

# Animatomy: an Animator-centric, Anatomically Inspired System for 3D Facial Modeling, Animation and Transfer

Byungkuk Choi  
litlpoet@gmail.com  
Wētā Digital  
New Zealand

Haekwang Eom  
heom@wetafx.co.nz  
Wētā Digital  
New Zealand

Benjamin Mouscadet  
bmouscadet@wetafx.co.nz  
Wētā Digital  
New Zealand

Stephen Cullingford  
stevejc@wetafx.co.nz  
Wētā FX  
New Zealand

Kurt Ma  
kma@wetafx.co.nz  
Wētā Digital  
New Zealand

Stefanie Gassel  
sgassel@wetafx.co.nz  
Wētā Digital  
New Zealand

Suzi Kim  
skim@wetafx.co.nz  
Wētā Digital  
New Zealand

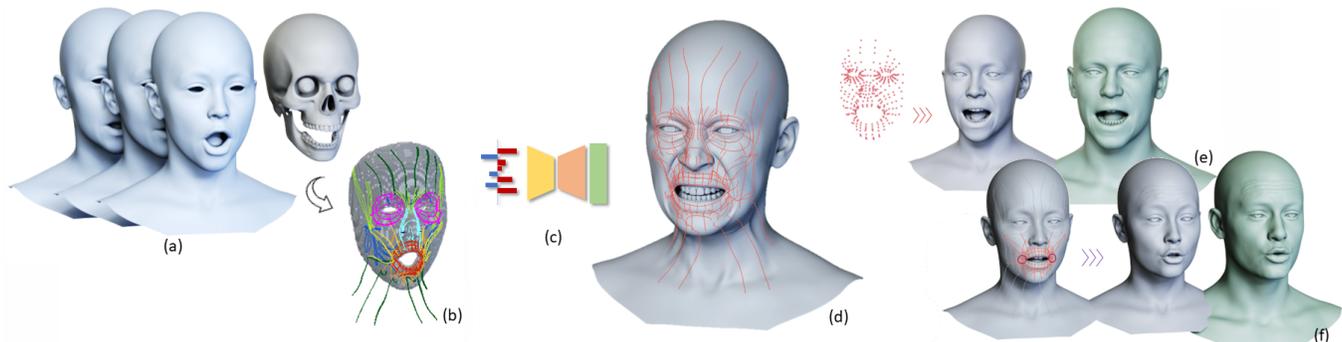
Andrew Moffat  
amoffat@wetafx.co.nz  
Wētā Digital  
New Zealand

Millicent Maier  
mmaier@wetafx.co.nz  
Wētā Digital  
New Zealand

Marco Revelant  
mrevelant@wetafx.co.nz  
Wētā FX  
New Zealand

Joe Letteri  
letteri@wetafx.co.nz  
Wētā FX  
New Zealand

Karan Singh  
karan@dgp.toronto.edu  
Wētā Digital  
New Zealand  
University of Toronto  
Canada



**Figure 1:** *Animatomy* is a high-end facial animation pipeline built on a novel face parameterization using contractile muscle curves. We present the construction and fitting of the muscle curves to a set of dynamic 3D scans for an actor (a), using a passive muscle simulation (b). Muscle contractions (strains) parameterize these scans and are used to learn a manifold of plausible facial expressions (c). The strains, in turn, control skin deformation (d) and readily transfer expression from an actor to characters. In production, the strains can be animated by performance capture (e) and animator interaction (f). ©Wētā FX.

## ABSTRACT

We present *Animatomy*, a novel anatomic+animator centric representation of the human face. Present FACS-based systems are plagued with problems of face muscle separation, coverage, opposition, and redundancy. We, therefore, propose a collection of muscle fiber curves as an anatomic basis, whose contraction and relaxation provide us with a fine-grained parameterization of human facial expression. We build an end-to-end modular deformation architecture using this representation that enables: automatic optimization of

the parameters of a specific face from high-quality dynamic facial scans; face animation driven by performance capture, keyframes, or dynamic simulation; interactive and direct manipulation of facial expression; and animation transfer from an actor to a character. We validate our facial system by showing compelling animated results, applications, and a quantitative comparison of our facial reconstruction to ground truth performance capture. Our system is being intensively used by a large creative team on *Avatar: The Way of Water*. We report feedback from these users as qualitative evaluation of our system.

## CCS CONCEPTS

• Computing methodologies → Mesh models; Animation; Graphics systems and interfaces.

## KEYWORDS

Facial modeling, facial expression, facial animation, data-driven animation, animation system, deformation, motion capture

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

SA '22 Conference Papers, December 6–9, 2022, Daegu, Republic of Korea

© 2022 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-9470-3/22/12...\$15.00

<https://doi.org/10.1145/3550469.3555398>

**ACM Reference Format:**

Byungkuk Choi, Haekwang Eom, Benjamin Mouscadet, Stephen Cullingford, Kurt Ma, Stefanie Gassel, Suzi Kim, Andrew Moffat, Millicent Maier, Marco Revelant, Joe Letteri, and Karan Singh. 2022. *Animatomy: an Animator-centric, Anatomically Inspired System for 3D Facial Modeling, Animation and Transfer*. In *SIGGRAPH Asia 2022 Conference Papers (SA '22 Conference Papers)*, December 6–9, 2022, Daegu, Republic of Korea. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3550469.3555398>

**1 INTRODUCTION**

The innate human ability to both perceive and convey the subtlest of facial nuance makes animating an engaging anthropomorphic face arguably the most challenging aspect of character animation. The greater the rendered realism, the smaller the margin for animated imperfection before the character irreversibly plunges into the Uncanny Valley [MacDorman et al. 2009]. We present *Animatomy*, a novel anatomy+animator centric 3D face modeling and animation pipeline in use on *Avatar: The Way of Water*.<sup>1</sup>

Over the past few decades, the Facial Action Coding System (FACS) [Ekman and Friesen 1978] has become a popular baseline representation for facial animation. While FACS has allowed a level of standardization and interoperability across facial rigs, FACS was designed from a psychological standpoint to capture voluntary, distinguishable snapshots of facial expression, and has clear limitations when applied to computer animation [Deng and Noh 2008; Seymour 2019]. FACS Action Units (AUs) have well-known problems with *anatomic fidelity* (AUs that combine the action of multiple facial muscles or do not involve facial muscles at all), *localization and animation control* (AUs that can be redundant, opposing in action, strongly co-related, or mutually exclusive), and *facial deformation* (AUs only approximate the complex shape deformations of a hinged jaw and flexible lips) [Parke and Waters 2008]. In practice, animators address these limitations ad-hoc and, as needed, augment FACS with large, unwieldy instances of specific and corrective deformer.

