standard finite element approach for simulating facial tissue. We compute the total elastic energy W of the facial tissue from elemental contributions U as

$$W(\mathbf{X},\mathbf{x},\mathbf{P}) = \sum_e U(\mathbf{X}|_e,\mathbf{x}|_e,\mathbf{P}|_e), \tag{4}$$

where $\mathbf{x}$ and $\mathbf{X}$ are the nodal positions in the deformed and undeformed configurations, respectively, and $\mathbf{P}$ are per-element material parameters. We further denote $|_e$ as the restriction of vertices and parameters to the ones incident to element e. It is important to note that every frame of the animation defines a different rest configuration $\mathbf{X}$, computed as a linear combination of per-pose element meshes according to Eq. 2. Similarly, every animation frame defines a different material configuration $\mathbf{P}$, computed as a linear combination of per-pose materials according to Eq. 3.

While many choices for the elastic potential U are compatible with our approach, we choose an extended St. Venant-Kirchhoff material with an altered second term for better inversion recovery [PDA03] and define

$$U(\mathbf{X},\mathbf{x},\mathbf{P}) = \mu[\mathrm{tr}(\mathbf{E}(\mathbf{X},\mathbf{x}))]^2 + \frac{\lambda}{2}\left(\frac{V(\mathbf{x})}{V(\mathbf{X})} - 1\right)^2, \tag{5}$$

where $\mathbf{P} = [\mu,\lambda]^T$ are the Lamé constants that govern stiffness and volume preservation, $\mathbf{E} = \frac{1}{2}\left(\mathbf{F}^T\mathbf{F} - \mathbf{I}\right)$ denotes the Green strain tensor with deformation gradient $\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}}$, and V denotes the volume of a given configuration and $\mathrm{tr}(\cdot)$ the trace operator.

Dynamic Simulation. The dynamics of the facial tissue are governed by Newton's second law of motion,

$$\mathbf{M}\mathbf{a} + \nabla_{\mathbf{x}} W(\mathbf{X},\mathbf{x},\mathbf{P}) = \mathbf{f}_{\text{ext}}, \tag{6}$$

where $\mathbf{M}$ is the mass matrix and $\mathbf{a}$ collects all nodal accelerations.

Assuming, for now, that the rest configuration $\mathbf{X}$ and parameters $\mathbf{P}$ remain unchanged, we can integrate the equations of motion using, e.g., the first-order implicit Euler scheme. To this end, we express the nodal velocities at the next time step as

$$\mathbf{v}_n(\mathbf{x}_n) = \frac{\mathbf{x}_n - \mathbf{x}_p}{h}, \tag{7}$$

where $\mathbf{x}_n$ and $\mathbf{x}_p$ are nodal positions at the next and previous time step, respectively, and h denotes the time step size. Similarly, we define nodal accelerations as

$$\mathbf{a}_n(\mathbf{x}_n) = \frac{\mathbf{v}_n(\mathbf{x}_n) - \mathbf{v}_p}{h} = \frac{\mathbf{x}_n - \mathbf{x}_p}{h^2} - \frac{\mathbf{v}_p}{h}, \tag{8}$$

resulting in a system of nonlinear equations

$$\mathbf{M}\mathbf{a}_n(\mathbf{x}_n) + \nabla_{\mathbf{x}} W(\mathbf{X},\mathbf{x}_n,\mathbf{P}) = \mathbf{f}_{\text{ext}}, \tag{9}$$

which we can solve for the unknown positions $\mathbf{x}_n$. In practice, we follow Martin et al. [MTGG11] and minimize an equivalent dynamic objective

$$f_{\text{dynamic}}(\mathbf{x}_n) = \frac{h^2}{2}\mathbf{a}_n^T\mathbf{M}\mathbf{a}_n + W(\mathbf{X},\mathbf{x}_n,\mathbf{P}) - \mathbf{x}_n^T\mathbf{f}_{\text{ext}}, \tag{10}$$

which assumes the external forces are conservative. This dynamic objective can be altered to handle non-conservative forces as described by Gast et al. [GSS*15] and Narain et al. [NOB16].

