An Implicit Physical Face Model Driven by Expression and Style

LINGCHEN YANG, ETH Zurich, Switzerland
GASPARD ZOSS, DisneyResearch|Studios, Switzerland
PRASHANTH CHANDRAN, DisneyResearch|Studios, Switzerland
PAULO GOTARDO, DisneyResearch|Studios, Switzerland
MARKUS GROSS, ETH Zurich, Switzerland and DisneyResearch|Studios, Switzerland
BARBARA SOLENTHALER, ETH Zurich, Switzerland
EFTYCHIOS SIFAKIS, University of Wisconsin Madison, USA
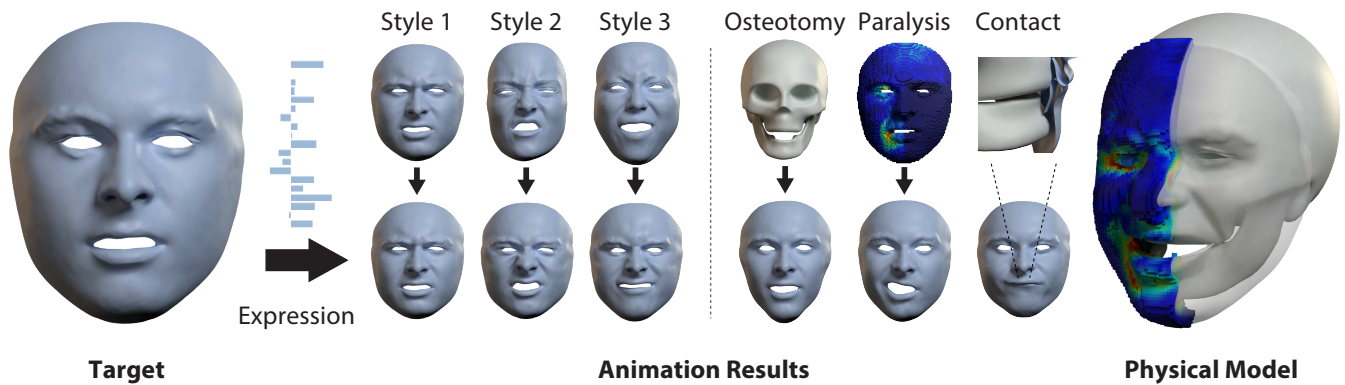DEREK BRADLEY, DisneyResearch|Studios, Switzerland

Fig. 1. Our physical face model, learned from multi-identity data, is driven by expression and style, allowing to synthesize different styles for the same expression. In addition, it supports various physical effects, such as bone reshaping (osteotomy), paralysis, contact and collision handling, etc.

3D facial animation is often produced by manipulating facial deformation models (or rigs), that are traditionally parameterized by expression controls. A key component that is usually overlooked is expression "style", as in, *how* a particular expression is performed. Although it is common to define a semantic basis of expressions that characters can perform, most characters perform each expression in their own style. To date, style is usually entangled with the expression, and it is not possible to transfer the style of one character to another when considering facial animation. We present a new face model, based on a data-driven implicit neural physics model, that can be driven by both expression and style separately. At the core, we present a framework for learning implicit physics-based actuations for multiple subjects simultaneously, trained on a few arbitrary performance capture sequences from a small set of identities. Once trained, our method allows generalized physics-based facial animation for any of the trained identities, extending to unseen performances. Furthermore, it grants control over the animation style, enabling style transfer from one character to another or blending styles of different characters. Lastly, as a physics-based model, it is capable of synthesizing physical effects, such as collision handling, setting our method apart from conventional approaches.

CCS Concepts: • **Computing methodologies → Physical Simulation**; **Neural networks**.

Additional Key Words and Phrases: Facial Expression and Style, Physics-Based Facial Animation, Data-Driven Physical Animation.

1 INTRODUCTION

Facial animation is a key component in computer graphics, especially for video games, visual effects, telepresence and beyond. 3D faces are generally controlled by facial rigs, which offer intuitive parameters to control the facial deformation. A common example is blendshape-based animation [Lewis et al. 2014], where faces are created by combining weighted contributions of different blendshapes. Here, the blendweight vector defines the *expression* of the character. Animation is defined by a sequence of blendweight vectors, often called animation curves, which can be key-framed by artists or reconstructed from motion-capture data. Given different

characters with semantically corresponding blendshape rigs, the same blendweights can be applied to each rig in order to obtain the same expression but performed by the different characters, in their own style, defined by each character's unique blendshapes.