Instead, we draw inspiration from *Mimic* [Hjortsjö and Salisbury 1970], an older, anatomically grounded language of facial expression. Mimic muscles are imagined as fibers, primarily around or radially emanating from facial orifices (mouth and eyes), but also attaching superficially to the skin, such as at the nasolabial furrow. Facial expressions manifest as contractions of these mimic muscles. *Animatomy* muscles are conceptually similar but explicitly motivated by our goals of facial performance capture for high-end animation: *a data-driven workflow; anatomic plausibility; animator control; deformation fidelity; and expression transfer*.

We constructed our muscle-based parameterization by inverse simulating a representative set of skeletal face muscles embedded within a tetrahedralized flesh mask, similar to [Sifakis et al. 2005; Srinivasan et al. 2021]. We then selected an artist-curated, minimal yet meaningful set of muscle fiber curves to capture muscle contraction along pennation directions. Sheet muscles have fiber bundles that can contract selectively, and these are captured using multiple parallel fiber curves. We observed that, compared to contraction, the bend and twist of muscle fiber curves were minimal. We thus capture both the bending of muscle and the volumetric squash and stretch orthogonal to the contracting muscle, by introducing curves

orthogonal to the muscle fibers, and attached to soft tissue in the flesh mask. While this deviates from a strict anatomic mapping of curves to muscle fibers, it facilitates an animator-friendly representation of a deformable face as a homogeneous collection of contractile 3D curves [Singh and Fiume 1998]. A unitless *strain* value captures the change in length of an activated muscle curve relative to its length in a neutral state. Our *Animatomic* face expression is thus parameterized by a vector of strains corresponding to the 178 muscle fiber curves we define for a human face (Fig. 1).

*Overview.* Related work (§2) is followed by principles guiding our system design (§3) and the creation of an actor face-rig using 3D scanning, muscle curve fitting, and simulation (§4). We then present our data-driven computation of a muscle strain-based face expression manifold and our strain-to-skin deformation algorithm (§5, §6). We detail system components needed for high-end film production, built atop our *Animatomic* face model (§7): animation using performance capture; interactive face manipulation tools; and transfer of expressive facial animation from the actor to other characters. Our evaluation (§8) is both quantitative (parameter ablation and ground-truth comparison) and qualitative (professional animator critique). We also discuss limitations and future work (§9).

*Contribution.* We present a novel curve-based face representation, central to a complete face animation pipeline. Our system is larger than the sum of its parts, designed to meet a judicious combination of anatomic, artistic, and performance capture needs.

**2 PRIOR WORK**

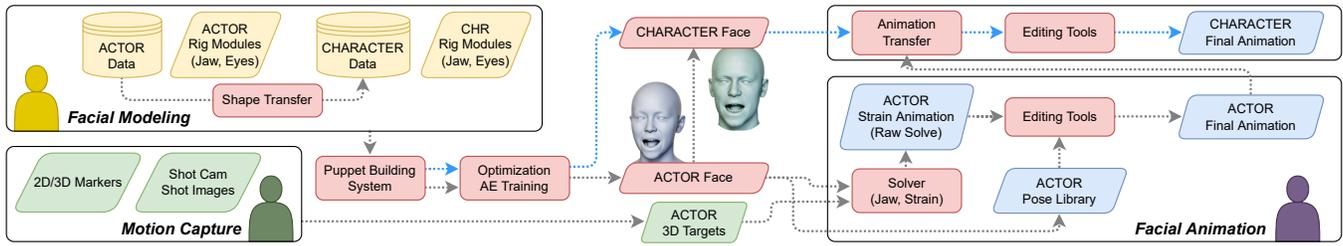
We broadly classify many decades of relevant research in facial animation [Parke and Waters 2008] as follows.

*Blendshapes and Facial Rigging.* Blendshapes are an artist-sculpted set of target faces, often aligned with FACS AUs, used extensively in film and games over decades. Facial expressions are produced as a linear combination of blendshapes [Lewis et al. 2014]. While blendshapes give artists full control over the face, modeling and animating a realistic face require a significant artistic skill. Adding corrective shapes is common and high-end blendshape rigs are unwieldy with many redundant, correlated, and mutually exclusive shapes (e.g., 946 blendshapes used to animate ‘Gollum’ in the *Lord of the Rings* [Raitt 2004]). In contrast, our muscle curves provide a 178-parameter facial representation that is rather compact.

The interactive manipulation of the high-dimensional space of blendshape weights is also difficult, motivating research in control layouts [Kim and Singh 2021] and inverse weight computation to fit direct manipulation of the face [Lewis and Anjyo 2010] or sketched facial features [Cetinaslan et al. 2015]. Further, blendshapes neither guarantee the plausibility of facial expressions nor span all plausible expressions [Abdrashitov et al. 2020]. *Animatomy* admits interaction with muscle curves, brush-based direct manipulation, and a parametric manifold of plausible expressions using a muscle simulation to fit a comprehensive corpus of dynamic face scans.

Facial rigging generally refers to the various controls, including skeletal joints (e.g., to control a jaw and eyeballs), and other deformers, to manipulate facial expressions [Orvalho et al. 2012]. Several deep-learning-based rig approximations were proposed to replace the complexity of facial rigs [Bailey et al. 2020; Song et al. 2020]. In

<sup>1</sup>While a film trailer is publicly viewable, we are unable to show footage or tests using film assets before the film’s release later this year.



**Figure 2: Animatomy provides a complete face animation pipeline to meet a sensible combination of different user groups’ anatomic, artistic, and performance capture needs. Essential techniques presented in this paper are marked in red. ©Wētā FX.**

contrast, our muscle curves complement skeletal deformation and operate as part of a typical animation chain.

*Data-driven and Morphable Models.* Statistical modeling of 3D faces ranges from early Principal Component Analysis (PCA) based morphable models to modern techniques that employ deep learning [Egger et al. 2020]. A realistic 3D facial model can be reconstructed from images [Feng et al. 2021; Kemelmacher-Shlizerman and Seitz 2011; Luo et al. 2021], video [Grassal et al. 2021; Lombardi et al. 2018; Suwajanakorn et al. 2014, 2015], and scanned data [Booth et al. 2016; Dai et al. 2020; Li et al. 2017; Paysan et al. 2009]. These techniques generally begin with a template face mesh and deform it to fit the dataset while optimizing model parameters. Recent approaches also aim to model shape and appearance simultaneously from large corpora [Egger et al. 2020].

The FLAME [Li et al. 2017] model covers a range of face modeling contexts, and the approach has been extended to support controllable deformation [Feng et al. 2021] and expression realism [Abdrashitov et al. 2020; Schwartz et al. 2020]. Recently, non-linear deep face models [Chandran et al. 2020, 2022b; Jiang et al. 2019; Li et al. 2020; Ranjan et al. 2018] improve facial expressivity, and local patch based approaches [Chandran et al. 2022a; Wu et al. 2016] enable accurate facial reconstruction from small training datasets. While these models are typically not amenable to animator editing, we draw inspiration from them in our formulation of a muscle strain-based expression manifold and skin deformation.