5.2. Rest Shape and Material Changes

Our approach relies on changes to the rest state and material properties of the simulation mesh in order to control the motion of the facial tissue. A central challenge that we face is that changes to the rest configuration $\mathbf{X}$, if done naïvely, will induce spurious forces that can manifest as visually disturbing oscillations as demonstrated in Fig. 5. In order to prevent such artifacts, we assume that the inertial forces remain invariant when changing the rest pose and update the deformed configuration in order to maintain dynamic balance.

Dynamic Rebalancing. In order to formalize this idea, assume that the rest configuration and parameters change from $\mathbf{X}_p$ to $\mathbf{X}_n$ and $\mathbf{P}_p$ to $\mathbf{P}_n$, respectively. We first perform a dynamic time step using the previous configuration $\mathbf{X}_p$ and $\mathbf{P}_p$, resulting in intermediate positions $\mathbf{x}'_n$. Next, we compute the inertial force $\mathbf{f}_{\text{inertial}} = \mathbf{M}\mathbf{a}_n(\mathbf{x}'_n)$. At this point, the inertial forces are in balance with the elastic and other external forces. However, simply updating the rest pose and the material parameters would violate this balance and induce artifacts. Instead, we seek to compute corresponding changes in the deformed configuration such that dynamic balance is maintained. This *dynamic rebalancing* amounts to solving a static problem,

$$\mathbf{f}_{\text{inertial}} + \nabla_{\mathbf{x}} W(\mathbf{X}_n,\mathbf{x}_n,\mathbf{P}_n) = \mathbf{f}_{\text{ext}}, \tag{11}$$

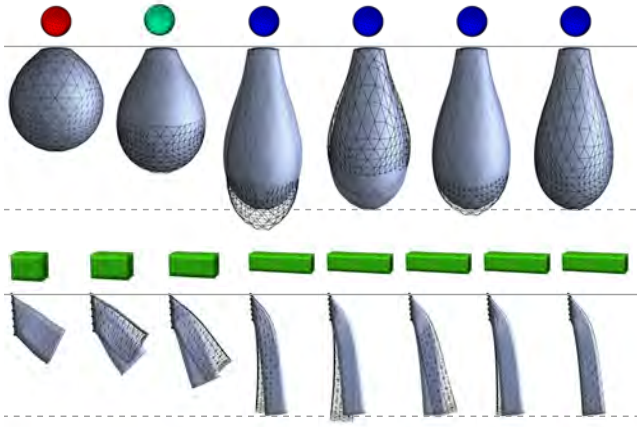

Figure 5: Changing material parameters (top row) or rest shape (bottom row) over time naïvely induces spurious forces to the simulation, leading to disturbing oscillations (wireframe). Our proposed dynamic rebalancing approach allows to change these properties over time without adverse effects (blue shaded).

in order to obtain the final positions $\mathbf{x}_n$. In practice, we do so by minimizing the objective

$$f_{\text{static}}(\mathbf{x}_n) = \mathbf{x}_n^T \mathbf{f}_{\text{inertial}} + W(\mathbf{X}_n, \mathbf{x}_n, \mathbf{P}_n) - \mathbf{x}_n^T \mathbf{f}_{\text{ext}}. \tag{12}$$

The full simulation is summarized as pseudo code in Alg. 1. The proposed update scheme supports changes to both shape and material properties over time without introducing spurious forces. It also guarantees that, in the absence of any external forces, the facial animation will exactly match the input animation as created by the artist.

Algorithm 1 Blendshape-Driven Simulation

Require: Initial state $\mathbf{x}_p = \mathbf{X}_p = \mathbf{X}(0)$, $\mathbf{P}_p = \mathbf{P}(0)$, $\mathbf{v}_p = \mathbf{0}$
while Simulating **do**
  Step time $t = t + h$
  Blend volumes $\mathbf{X}_n = \mathbf{X}_0 + \mathbf{B}_\mathbf{X}\mathbf{w}(t)$
  Blend materials $\mathbf{P}_n = \mathbf{P}_0 + \mathbf{B}_\mathbf{P}\mathbf{w}(t)$
  Solve $\mathbf{x}'_n = \text{argmin}_\mathbf{x} f_{\text{dynamic}}(\mathbf{x})$ with $\mathbf{X} = \mathbf{X}_p$ and $\mathbf{P} = \mathbf{P}_p$
  Update velocities $\mathbf{v}_n(\mathbf{x}'_n)$ and accelerations $\mathbf{a}_n(\mathbf{x}'_n)$
  Compute inertial forces $\mathbf{f}_{\text{inertial}} = \mathbf{M}\mathbf{a}_n$
  Solve $\mathbf{x}_n = \text{argmin}_\mathbf{x} f_{\text{static}}(\mathbf{x})$
  Step quantities $\mathbf{X}_p = \mathbf{X}_n$, $\mathbf{P}_p = \mathbf{P}_n$, $\mathbf{x}_p = \mathbf{x}_n$, $\mathbf{v}_p = \mathbf{v}_n$
end while