An interesting component of facial animation that is largely overlooked is the individual *style* of performing expressions. For example, if you ask two different people to perform a large smile (which is often one of the basis blendshapes), undoubtedly the two people will activate different muscles with different intensities, depending on the particular individual style. This style component is usually baked-in to the blendshapes for each character, and cannot be easily manipulated by artists. In this work, we propose a new facial animation model that allows control over both expression and style.

As we hinted at, individual styles can be characterized by different muscle activations that one character may perform as compared to another. This suggests the use of a physics-based actuation model for facial animation. Several physical face models have been proposed, evolving from the initial, labor-intensive first-principles muscle models [Sifakis et al. 2005] to very recent and innovative implicit models that *obviate* the need for precise anatomical structures [Yang et al. 2022]. While the latter approach seems attractive for our application, the model by Yang et al. is designed for a single identity, and a completely new model would need to be trained for each individual muscle actuation style. In order to understand and represent the nuances in actuation styles across identities, we need to be able to analyze multiple identities simultaneously, within the same physical model. As such, we propose a novel implicit physical face model that is concurrently trained on animation data from multiple identities. By incorporating a broader range of human expression styles, we aim to create a more inclusive, comprehensive, and holistic system for simulating facial actuations.

For this goal, we make several informed design decisions. Our initial observation centers on the use of an implicit neural network, which is agnostic to anatomy discretization. This feature allows us to bypass the laborious process of building topologically-consistent simulation meshes across different identities, a process that often grapples with the trade-off between skin-conforming accuracy and element quality. Consequently, we enable each identity to have its own precise simulation mesh with well-sized elements. Second, in observing that each individual possesses a unique material space defining their actuation field, we recognize that directly learning our model within these spaces neglects the shared muscle structures common among individuals. Consequently, to foster a more unified muscle structure, we propose a canonical space to let our model learn from such regularities. Thus, each identity-specific material space is continuously mapped to this shared space, separating the learning and simulation spaces. Third, we parameterize expressions using traditional blendshape weights for straightforward control. Meanwhile, style is parameterized by quantized style codes, one for each identity. Our network is trained using performance data from a small selection of identities. In addition, we show how Lipschitz weight regularization, previously used only in geometry generation [Liu et al. 2022], can be used to help smooth actuation generation, and, more importantly, disentangle the expression and style spaces. Lastly, a robust and versatile collision model is integrated.

Allowing multi-identity training for an implicit physics-based model opens up several interesting possibilities that have not been demonstrated before. For example, we show that it is possible to learn generalized actuations across the identities. As such, we do not require the different identities to perform the same expressions for training, and can generalize to new animations not seen at training time. Furthermore, as a key innovation of our method, multi-identity training allows us to learn identity-specific styles and interpolate between them at inference time. These properties offer distinct advantages over traditional facial rigs, which we demonstrate across animation tasks such as style transfer and retargeting. Finally, we manifest that our physics-based model allows physical effects like contact and collision handling, manipulating the anatomy (e.g. resizing bones), the addition of extra forces (such as friction), and artistic control of muscle activation strength (e.g. to generate realistic facial paralysis). These applications highlight the superiority of our method, setting it apart from conventional approaches.

2 RELATED WORK

We discuss related research in the areas of physics-based facial animation, facial animation retargeting and animation style transfer. For a more comprehensive overview, we refer to Egger et al. [2020].

2.1 Physics-Based Facial Animation

Physics-based methodologies have been employed to animate faces, aiming to capture complex, nonlinear deformations more accurately [Sifakis et al. 2005]. This often involves a volumetric discretization process that incorporates anatomical features, such as bones and muscles, to create a more lifelike representation. The physics simulators can then model intricate effects, including internal tissue and bone contacts and interactions with external objects. This has been demonstrated, for example, in the simulation of temporal dynamics due to inertia and external forces [Ichim et al. 2017], mandible kinematics [Dorda et al. 2022], or lip sliding [Kim et al. 2019]. More recently, physics simulators have also been used to learn anatomically inspired face models for animation and retargeting [Choi et al. 2022]. However, developing these simulation-ready facial meshes is laborious. Given this, there's been an inclination to merge physics with traditional blendshape models. This amalgamation attempts to bring together the best of both worlds: the simplicity of blendshapes and the enhanced realism of physics-based simulations [Barrielle et al. 2016]. While these combined models do offer benefits, such as a physics layer to account for secondary movements and interactions [Ichim et al. 2016], they are inherently limited in terms of physical realism.