*Muscle-based Physical Model.* High-resolution physics and anatomy-based simulations can typically be used to account for collisions, external forces, and dynamic behavior, trading accuracy for speed and animation friendliness. Barrielle et al. [2016] parameterize a face animation with the external forces acting upon it while others [Bao et al. 2019; Ichim et al. 2017; Sifakis et al. 2005] integrate simulations of muscles or their activations within their models. We chose to use simulations only to create a large dataset with which we trained an approximated deformation model of muscle fibers.

### 3 SYSTEM DESIGN PRINCIPLES AND GOALS

In consultation with face modelers, riggers, and animators, we design *Animatomy* (Fig. 2) to reconcile and balance anatomic, artistic, and performance animation design requirements.

- *Artistic Control:* While the performance capture of physical actors principally animates *Animatomy* faces, we require the ability of animators to edit the results and to hand-craft animation in scenarios where physical capture is difficult. Our system is thus

designed as a set of deformation nodes integrated into production pipelines in a commercial animation system like Maya. Thus, our face representation must allow forward (inside-out), and inverse (outside-in) type animation control of facial expressions.

- *Anatomically Grounded Representation:* We aim to design a system for anthropomorphic faces strongly tied to a human face’s musculature. Beyond a generic representation of 3D geometry, we thus desire a parameterization that explicitly embodies the anatomy of the human face. While many surface and volume muscle representations exist, we require that the simulated behavior of the muscle is captured for efficient kinematic control.
- *Data-driven Workflow:* We need an end-to-end automated system based on a face representation that generates high-resolution faces from artist-curated 3D scans. All system components aim to optimize the face to conform to an actor’s input data.
- *Transferable Animation:* The visual effects pipeline for motion-captured character animation relies on a two-step process. A digital double of the actor is first made and animated to match the captured performance with high fidelity before transferring the animation to the character, with minimal user oversight.

*Animatomy* reconciles the above needs by using 3D curves to model muscle fibers and curves orthogonal to muscle fibers to capture muscle volume and bending (Fig. 1(b, d)). The curves are passively simulated within a volumetric musculature to conform to input 3D skin data. The contractile behavior of the simulated curves provides an effective low-dimensional parameterization of 178 muscle strains that transfer well across diverse face morphology. Animation transfer across diverse but humanoid characters using a shared muscle strain space is effortless and accurate, requiring less user correction than any prior system known to us.

### 4 DATA PREPARATION

We build our face model using machine learning and optimization processes trained on a significant amount of curated ground truth data. In particular, we need corresponding sequences of actor meshes, joint transformations, and simulated muscle fibers to build a production facial rig, as shown in Fig. 3.

*Actor Dynamic Scans.* We reconstruct 3D shapes of an actor’s face using photogrammetry (3DF Zephyr [2022]) to model the rest state of skin, eyes, teeth, and maximal ranges for unassisted jaw opening, protrusion, and lateral movement. Our scanning follows the LightStage [Debevec et al. 2000] setup, and includes post-processing

the raw scans. A skull is fit inside the scanned model by approximating tissue depth data with medical/forensic pegs placed on the skin and skull, and varying peg-length based on actor age, gender, ethnicity, and Body Mass Index (BMI) [De Greef et al. 2006]. The actor is asked to perform FACS actions, a comprehensive range of emotions, and utter various phonemes and Harvard sentences [Rothausser 1969], that result in a reconstructed 3D facial dataset: comprising 80 motion clips ( $\approx 7,000$  frames) in which facial actions (FACS+emotions) and speech are present in equal proportion. We get a temporally-aligned mesh sequence using sequential registration (R3DS Wrap [2022]), and head movement removed by rigid stabilization [Beeler and Bradley 2014].

*Eyes and Jaw Alignment.* We solve mandible movement using a complex non-linear jaw rig [Zoss et al. 2018] by running a least-squares optimization to fit the jaw rig (§5.5) to the mesh sequence. The resulting animation is verified by observing teeth alignment against images captured from each camera. We compare soft tissue depth between the skin surface and mandible to verify frames where the surrounding soft tissue occludes the teeth. The resulting 3D transformations represent the mandibular movement for the performed action or speech. The transformations are also used to reconstruct the inner mouth occluded or shadowed in the original photogrammetry. Our eye model approximates the actor’s sclera, cornea, and iris. Eye gaze direction is adjusted in each frame of the mesh sequence by rotating the eyeballs so that the iris model aligns with the limbal ring and pupil, visible on the images captured from each camera. Multiple camera angles are used to verify the alignment and account for light refracted by the cornea. Well-aligned rotations for the eyes allow us to correct for minor deformation artifacts in the surrounding geometry. A small frontal translation (eye bulge) is tied to eye rotation to enhance eye realism.

*Actor Dynamic Muscle Simulation.* To parameterize the fiber-based muscle model, we first define a tetrahedral volume discretizing the soft tissue of the face in the rest-pose. This volume conforms to the skin and the bones similar to [Ichim et al. 2017; Sriniwasan et al. 2021]. A passive, quasi-static simulation is then performed to this volume for whole scan sequences with skin vertices and the skull enforced as positional constraints for tetrahedral elements [Lesser et al. 2022]. We simulate 135K tetrahedrons constrained with multiple positional, sliding, and collision constraints. We compute a barycentric embedding for control points for the



**Figure 3: Data example: actor mesh, volumetric representation, muscle fibers with eye and jaw alignment, and all-inclusive model. ©Wētā FX.**

anatomic muscle curves [Zarins 2018], within the rest pose tetrahedral volume. For each simulated frame, we use their barycentric coordinates to extract simulated muscle curves.

## 5 ANATOMICALLY INSPIRED FACIAL MODEL

*Animatomy* deformation architecture is inspired by the FLAME [2017] and uses a similar vertex-based skinning approach with corrective shapes, in our case with  $N = 85,000$  vertices, and  $K = 3$  joints (jaw and eyeballs). The FLAME system relies on PCA weights to drive facial animation, reducing shape and expression spaces to their principal components. Our model, in contrast, is based on muscle strains, which provide an anatomically meaningful and animator-friendly basis to represent and control facial expressions.