Finally, in order to handle self-collisions and contact, we add standard penalties of the form $\frac{1}{2}kd^2$ to our potential W, where d denotes the penetration depth between two intersecting surface triangles, and k is a corresponding stiffness coefficient.

6. Model Initialization

We now provide details of our initialization process required for constructing the face model described in Section 4. Looking first at the geometry, we cover the creation of a blendvolume basis, followed by our intuitive method to initialize the blendmaterial basis.

6.1. Blendvolume Basis

Our method takes a neutral mesh $\mathbf{S}_0$ as input, along with a blendshape basis $\mathbf{B}_\mathbf{S}$ with corresponding skull and per-shape jaw bones. In order to create the finite element volume $\mathbf{X}_0$ we tetrahedralize [Si06] the neutral shape $\mathbf{S}_0$, filling the space between the outer surface and the bones.

In order to enrich animations created on the blendshape rig, we require a blendvolume rig $\mathbf{B}_\mathbf{X}$ that is in direct shape correspondence with $\mathbf{B}_\mathbf{S}$. To this end, we deform the neutral volume (i.e. element mesh) $\mathbf{X}_0$ into each facial expression defined by $\mathbf{S}_0 + \mathbf{B}^i_\mathbf{S}$, and store the positional offsets as the columns of $\mathbf{B}_\mathbf{X}$. Several deformation approaches exist for creating the per-expression element meshes. One of the most obvious choices would be a volumetric Laplacian deformation [ZHS*05], but Laplacian formulations do not prevent inversion of individual elements [SKPSH13]. Since the resulting element meshes will be combined to serve as rest configurations for our simulations, inversion-free configurations are essential. For this reason, we use simulation to transfer the neutral volume to each of the facial expressions.

Directly simulating from the rest configuration $\mathbf{X}_0$ into each expression pose is very challenging, since, by definition, the blendshapes are the extremal poses of the shape space. For robust deformation, we compute the difference vectors $\mathbf{d}_i$ between face surface and corresponding nodes of $\mathbf{X}_0$ and incrementally stiffen a set of spring potentials in static solves of the form

$$f_{\text{static}}(\mathbf{X}_n) = W(\mathbf{X}_p, \mathbf{X}_n, \mathbf{P}_0) - \sum_i \frac{1}{2} k \mathbf{d}_i^2 \,, \tag{13}$$

where we initialize the previous rest pose $\mathbf{X}_p$ with $\mathbf{X}_0$ and update $\mathbf{X}_p$ with $\mathbf{X}_n$ after convergence of a static step. During this initial deformation, we deliberately avoid calculating direct internal constraints for nodes that are adjacent to bones in order to allow the tissue to slide over the bones when appropriate, providing enough degrees of freedom to match the facial expression surface mesh exactly. Furthermore, we deliberatly reduce the volume preservation term in Eq. 5 by setting λ to 100 for all static solves to allow for volume change where necessary due to muscle bulging. Note that we keep weak volume preservation by setting $V(\mathbf{X})$ to $V(\mathbf{X}_0)$ to safeguard against degenerate element configurations. Once converged, the element mesh conforms to the facial expression but may not align internally with the bones. To obtain the final inversion-free expression configuration, we project these inner boundary nodes to the closest surface point on the bones, then perform a second static solve with these additional constraints.

6.2. Blendmaterial Basis

As mentioned previously, muscles can locally change the physical material properties of facial tissue, for instance becoming stiffer when muscles tighten. These changes are very local, requiring a model that supports spatio-temporal changes in material parameters. In our case, these material changes are encoded in a blendmaterial basis $\mathbf{B_P}$, which is also in perfect shape correspondence with $\mathbf{B_S}$.

Initializing per-shape spatially-varying material parameters, even for a simple volume like ours, can be a very tedious task. Our system allows complete artistic control over the expression-dependent material properties by offering a simple and intuitive interface for *painting* properties on the blendshape mesh surfaces. For each expression, an artist is able to paint distinct regions, e.g. in different colors, and set parameters directly. We then solve the heat equation on our element mesh to diffuse the parameters throughout the volume. Surface painting is a familiar and intuitive interface for artists who use it for tasks such as texture and material painting for rendering. Other methods, such as volume painting or even sculpting muscles regions and transferring the corresponding material parameters to our volume, are valid alternatives.

7. Implementation Details

In this section we cover details on how to apply "partial" dynamic rebalancing for artistically blending back in spurious forces if desired, continuously interpolating between full rebalancing and no rebalancing. We also discuss the treatment of teeth, how to deal with excessive resolution rigs, minimize our objectives, and set material parameters.

Partial Dynamic Rebalancing. If desired by an artist, spurious forces induced by blended rest shapes or material properties can be dialed in with a single parameter $\alpha \in [0,1]$. When rebalancing, we aim to preserve the inertial forces $\mathbf{f}'_{\text{inertial}} = \mathbf{M}\mathbf{a}_n(\mathbf{x}'_n)$ after a dynamic step with no configurational changes present (i.e. $\mathbf{X} = \mathbf{X}_p$ and $\mathbf{P} = \mathbf{P}_p$). Without rebalancing, we could perform dynamic steps directly on the next configuration $\mathbf{X}_n$ and $\mathbf{P}_n$, leading to inertial forces $\mathbf{f}''_{\text{inertial}} = \mathbf{M}\mathbf{a}_n(\mathbf{x}''_n)$ with $\mathbf{x}''_n$ denoting the positions that minimize objective f_{dynamic}. Hence, we can dial in a fraction of the spurious forces by minimizing the static objective f_{static} with $\mathbf{f}_{\text{inertial}} = \alpha\mathbf{f}'_{\text{inertial}} + (1-\alpha)\mathbf{f}''_{\text{inertial}}$. By setting $\alpha = 1$ we perform a full dynamic rebalancing step while $\alpha = 0$ does not correct for any spurious forces (see Fig. 6 for an illustration).

Mouth Cavity, Jaw and Teeth Collision Surfaces. To prevent the lips from collapsing inwards, we add teeth-shaped collision surfaces and attach them to either the upper or lower bones for correct transformations during posing. To guarantee intersection-free rest shape configurationsof our blendvolume we detect and resolve collisions using simulation. The jaw is placed manually for each expression by an artist. While the jaw locations may not be accurate in this case, we found this approximate positioning to be sufficient in practice.

Embedded Simulation. Production-level blendshape rigs commonly collect differential data for 100k vertices or more. However, for enriching these rigs with secondary motion effects, this level of detail is excessive and unnecessary. We therefore decimate our highly detailed surface rigs (50-150k vertices) to about 5k vertices. Decimation takes place on the neutral expression $\mathbf{S}_0$ and is then propagated to the complete basis $\mathbf{B_S}$ using Laplacian deformation [SCOL*04]. Simulation (Sec. 5) and initialization (Sec. 6) are then performed on this embedded rig with up to 9k nodes. Finally, resulting deformed configurations are mapped back to the high-resolution surface using the low-resolution vertex positions as constraints to a Laplacian system.

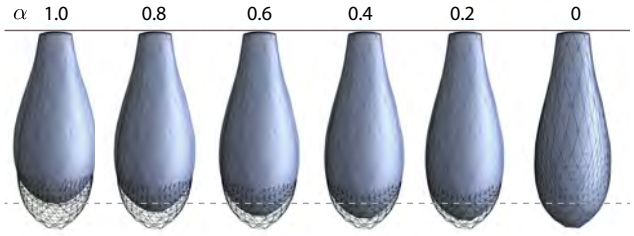

Figure 6: Partial rebalancing for the third frame in Fig. 5 for various values of α (top). By interpolating from $\alpha = 1$ (left) to $\alpha = 0$ (right), our method (blue surface) can dial in dynamics going from a fully rebalanced solution to exactly matching the dynamics created by additional spurious forces (wireframe).