Function  $M(\vec{\theta}, \vec{\gamma}) : \mathbb{R}^{|\vec{\theta}| \times |\vec{\gamma}|} \rightarrow \mathbb{R}^{3N}$  describes our model, mapping a vector describing pose (jaw and eyes transformations)  $\vec{\theta} \in \mathbb{R}^{|\vec{\theta}|}$  and expression (encoded by muscle strains  $\vec{\gamma} \in \mathbb{R}^{|\vec{\gamma}|}$ ), to  $N$  vertices. As shown in Fig. 4, the model consists of a neutral mesh  $\vec{T} \in \mathbb{R}^{3N}$  (unposed and expressionless); the corresponding rest-pose vector  $\vec{\theta}^*$ ; corrective pose blendshapes  $B_P(\vec{\theta}; \mathcal{P}) : \mathbb{R}^{|\vec{\theta}|} \rightarrow \mathbb{R}^{3N}$  to correct pose deformations that cannot be produced by linear blend skinning (LBS); strain-driven blendshapes  $B_E(\vec{\gamma}; \mathcal{E}) : \mathbb{R}^{|\vec{\gamma}|} \rightarrow \mathbb{R}^{3N}$  capturing facial expressions; and a strain-jaw autoencoder  $AE_\Phi(\vec{\gamma}, \vec{\theta}) : \mathbb{R}^{|\vec{\gamma}| + |\vec{\theta}|} \rightarrow \mathbb{R}^{|\vec{\gamma}|}$  (parameterized by its weights  $\Phi$ ), to enforce non-linear muscle strain behavior.  $M$  is formulated as:

$$M(\vec{\theta}, \vec{\gamma}) = W(T_P(\vec{\theta}, \vec{\gamma}), J, \vec{\theta}, \mathcal{W}),$$

$$T_P(\vec{\theta}, \vec{\gamma}) = \vec{T} + B_P(\vec{\theta}; \mathcal{P}) + B_E(AE_\Phi(\vec{\gamma}, \vec{\theta}_{jaw}); \mathcal{E}).$$

$T_P$  denotes the addition of pose and expression displacements to the neutral mesh (§5.2, §5.3) and  $W(T_P, J, \vec{\theta}, \mathcal{W})$  is a skinning function [Loper et al. 2015] to transform the vertices of  $T_P$  around joints  $J \in \mathbb{R}^{3K+3}$ , linearly smoothed by skinning weights  $\mathcal{W} \in \mathbb{R}^{N \times K}$ .

### 5.1 Muscle Features (Strains)

From each muscle curve of length  $s$ , we derive a unitless real-valued strain  $\gamma = (s - \bar{s}) / \bar{s}$ , where  $\bar{s}$  is the length of the muscle curve at the neutral frame (rest-pose). Strain in our context, is thus a deviation from a muscle curve’s rest-pose length. A negative/positive strain is thus a muscle contraction/relaxation relative to its rest-pose tension. The strain values for all the  $|\vec{\gamma}|$  muscles at frame  $t$  are grouped together in a vector  $\vec{\gamma}^{(t)}$ , and we define  $\Gamma = \{\vec{\gamma}^{(t)} | t \leq T\}$  for a given sequence of  $T$  frames.

### 5.2 Eyes and Jaw Base Deformation

Let  $R(\vec{\theta}) : \mathbb{R}^{|\vec{\theta}|} \rightarrow \mathbb{R}^{9K+3}$  be a function from a pose vector  $\vec{\theta}$  (corresponding to jaw and eyeballs rig controls) to a vector containing the concatenated elements of all the corresponding rigid transformation matrices ( $\mathbb{R}^{3 \times 3}$  rotations for eyeballs, and  $\mathbb{R}^{3 \times 4}$  rigid transformation for the jaw). Let also  $\vec{\theta}^*$  be the rest pose, corresponding to the neutral frame. The pose blendshape function is then defined as

$$B_P(\vec{\theta}; \mathcal{P}) = \sum_{k=1}^{9K+3} (R_k(\vec{\theta}) - R_k(\vec{\theta}^*)) P_k,$$

where  $R_k(\vec{\theta})$  and  $R_k(\vec{\theta}^*)$  denote the  $k$ -th element of  $R(\vec{\theta})$  and  $R(\vec{\theta}^*)$ , respectively. The vector  $P_k \in \mathbb{R}^{3N}$  describes the corrective vertex displacements from the neutral pose activated by  $R_k$ , and the pose

space  $\mathcal{P} = [P_1, \dots, P_{9K+3}] \in \mathbb{R}^{3N \times (9K+3)}$  is a matrix with all corrective pose blendshapes.

### 5.3 Strain-to-Skin Deformation Matrix

Linear blendshapes model the strain-to-skin deformation matrix to produce skin expressions as

$$B_E(\vec{\gamma}; \mathcal{E}) = \sum_{i=1}^{|\vec{\gamma}|} E_i \gamma_i = \mathcal{E} \vec{\gamma},$$

where  $\mathcal{E} = [E_1, \dots, E_{|\vec{\gamma}|}] \in \mathbb{R}^{3N \times |\vec{\gamma}|}$  denotes the optimized strain-to-skin deformation basis. These blendshapes are driven only by the strain vector and not by pose. §5.4 describes how and why the autoencoder first processes this strain vector.

### 5.4 Strain Autoencoder

Though the underlying concept of muscle elongation or contraction might be intuitive, driving a facial expression with the strain vector is not always straightforward. We utilize an autoencoder (AE) to assist artists by constraining the strain vector to remain within the boundaries of plausible face animation, and we call this space the expression manifold. Human interpretation defines plausibility here, and this manifold is thus estimated with a curated sampling of multiple facial expressions and their corresponding strain vectors.

The purpose of the AE is to perform a projection onto this space without being restrictive. It should naturally support the animators in directing any desired facial expression as long as they remain within the manifold while preventing mistakes and incorrect manipulations from producing uncanny expressions. To achieve this behavior, we train a small-scale neural network of three encoding and three decoding layers, which first project the input vector into a latent space before reconstructing the strain values. The latent space is about two times smaller in dimension than the input space to apply a lossy compression and force the AE to exhibit the projecting behavior. The strain vector lacks a structure we could leverage in our network; hence we rely solely on dense (fully-connected) layers and add cELU and Tanh activation functions for non-linearity.

Because of the natural predisposition of specific unrelated facial muscles to be activated in unison, like the lip corner puller causing the eyes to squint, the AE tends to learn some—and replicates these—regional contaminations, which are undesirable for artistic control. In order to mitigate this issue and enforce some form of localized influence (changing a muscle should not affect remote parts of the face), we partition the strain vector  $\vec{\gamma}$  into two regions: the muscles related to the upper (resp. lower) part of the face are grouped in the vector  $\vec{\gamma}_u$  (resp.  $\vec{\gamma}_l$ ). There is no overlap between the regions; hence  $|\vec{\gamma}| = |\vec{\gamma}_u| + |\vec{\gamma}_l|$ . Similarly, the autoencoder is comprised of two separate autoencoder networks with the structure described above, one for each partition of  $\vec{\gamma}$ . Note that we refer to the autoencoder in a singular form due to the shared properties between the two.

The AE responsible for processing the lower vector  $\vec{\gamma}_l$  (resp. upper) is conditioned with the jaw pose (resp. eyes pose). The rigid jaw transformation is part of the input but not of the output and only serves to stabilize the AE with its ground-truth nature. The output of both AE networks is concatenated together to produce the final strain vector, which then drives the expression blendshape.