Performance. To minimize the objectives in Eq. 10 and Eq. 12 we use a standard Newton method with analytical gradients and Hessians [NW06], which requires approximately 10 iterations until convergence for the examples shown in this paper. We use a time step size of $1/300$ seconds and each iteration of Alg. 1, including collision detection, takes up to 4 seconds on an Intel Core i7 @ 3.2 GHz. As we target post-production scenarios, such as recombining body motion and blendshape facial animation, our application is less time-critical but requires high-quality simulations. Alternatively, our objectives could be minimized using faster solvers, such as the ADMM solver adapted for physical simulation by Narain et al. [NOB16].

Material Parameters. We use a mass density of 1100 kg/m^3 and set the Lamé constants in Eq. 5 to the values 3000 *Pa* and 2500 *Pa*, respectively, and vary μ between 3000 *Pa* and 20000 *Pa*. For all human faces λ is set to 2500 *Pa*, and for the Fantasy Creature it varies from 1000 *Pa* to 2500 *Pa*.

8. Results

We have used our enrichment technique to add secondary motion effects to a total of three faces of both realistic (see Figs. 1, 8 and 10) and fantasy characters (see Figs. 7 and 9), in addition to the illustrative examples shown in Figs. 5 and 6.

Updating Rest Shape and Materials. As one can clearly see in the accompanying video and Fig. 5, rest shape or material updates lead to changes of the elastic potential W and in turn, induce spurious forces in dynamic simulations, if not explicitly prevented. In Fig. 5 (top), we soften the material (from left to right), leading to vertical oscillations if inertial force preservation is not enforced. In this visualization, the wireframe mesh is the result of naïve simula-

Figure 7: Our Fantasy Creature undergoes extreme changes in both rest shape and stiffness: (from left to right) volume and material drastically shrink and stiffen during a rapid head-shaking sequence, illustrating the full potential of our method.

tion, compared to the blue shaded result of our dynamic rebalancing method. When elongating an elastic bar (bottom, from left to right), we observe horizontal oscillations if dynamic rebalancing is not active.

Human Faces. In Fig. 1 we illustrate a neutral expression undergoing rapid side-to-side head motion and fast up-down running motion. The additional secondary dynamics are visible especially in the lip region. Additionally, we show a smile expression undergoing the same running motion, where the secondary effect is much more subtle due to the stiffened tissue caused by the smile expression (shown on the far right). We encourage the reader to refer also to the accompanying video in order to fully appreciate these dynamic effects, along with other expressions and motions.

In an example on a second character (see Fig. 8), we enhance a running sequence with facial expressions changing from squinting on one side to squinting on the other (A, top and bottom rows) with plausible secondary dynamics. The necessity of spatio-temporal material changes is clearly illustrated in the columns on the right: (B) a spatially uniform material and (C) a spatially-varying but temporally constant material do not lead to physically-plausible effects. Only when using spatio-temporally varying materials, we can achieve the expressiveness needed for the desired effects (D).

Fantasy Characters. To demonstrate the versatility of our approach, we apply our method to a fantasy creature made of mud. In Fig. 7 (and towards the end of the accompanying video) we illustrate extreme shape, volume, and stiffness changes. Shapes and materials change from a "wet and soft" to a "dry and stiff" state. As we illustrate in Fig. 9, our method is capable of correcting lip self-collisions as well as handling other contacts, constraining secondary motion effects to the physically feasible set.

Evaluation. We evaluate the benefit of our dynamic rebalancing method on a simple speech sequence in Fig. 10. In the absence of head motion and external forces, our method reproduces the input animation precisely, while traditional dynamic simulation overshoots the animated lip motion and lags behind the speech sequence (note that this is even more evident in the accompanying video). Avoiding such spurious motions is particularly important in speech, where delay or incompatibility between the voice and the mouth motion can be very disturbing for the viewer.

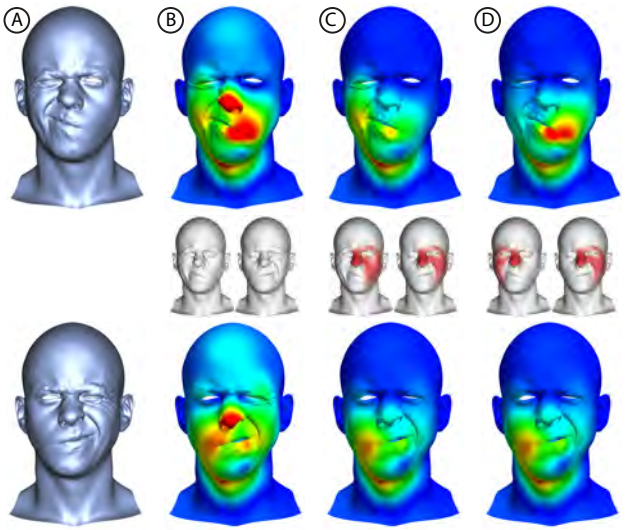

Figure 8: Enriching a running sequence with an expression change from a right (A, top) to a left squint (A, bottom): when using a homogeneous material (B, middle row) we observe implausibly large deformations in stiff regions of the face visualized on a rainbow color map with small displacements in blue and large displacements in red. If we spatially vary the material but keep it temporally constant (C), we observe plausible secondary motion for the left (bottom) but not for the right squint (top). By spatio-temporally varying materials we achieve the desired effects on both expressions (D).

9. Conclusion

In this work, we introduce a method to add physical simulation to traditional blendshape rigs. As an alternative to complex muscle-based anatomical models, we combine basic artist-directed animation with the benefits of physical simulation for creating secondary motion. Central to our approach is a method to adapt the rest shape during simulation, according to the input animation, without inducing unwanted forces. This ensures that, in the absence of external forces or rigid motion our result will exactly match the input animation. Furthermore, we introduce an approach to allow spatio-

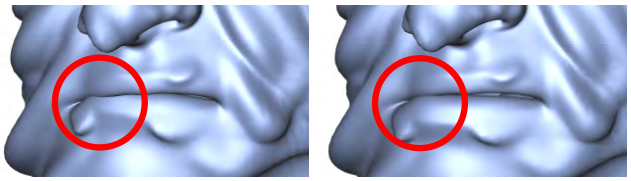

Figure 9: Handling contact and collisions is an integral component of our approach, illustrated here on a self-collision of the upper and lower lip: (left) with no collision handling we observe a self-intersection in proximity of the lip corner while the intersection is correctly resolved when collision handling is active (right).

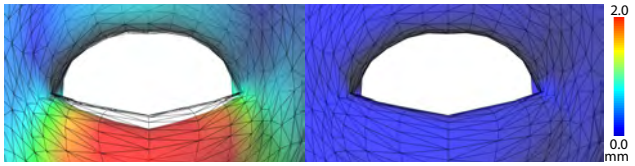

Figure 10: A speech sequence with no external forces or head motion: (left) inertia and spurious forces created by varying the rest shape cause conventional simulation (surface) to deviate from the input animation (wireframe), leading to a disturbing mismatch between audio and visuals. The results of our method (surface) match the input animation precisely (right). The color encodes the error.

temporally changing material parameters during simulation, in order to account for locally changing tissue stiffness caused by muscle activation. Finally, our framework is very simple and intuitive to control, as defining expression-dependent material parameters is as simple as painting on the blendshape geometry. As a result, our method adds realistic physical effects to facial animations, which we demonstrate on several examples of both humans and a fantasy creature.

Limitations. Our approach is not without limitations. Our volumetric abstraction cannot faithfully mimic true muscle activations, however, from an animation point of view, our results show very realistic and plausible motions. Processing time with our method increases over traditional blendshape animation, and one drawback is that if the input animation is edited after computing the secondary motion, the simulation must be computed again for the whole sequence (at least from the point of editing onwards). Determining fast approximate updates to the secondary motion corresponding to primary animation edits would be an interesting area for future work.

Acknowledgements

We wish to thank our actors Maria Cabral and Steven McDonagh. We would also like to thank Maurizio Nitti for designing our Fantasy Creature and his help with rendering as well as Fabian Hahn for sharing his code with us.