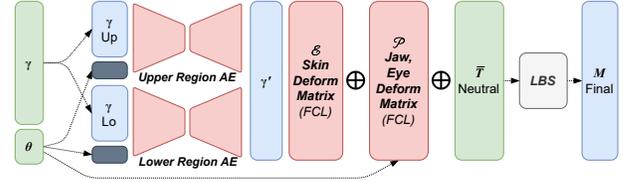


Figure 4: *Animatomy* facial model. AE: autoencoder, FCL: fully connected layer, LBS: linear blend skinning. ©Wētā FX.

### 5.5 Jaw Proxy Model

A facial rig to animate accurate jaw animation relies on a complex non-linear function [Zoss et al. 2018], which maps the 3D jaw controls to the applied 6D rigid transformation. It is embedded deep inside the jaw rig and cannot easily be formulated analytically. Because we use gradient methods to train and solve our model (respective §6 and §7.1), end-to-end differentiability is indispensable. Consequently, we approximate the mapping with an easily trainable and infinitely differentiable Radial Basis Function (RBF) network  $\chi$ .

The rigid jaw transformation to map to is a 6D vector (translation and axis-angle rotation). We adopt the Gaussian kernel as the RBF. Let  $\mu$  and  $\sigma$  be its parameters and  $\mathbb{R}^3$  be its input space:

$$\forall p, \mu, \sigma \in \mathbb{R}^{3 \times 3 \times 1}, g_{\mu, \sigma}(p) = \exp\left(-\sigma^2 \|p - \mu\|^2\right).$$

Given the parameters  $\{\psi_i, \mu_i, \sigma_i | i \leq N\} \in \mathbb{R}^6 \times \mathbb{R}^3 \times \mathbb{R}$  and the number of neurons  $M = 50$ , we have the RBF network  $\chi$ :

$$\forall p \in \mathbb{R}^3, \chi(p) = \frac{\sum_{i=1}^M \psi_i g_i(p)}{\sum_{i=1}^M g_i(p)}, \text{ with } g_i = g_{\mu_i, \sigma_i}.$$

## 6 MODEL TRAINING

The unknown parameters of our model are the LBS weights  $\mathcal{W} = \{\omega_{ik}\} \in \mathbb{R}^{N \times K}$ , the pose correction blendshapes  $\mathcal{P} = \{P_k | k \leq K\}$ , the strain-to-skin expression deformation matrix  $\mathcal{E} = \{E_s | s \leq |\vec{\gamma}|\}$ , and the AE weight parameters  $\Phi$ , which we train successively in this order on a dataset of around 7,000 corresponding ground truth meshes  $\vec{V}$ , poses  $\vec{\theta}$ , and strain vectors  $\vec{\gamma}$ .

We first train the LBS weights to minimize  $\|\vec{V} - W(\vec{T}, J, \vec{\theta}, \mathcal{W})\|$ . This is not enough to account for all pose-related deformations, so we train the pose correction blendshape on the residual error by minimizing  $\|W^{-1}(\vec{V}) - \vec{T} - \mathcal{B}_P(\vec{\theta}, \mathcal{P})\|$  of the *unposed mesh*. Note, our LBS function  $W$  is invertible because the  $K$  weight maps partition the skin mesh vertices. Once all pose-related deformations are computed, the residual error is captured by the expression blendshape, which minimize  $\|\widehat{V} - \mathcal{B}_E(\vec{\gamma}, \mathcal{E})\|$ , where  $\widehat{V}$  is the unposed mesh whose norm we minimized in the previous step. At this stage, we optimize  $\mathcal{E}$  and fine-tune  $\vec{\gamma}$  to have the lowest error (§A.3).

The loss functions we minimize typically also have regularization terms that reduce the influence of the parameters over specific outputs, primarily to reduce the amount cross-talk. The training losses are further detailed in §A. When the three training steps are done, we have  $\vec{V} = W(\vec{T} + \mathcal{B}_P(\vec{\theta}, \mathcal{P}) + \mathcal{B}_E(\vec{\gamma}, \mathcal{E}))$  with high accuracy.

To perform a projection as described in §5.4, we train the AE to fit on the training strains. This learns a latent representation of the manifold implicitly defined by our ground truth samples

(all plausible expressions). Consequently, we preserve the strains within this implicit space and correct those that do not conform to the manifold. Thus, the equality  $AE_{\Phi}(\gamma) \approx \gamma$  holds true for plausible expressions only. In addition, the neutral shape is critical for the animators, and its corresponding strain vector  $\gamma_0$  has to be perfectly preserved by the AE. Hence we must enforce the equality  $AE_{\Phi}(\gamma_0) = \gamma_0$  with a simple implementation trick (see §A.4).

## 7 USE IN FILM PRODUCTION

### 7.1 Performance Capture Driven Animation

We animate the actor’s digital double that matches the facial expressions with very high fidelity and then transfer the animation to the character model. We find the optimal pose and strain inputs for each given frame, as shown in Fig. 5. Given the 3D-tracked motion capture facial markers, we first build target meshes that explain the given markers the best. Mapping facial meshes to marker space is a projection; hence its inverse is under-constrained and ill-defined, but we leverage our training dataset [Li et al. 2013; Seol et al. 2016] to find the best pseudo-inverse for our use case (see §B.1).

Our facial model is end-to-end differentiable by design; hence we solve this inverse problem with gradient descent as a two-step process. The first step (§B.2) computes the pose inputs  $\vec{\theta}$  (eyes rotations and jaw controls), and our jaw proxy model (§5.5) plays a central role in computing the gradients. The solver at this stage captures all skin deformations correlated to the joints’ rigid transformations. We achieve this by keeping the strains constant equal to the neutral vector  $\gamma_0$ . The resulting poses can then be projected into camera space and compared to the shot images. Here artists can visually validate the results, and manually re-align the teeth, if further accuracy is required. The second step (§B.3) fits the strains to capture all residual skin deformation. Implausible expressions are avoided by adding a regularization term to keep strain values within the space that the AE was trained to preserve. In the absence of artist validation, multiple alternating iterations of the two steps can yield better results. Our solver assumes coherent sequences instead of individual frames to enforce temporal coherence. When solving long shots ( $\geq 1K$  frames), we avoid memory issues by partitioning the sequence into tractable sub-sequences, blended afterward.

### 7.2 Tools for Animator Interaction

The inherent ability of a compact set of muscle strains to fully describe any facial expression illustrates its potential to represent an animated face. Nevertheless, it is a significant paradigm shift for animators, experienced in using FACS-blendshapes to animate the face. To facilitate adoption, we developed a set of brush-based, animator-centric tools, to interface with our facial system in *Maya*. The tools operate either on the strains (muscle manipulation; inside-out) or the mesh (direct manipulation [Lewis and Anjyo 2010]; outside-in). They interact locally with the facial model, using a radial, brush-influence area around the mouse cursor (see Fig. 1(f)). The length of the stroke can modulate the strength of the brush, and a symmetric mode optionally mirrors the effect of the stroke, either bilaterally, or radially around the mouth and eyes.