References

[BHB*11] BEELER T., HAHN F., BRADLEY D., BICKEL B., BEARDSLEY P., GOTSMAN C., SUMNER R. W., GROSS M.: High-quality passive facial performance capture using anchor frames. *ACM Trans. Graphics (Proc. SIGGRAPH) 30* (2011), 75:1–75:10. 2

[BHPS10] BRADLEY D., HEIDRICH W., POPA T., SHEFFER A.: High resolution passive facial performance capture. *ACM Trans. Graphics (Proc. SIGGRAPH) 29* (2010), 41:1–41:10. 2

[BJTM08] BARBARINO G., JABAREEN M., TRZEWIK J., MAZZA E.: Physically based finite element model of the face. In *LNCS* (2008), pp. 1–10. 2

[BMWG07] BERGOU M., MATHUR S., WARDETZKY M., GRINSPUN E.: Tracks: Toward directable thin shells. *ACM Trans. Graph. 26*, 3 (July 2007). 3

[BSC16] BARRIELLE V., STOIBER N., CAGNIART C.: Blendforces: A dynamic framework for facial animation. In *Eurographics* (2016). 2

[CBF16] CONG M., BHAT K. S., FEDKIW R. P.: Art-Directed Muscle Simulation for High-End Facial Animation. In *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation* (2016), Kavan L., Wojtan C., (Eds.), The Eurographics Association. 2

[CBJ*15] CONG M., BAO M., JANE L. E., BHAT K. S., FEDKIW R.: Fully automatic generation of anatomical face simulation models. In *Proc. SCA* (2015), pp. 175–183. 2

[CMT*12] COROS S., MARTIN S., THOMASZEWSKI B., SCHUMACHER C., SUMNER R., GROSS M.: Deformable objects alive! *ACM Trans. Graphics (Proc. SIGGRAPH) 31*, 4 (2012), 69:1–69:9. 3

[FJA*14] FYFFE G., JONES A., ALEXANDER O., ICHIKARI R., DEBEVEC P.: Driving high-resolution facial scans with video performance capture. *ACM Trans. Graphics 34*, 1 (2014), 8:1–8:14. 2

[FLP14] FAN Y., LITVEN J., PAI D. K.: Active volumetric musculoskeletal systems. *ACM Trans. Graph. 33*, 4 (July 2014), 152:1–152:9. 4

[GOM*09] GALOPPO N., OTADUY M. A., MOSS W., SEWALL J., CURTIS S., LIN M. C.: Controlling deformable material with dynamic morph targets. In *Proc. of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (February 2009). 3

[GSS*15] GAST T. F., SCHROEDER C., STOMAKHIN A., JIANG C., TERAN J. M.: Optimization integrator for large time steps. *IEEE Transactions on Visualization and Computer Graphics 21*, 10 (Oct 2015), 1103–1115. 5

[HMT*12] HAHN F., MARTIN S., THOMASZEWSKI B., SUMNER R., COROS S., GROSS M.: Rig-space physics. *ACM Trans. Graphics (Proc. SIGGRAPH) 31*, 4 (2012), 72. 3

[HTC*13] HAHN F., THOMASZEWSKI B., COROS S., SUMNER R., GROSS M.: Efficient simulation of secondary motion in rig-space. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2013), SCA '13. 3

[IKNDP16] ICHIM A.-E., KAVAN L., NIMIER-DAVID M., PAULY M.: Building and animating user-specific volumetric face rigs. In *SCA* (2016). 3

[KH12] KLAUDINY M., HILTON A.: High-detail 3d capture and non-sequential alignment of facial performance. In *3DIMPVT* (2012). 2

[KHS01] KÄHLER K., HABER J., SEIDEL H.-P.: Geometry-based muscle modeling for facial animation. In *Graphics Interface* (2001). 2

[KKA05] KONDO R., KANAI T., ANJYO K.-I.: Directable animation of elastic objects. In *Proc. SCA* (2005), pp. 127–134. 3

[LAR*14] LEWIS J. P., ANJYO K., RHEE T., ZHANG M., PIGHIN F., DENG Z.: Practice and theory of blendshape facial models. In *EG 2014 - State of the Art Reports* (2014). 2

[LTW95] LEE Y., TERZOPOULOS D., WATERS K.: Realistic modeling for facial animation. In *SIGGRAPH '95* (1995). 2

[LXB16] LI Y., XU H., BARBIČ J.: Enriching triangle mesh animations with physically based simulation. *IEEE Transactions on Visualization and Computer Graphics PP*, 99 (2016), 1–1. 3

[MHSH10] MITHRARATNE K., HUNG A., SAGAR M., HUNTER P.: An efficient heterogeneous continuum model to simulate active contraction of facial soft tissue structures. In *Proc. IFMBE* (2010). 2

[MTGG11] MARTIN S., THOMASZEWSKI B., GRINSPUN E., GROSS M.: Example-based elastic materials. *ACM Trans. Graph. 30*, 4 (2011). 3, 5

[MWF*12] MA W.-C., WANG Y.-H., FYFFE G., CHEN B.-Y., DEBEVEC P.: A blendshape model that incorporates physical interaction. *Computer Animation and Virtual Worlds 23*, 3-4 (2012), 235–243. 2

[MZS*11] MCADAMS A., ZHU Y., SELLE A., EMPEY M., TAMSTORF R., TERAN J., SIFAKIS E.: Efficient elasticity for character skinning with contact and collisions. *ACM Trans. Graph. 30*, 4 (July 2011), 37:1–37:12. 3

[NOB16] NARAIN R., OVERBY M., BROWN G. E.: ADMM $\supseteq$ projective dynamics: Fast simulation of general constitutive models. In *Proc. SCA* (2016), pp. 21–28. 5, 7

[NW06] NOCEDAL J., WRIGHT S. J.: *Numerical Optimization*, 2nd ed. Springer, New York, 2006. 7

[PDA03] PICINBONO G., DELINGETTE H., AYACHE N.: Non-linear anisotropic elasticity for real-time surgery simulation. *Graphical Models 65*, 5 (2003), 305 – 321. 5

[PMRMB15] PONS-MOLL G., ROMERO J., MAHMOOD N., BLACK M. J.: Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics, (Proc. SIGGRAPH) 34*, 4 (Aug. 2015), 120:1–120:14. 3

[RP07] RÖHRLE O., PULLAN A.: Three-dimensional finite element modelling of muscle forces during mastication. *J. Biomech. 40*, 15 (2007). 2

[SCOL*04] SORKINE O., COHEN-OR D., LIPMAN Y., ALEXA M., RÖSSL C., SEIDEL H.-P.: Laplacian surface editing. In *Proceedings of the EUROGRAPHICS/ACM SIGGRAPH Symposium on Geometry Processing* (2004), ACM Press, pp. 179–188. 7

[Si06] SI H.: Tetgen - a quality tetrahedral mesh generator and three-dimensional delaunay triangulator. *Weierstrass Institute for Applied Analysis and Stochastic, Berlin, Germany* (2006). 6

[SKPSH13] SCHÜLLER C., KAVAN L., PANOZZO D., SORKINE-HORNUNG O.: Locally injective mappings. *Computer Graphics Forum (proceedings of EUROGRAPHICS/ACM SIGGRAPH Symposium on Geometry Processing) 32*, 5 (2013), 125–135. 6

[SNF05] SIFAKIS E., NEVEROV I., FEDKIW R.: Automatic determination of facial muscle activations from sparse motion capture marker data. In *ACM Trans. Graphics* (2005), vol. 24, pp. 417–425. 2

[TW90] TERZOPOULOS D., WATERS K.: Physically-based facial modeling, analysis, and animation. *J. of Vis. and Comp. Anim. 1*, 2 (1990), 73–80. 2

[WM15] WARBURTON M., MADDOCK S.: Physically-based forehead animation including wrinkles. *Comput. Animat. Virtual Worlds* (2015). 2

[XB16] XU H., BARBIČ J.: Pose-space subspace dynamics. *ACM Trans. on Graphics (SIGGRAPH 2016) 35*, 4 (2016). 3

[YSZ09] YOU L., SOUTHERN R., ZHANG J. J.: Adaptive physics–inspired facial animation. In *Motion in Games*. 2009, pp. 207–218. 2

[ZHS*05] ZHOU K., HUANG J., SNYDER J., LIU X., BAO H., GUO B., SHUM H.-Y.: Large mesh deformation using the volumetric graph laplacian. In *ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), SIGGRAPH '05, ACM, pp. 496–503. 6