**7.2.1 Muscle Animation.** Our *Anatomy* tool-set operates at various levels of abstraction. At the finest level users have complete

control over individual muscle strains. Here muscle curves displayed beneath the skin surface, can be contracted/ elongated using brush strokes, to interactively deform the mesh.

**7.2.2 Strain-based Pose Library.** Analogous to FACS-blendshape control, the pose brush tool provides high level expression control. A pose is defined by a set of associated strain values. Animators can curate a pose library with typical expressions for each facial rig. Selected strains of these poses can be dialed in/out towards their absolute values, or relative to the current facial expression.

**7.2.3 Direct Manipulation.** At an abstract level, the direct manipulation tool allows artists to sculpt mesh vertices directly to desired expressions. The brush strokes here, directly deform mesh vertices to provide a target skin mesh. Similar to the inverse optimization presented in §7.1 and §B (but without a temporal term), we compute strain values as well as jaw+eye controls that best-fit the target mesh. Our GPU-based implementation provides near real-time performance ( $\approx 15$  fps), suitable for interaction, but costlier than the forward deformations using strain-based brushes.

### 7.3 Actor-to-Character Transfer

Our pipeline foresees a different model and rig for the actor and the character. To maximize the parity between an actor and a character face in the animation transfer stage, we strategically design our character training process to share the corresponding actor’s underlying muscle behavior. To achieve this, we perform shape transfer before the character training stage to perfectly align transferred skin meshes with the actor’s dataset in the correct order. Then, instead of considering an independent set of the character’s muscle curves in the training stage, we use the actor’s strain values and strain autoencoder to optimize the strain-to-skin blendshapes (§5.3, §5.4). Consequently, the final character facial model will have the shared strain autoencoder identical to the actor (see Fig. S4).

**7.3.1 Shape Transfer.** We utilize cage-based transfer [Orvalho et al. 2008] to transfer the character’s skin from the actor. We first calculate a correspondence matrix using RBF between the actor and the character neutral shape at cage resolution. When the proportion of specific regions of the actor’s and character’s face is similar, we can compute multiple such correspondence matrices. Handling the eye and jaw regions separately for example, using user-defined weight maps, can allow more accurate expression transfer for those parts.

We also fit the actor’s jaw rig to the character and use it to compensate for the deviation in teeth topography and skull anatomy. We compute LBS weights for the jaw joints from an example scan of the actor and use the same UV-based weights for the character. Given matching jaw set-ups, we *unpose* the jaw rotation to handle the substantial movement around the mandible region that is not appropriately captured by the cage displacements. This *unposing* stage is also helpful in aligning the rigid inner mouth regions.

To fine-tune specific expressions and increase the precision and fidelity often required by art direction, users can add a small number of guide shapes (Fig. 6). Given approximately ten pairs of matching actor and character guide shapes (similar to corrective shapes), we express each actor scan as a linear combination of the neutral and the guide shapes. We then refine the corresponding character



Figure 5: The actor’s performance (gray) is solved for 340 markers from validation ground-truth expressions (blue). ©Wētā FX.

shape by layering a linear combination of delta displacements computed from the character’s transferred guide shapes with the same weights (see §C of the supplemental material for further technical detail). Multiple *Na’vi* characters in *Avatar: The Way of Water* were successfully transferred using no more than 20 guide shapes.

**7.3.2 Animation Transfer.** As an actor and a corresponding character pair share their strain space from the training stage, transferring animation from one to the other is trivial. We can directly connect the strain and control parameters between the actor and the character for real-time animation transfer. Note that no expression cloning or re-targeting, commonly used in FACS-based systems [Kim et al. 2021; Ribera et al. 2017; Seol et al. 2012], is necessary.



Figure 6: Actor, character without guide shapes and with fixed mouth via guide shapes (from left to right). ©Wētā FX.

## 8 EVALUATION

### 8.1 Quantitative Evaluation

**8.1.1 Strain and Latent Space Dimensions.** The choice of 178 strains was an artist-curated trade-off between reconstruction accuracy, anatomic completeness and animator-control. More than 200 strains add redundant complexity for artists and less than 130 strains prevent accurate reconstruction (Fig. S7), especially around the mouth. We also experimented with the latent space size of our autoencoders, settling on half of the input size, to balance reconstruction accuracy of training data and tolerance for implausible facial expressions.

**8.1.2 Animatomy vs. FACS-based model.** We compared our 178 strain *Animatomy* solver (see §7.1), against a FACS-blendshape solver (a variant of [Lewis and Anjyo 2010] using 200 target shapes chosen from the training data). *Animatomy* reconstructed unseen ground-truth expressions better than the FACS-based solution, as shown by the mean-squared vertex error (maximum vertex error in parenthesis) for both models below.

(unit: mm)	Shot 1	Shot 2	Shot 3
<i>Animatomy</i>	0.378 (2.751)	0.239 (2.096)	0.257 (2.255)
FACS	0.521 (2.794)	0.390 (2.111)	0.490 (3.139)

### 8.2 Artist Feedback

*Animatomy* is presently in intensive production use on over a dozen speaking humanoid characters of varying proportions for *Avatar: The Way of Water*. Below we briefly distill free-form feedback (guided by questions in the supplement §D) solicited from four seasoned experts on all aspects of our system.

#### 8.2.1 Actor Face Model and Rig (Puppet) Creation (§4).

- + Muscle strains simulated on dynamic scans produce a highly accurate reconstruction of a scanned actor, superior to FACS-blendshape systems with comparable data; Muscle curves are a good spatial proxy for selection, visualization, deformation, and reinforce artist knowledge of facial anatomy; Modular workflow aids in model troubleshooting and allows selective improvement.
- Scan protocol of  $\approx 80$  clips can be time-consuming for celebrity actors and computationally expensive to process; Dynamic scans are temporally unique to actors and hard to re-pose.

#### 8.2.2 Performance Capture Driven Face Animation (§7.1).

- + Extremely high-quality 3D face reconstruction, faithful to actor performance, in nuance, subtlety and timing; Significantly less manual fix-up better than existing FACS-blendshape systems.
- Requires animator familiarization with an anatomical strain basis; Animators may have to stray off the strain manifold, as actors can be more expressive on set than during a 3D scan session.

#### 8.2.3 Interactive Face Manipulation (§7.2).

- + Brush-based tools provide users a familiar direct manipulation interface, while automatically computing muscle strains to realize the edited skin; Keyframe editing of muscle strains effective for generating on-model (manifold) animation; Creation from scratch is harder using strains than a pose library, making the strain-based pose library a valuable tool.
- Spatially localized editing of facial expression off-model (manifold) can be difficult to control, if projecting muscle strains back onto the face manifold (§5.4).

#### 8.2.4 Actor to Character Transfer and Animation (§7.3).

- + Strains are robust across a wide range of facial feature shapes and proportions; Low artist effort in authoring guide shapes.
- Consistent and repeatable poses in the actor data are important, to get stable weights for each guide shape.

## 9 DISCUSSION AND CONCLUSION

**Limitations.** Active production use is a strong validation of *Animatomy*, but also exposes limitations. The balance between muscle co-relation and localization on the face remains a challenge. We

developed our muscle strain auto-encoder and strain to skin deformation using fully connected networks to capture global face co-relations as a counterpoint to arbitrarily local blendshape expressions. This can prove problematic for animators looking to make spatially localized edits to the face. While muscle curves provide some intuition for animators, they prefer working with facial poses when quickly blocking animations from scratch. We have thus developed a tool to compute strain values for posed face libraries. Additionally, while actor muscle strains transfer well as-is to characters with significantly different feature proportions, we do assume a morphology similar to the human face.

*Future Work.* As a new production system, there are many avenues for future work including reducing the number of dynamic scans needed to train an actor rig; training actor-rigs using a single 3D scan and a large corpus of film clips of the actor; enabling the muscle curves to be dynamically actuated by muscle impulse; extending our muscle curve model which currently stops below the chin, to extend into the neck and possibly the entire body (currently parameterized by a typical joint skeleton); and investigating other network architectures, such as meshCNNs [Hanocka et al. 2019] to better control both spatial muscle localization, and global neurological co-relation between muscles, when deforming skin.

*Conclusion.* In summary, we have presented a successfully deployed, high-end facial animation system *Animatomy*, that is a departure from FACS-blendshape systems. Our system, based on a novel muscle curve representation of the face, enables fine grained, anatomically plausible animation control, and straightforward transfer of animation from an actor to a virtual character.

## ACKNOWLEDGMENTS

We are thankful to the many people besides the authors of this paper who contributed to *Animatomy*. Luca Fascione, Stuart Adcock, and David Luke were particularly influential in supervising and designing a research-oriented as well as production-friendly facial system. Christoph Sprenger, Muhammad Ghifary, Gergely Klár, Stephen Ward, Yeongho Seol, Tobias Schmidt, and Daniel Lond each contributed multiple years of research to make *Animatomy* a robust system. Matt Penman, Braden Jennings, Alex Telford, Leon Woud, Ben Goldberg, Nivedita Goswami, Sebastian Gassel, Josh Hardgrave, and Matthew Jeng put a tremendous amount of development effort into *Animatomy* to complete large-scale production requirements. We are also thankful for leadership and management support from Tom Buys, Joerg Fluegge, Kenneth Gimpelson, Derrick Auyoung, Dejan Momcilovic, Daniel Hodson, and Julia Jones. Many talented artists have worked closely with the Facial Research team. Thank you to Marco Barbati, Rachel Hydes, Bex Leybourne, and Allison Orr from the Facial Models and Motion departments.

## REFERENCES

- 3Dflow. 2022. *3DF Zephyr*. <http://3dflow.net/3df-zephyr-photogrammetry-software/>
- Rinat Abdrashitov, Fanny Chevalier, and Karan Singh. 2020. Interactive Exploration and Refinement of Facial Expression Using Manifold Learning. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*.
- Stephen W. Bailey, Dalton Omens, Paul Dilorenzo, and James F. O'Brien. 2020. Fast and Deep Facial Deformations. *ACM Trans. Graph.* 39 (2020).
- Michael Bao, Matthew Cong, Stephane Grabli, and Ronald Fedkiw. 2019. High-Quality Face Capture Using Anatomical Muscles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Vincent Barrielle, Nicolas Stoiber, and Cédric Cagniard. 2016. BlendForces: A Dynamic Framework for Facial Animation. *Computer Graphics Forum* 35 (2016).
- Thabo Beeler and Derek Bradley. 2014. Rigid Stabilization of Facial Expressions. *ACM Trans. Graph.* 33 (2014).
- James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David Dunaway. 2016. A 3D Morphable Model Learnt from 10,000 Faces. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Ozan Cetinaslan, Verónica Orvalho, and J.P. Lewis. 2015. Sketch-Based Controllers for Blendshape Facial Animation. In *EG 2015 - Short Papers*.
- Prashanth Chandran, Derek Bradley, Markus Gross, and Thabo Beeler. 2020. Semantic Deep Face Models. In *2020 International Conference on 3D Vision (3DV)*.
- Prashanth Chandran, Loïc Ciccone, Markus Gross, and Derek Bradley. 2022a. Local Anatomically-Constrained Facial Performance Retargeting. *ACM Trans. Graph.* 41 (2022).
- Prashanth Chandran, Gaspard Zoss, Markus Gross, Paulo Gotardo, and Derek Bradley. 2022b. Shape Transformers: Topology-Independent 3D Shape Models Using Transformers. *Computer Graphics Forum* 41 (2022).
- Hang Dai, Nick Pears, William Smith, and Christian Duncan. 2020. Statistical Modeling of Craniofacial Shape and Texture. *International Journal of Computer Vision* 128 (2020).
- Sven De Greef, Peter Claes, Dirk Vandermeulen, Wouter Mollemans, Paul Suetens, and Guy Willems. 2006. Large-scale in-vivo Caucasian facial soft tissue thickness database for craniofacial reconstruction. *Forensic science international* 159 (2006).
- Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. 2000. Acquiring the Reflectance Field of a Human Face. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*.
- Zhigang Deng and Junyong Noh. 2008. *Computer Facial Animation: A Survey*. Springer London.
- Bernhard Egger, William A. P. Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, Christian Theobalt, Volker Blanz, and Thomas Vetter. 2020. 3D Morphable Face Models—Past, Present, and Future. *ACM Trans. Graph.* 39 (2020).
- Paul Ekman and Wallace V. Friesen. 1978. Facial Action Coding System: A Technique for the Measurement of Facial Movement. (1978).
- Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. 2021. Learning an Animatable Detailed 3D Face Model from In-the-Wild Images. *ACM Trans. Graph.* 40 (2021).
- Philip-William Grassal, Malte Prinzler, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. 2021. *Neural Head Avatars from Monocular RGB Videos*.
- Rana Hanocka, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. 2019. MeshCNN: A Network with an Edge. *ACM Trans. Graph.* 38 (2019).
- C.H. Hjortsjö and W.F. Salisbury. 1970. *Man's face and mimic language*.
- Alexandru-Eugen Ichim, Petr Kadlecěk, Ladislav Kavan, and Mark Pauly. 2017. Phace: Physics-Based Face Modeling and Animation. *ACM Trans. Graph.* 36 (2017).
- Zi-Hang Jiang, Qianyi Wu, Keyu Chen, and Juyong Zhang. 2019. Disentangled Representation Learning for 3D Face Shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Ira Kemelmacher-Shlizerman and Steven M. Seitz. 2011. Face Reconstruction in the Wild. In *2011 International Conference on Computer Vision*.
- Joonho Kim and Karan Singh. 2021. Optimizing UI Layouts for Deformable Face-Rig Manipulation. *ACM Trans. Graph.* 40 (2021).
- Seonghyeon Kim, Sunjin Jung, Kwanggyoon Seo, Roger Blanco i Ribera, and Junyong Noh. 2021. Deep Learning-Based Unsupervised Human Facial Retargeting. *Computer Graphics Forum* 40 (2021).
- Steve Lesser, Alexey Stomakhin, Gilles Daviet, Joel Wretborn, John Edholm, Noh-Hoon Lee, Eston Schweickart, Xiao Zhai, Sean Flynn, and Andrew Moffat. 2022. Loki: a unified multiphysics simulation framework for production. *ACM Trans. Graph.* 41 (2022).
- J.P. Lewis, Ken Anjyo, Taehyun Rhee, Mengjie Zhang, Fred Pighin, and Zhigang Deng. 2014. Practice and Theory of Blendshape Facial Models. In *Eurographics 2014 - State of the Art Reports*.
- J.P. Lewis and Ken-ichi Anjyo. 2010. Direct Manipulation Blendshapes. *IEEE Computer Graphics and Applications* 30 (2010).
- Hao Li, Jihun Yu, Yuting Ye, and Chris Bregler. 2013. Realtime Facial Animation with On-the-Fly Correctives. *ACM Trans. Graph.* 32 (2013).
- Ruilong Li, Karl Bladin, Yajie Zhao, Chinmay Chinara, Owen Ingraham, Pengda Xiang, Xinglei Ren, Pratusha Prasad, Bipin Kishore, Jun Xing, and Hao Li. 2020. Learning Formation of Physically-Based Face Attributes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. 2017. Learning a Model of Facial Shape and Expression from 4D Scans. *ACM Trans. Graph.* 36 (2017).
- Stephen Lombardi, Jason Saragih, Tomas Simon, and Yaser Sheikh. 2018. Deep Appearance Models for Face Rendering. *ACM Trans. Graph.* 37 (2018).
- Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2015. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graph.* 34 (2015).

- Huiwen Luo, Koki Nagano, Han-Wei Kung, Qingguo Xu, Zejian Wang, Lingyu Wei, Liwen Hu, and Hao Li. 2021. Normalized Avatar Synthesis Using StyleGAN and Perceptual Refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Karl F. MacDorman, Robert D. Green, Chin-Chang Ho, and Clinton T. Koch. 2009. Too real for comfort? Uncanny responses to computer generated faces. *Computers in Human Behavior* 25 (2009).
- Verónica Orvalho, Pedro Bastos, Frederic Parke, Bruno Oliveira, and Xenxo Alvarez. 2012. A Facial Rigging Survey. In *Eurographics 2012 - State of the Art Reports*.
- Verónica Costa Orvalho, Ernesto Zacur, and Antonio Susin. 2008. Transferring the Rig and Animations from a Character to Different Face Models. *Computer Graphics Forum* 27 (2008).
- Frederic I. Parke and Keith Waters. 2008. *Computer Facial Animation* (2nd ed.). AK Peters Ltd.
- Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. 2009. A 3D Face Model for Pose and Illumination Invariant Face Recognition. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*.
- R3DS. 2022. *Wrap3*. <https://www.russian3dscanner.com/>
- Bay Raitt. 2004. The Making of Gollum. Presentation at U. Southern California Institute for Creative Technologies's Frontiers of Facial Animation Workshop.
- Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. 2018. Generating 3D Faces using Convolutional Mesh Autoencoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Roger Blanco i Ribera, Eduard Zell, J. P. Lewis, Junyong Noh, and Mario Botsch. 2017. Facial Retargeting with Automatic Range of Motion Alignment. *ACM Trans. Graph.* 36 (2017).
- EH Rothaus. 1969. IEEE Recommended Practice for Speech Quality Measurements. *IEEE Transactions on Audio and Electroacoustics* 17 (1969).
- Gabriel Schwartz, Shih-En Wei, Te-Li Wang, Stephen Lombardi, Tomas Simon, Jason Saragih, and Yaser Sheikh. 2020. The Eyes Have It: An Integrated Eye and Face Model for Photorealistic Facial Animation. *ACM Trans. Graph.* 39 (2020).
- Yeongho Seol, J.P. Lewis, Jaewoo Seo, Byungkuk Choi, Ken Anjyo, and Junyong Noh. 2012. Spacetime Expression Cloning for Blendshapes. *ACM Trans. Graph.* 31 (2012).
- Yeongho Seol, Wan-Chun Ma, and J.P. Lewis. 2016. Creating an Actor-Specific Facial Rig from Performance Capture. In *Proceedings of the 2016 Symposium on Digital Production (DigiPro '16)*.
- Mike Seymour. 2019. FACS at 40: Facial Action Coding System Panel. In *ACM SIGGRAPH 2019 Panels (SIGGRAPH '19)*.
- Eftychios Sifakis, Igor Neverov, and Ronald Fedkiw. 2005. Automatic Determination of Facial Muscle Activations from Sparse Motion Capture Marker Data. In *ACM SIGGRAPH 2005 Papers (SIGGRAPH '05)*.
- Karan Singh and Eugene Fiume. 1998. Wires: A Geometric Deformation Technique. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*.
- Steven L. Song, Weiqi Shi, and Michael Reed. 2020. Accurate Face Rig Approximation with Deep Differential Subspace Reconstruction. *ACM Trans. Graph.* 39 (2020).
- Sangeetha Grama Srinivasan, Qisi Wang, Junior Rojas, Gergely Klár, Ladislav Kavan, and Eftychios Sifakis. 2021. Learning Active Quasistatic Physics-Based Models from Data. *ACM Trans. Graph.* 40 (2021).
- Supasorn Suwajanakorn, Ira Kemelmacher-Shlizerman, and Steven M. Seitz. 2014. Total Moving Face Reconstruction. In *Computer Vision - ECCV 2014*.
- Supasorn Suwajanakorn, Steven M. Seitz, and Ira Kemelmacher-Shlizerman. 2015. What Makes Tom Hanks Look Like Tom Hanks. In *2015 IEEE International Conference on Computer Vision (ICCV)*.
- Chenglei Wu, Derek Bradley, Markus Gross, and Thabo Beeler. 2016. An Anatomically-Constrained Local Deformation Model for Monocular Face Capture. *ACM Trans. Graph.* 35 (2016).
- Uldis Zarins. 2018. *Anatomy of Facial Expressions*. Exonicus, Incorporated.
- Gaspard Zoss, Derek Bradley, Pascal Bérard, and Thabo Beeler. 2018. An Empirical Rig for Jaw Animation. *ACM Trans. Graph.* 37 (2018).