Recently, differentiable physics solvers have emerged as a popular tool in facial animation to solve complex inverse problems. These solvers, through techniques like the differentiable Finite Element Method, allow for the optimization of mechanical properties of the face, in particular, soft tissue parameters, heterogeneous stiffness and prestrain [Kadleček and Kavan 2019], for enhanced realism. They've also been applied in diverse scenarios, such as determining parameters for surgical treatments using a visible facial 3D surface [Dorda et al. 2022] or in muscle-based systems for both shape matching and facial pose recognition from a single image

input [Bao et al. 2018]. Importantly, newer models are shifting away from requiring complex muscle structures. Instead, they focus on a shape-targeting approach [Klár et al. 2020], where underlying elements of the mesh are all regarded as active and actuated to represent facial expressions. Srinivasan et al. [2021] combined neural networks with this general muscle model, to learn the muscle actuation mechanism. Later studies moved towards an implicit neural representation, making this actuation mechanism more compact, adaptable and resolution-independent [Yang et al. 2022]. Our proposed face model is similar to the one of Yang et al. [2022], but we train the network on captured data from multiple individuals simultaneously, which is key to learning different expression styles based on actuation. This multi-identity training comes with challenges regarding different style, geometry and simulation topology per identity, which we address in our method.

2.2 Facial Animation Retargeting

Our implicit physical face model can be used for the application of facial animation retargeting. The most commonly-used approach for retargeting is to employ blendshape-based facial rigs [Lewis et al. 2014], where face deformations are formed by blending linear combinations of pre-defined semantic blendshapes. If the source and target character rigs share the same semantics, retargeting is as simple as transferring the blendweights corresponding to an animation. Several attempts have been made to improve on blendshape-based facial retargeting, such as adding supplemental blendshapes [Kim et al. 2011], coefficient re-mapping for characteristic blendshapes [Song et al. 2011], calibrating the range of motion between source and target characters [Ribera et al. 2017], and considering local models [Chandran et al. 2022a; Liu et al. 2011; Xu et al. 2014].

Blendshape rigs are usually costly to create, due to the large number of facial shapes that are required for high quality, and retargeted results often require additional artist editing [Seol et al. 2011]. On the other hand, animation retargeting can be achieved using a basic deformation transfer approach [Sumner and Popović 2004], which requires only a single neutral expression of the target character. This method is only suitable when all deformation details, including fine-scale wrinkles, should be transferred from the source to the target character. One limitation of many face models is that they are linear, limiting the expressiveness of motion. To counter this limitation, nonlinear deep face models have been proposed [Chandran et al. 2020, 2022b]. Neural networks specifically designed for retargeting have also been proposed [Athar et al. 2022; Bai et al. 2021; Chaudhuri et al. 2019; Costigan et al. 2014; Zhang et al. 2022].

Other approaches to facial animation retargeting include expression cloning [Noh and Neumann 2001; Seol et al. 2012], retargeting through a 2D image space [Kim et al. 2021; Moser et al. 2021], dedicated lip-sync retargeting using contours [Bhat et al. 2013], and real-time puppeteering [Bouaziz et al. 2013; Cao et al. 2014; Li et al. 2013]. The current state-of-the-art is the local, anatomically-constrained method of Chandran et al. [2022a], which combines local blendshape weight transfer with skin and bone sliding constraints. All approaches described in this section, fail to consider the style of the expression during retargeting and do not offer animation style transfer. Our approach supports facial animation retargeting, allowing

retargeting of expression, style or both components from a source performance to a target character. Furthermore, existing retargeting approaches do not consider physical effects like lip contact, collision with internal teeth and bone structures, or other physical effects that allow artist control over the retargeted performance.

2.3 Animation Style Transfer

A core component of our model is its ability to transfer animation style between characters. Representing and transferring style has become very common in the domain of image processing [Huang and Belongie 2017; Jing et al. 2020] (refer to Jing et al. [2019] for a detailed survey). In terms of facial images and video, learning a particular speaker's motion style is a popular component of audio-based 2D facial animation [Pham et al. 2023; Suwajanakorn et al. 2017; Zhou et al. 2020]. This concept has been applied also in audio-driven 3D facial animation, where the goal is to create 3D animation from speech with individual style [Balamurugan et al. 2022; Cudeiro et al. 2019; Lahiri et al. 2021; Richard et al. 2021].

Full body motion style transfer for 3D skeleton animation has been a well-studied topic [Jang et al. 2022; Park et al. 2021a; Tao et al. 2022], including learning style from video input [Aberman et al. 2020]. To our knowledge, facial animation style transfer has been largely overlooked in literature, and given today's tools, we believe it is extremely difficult to create animations that contain the geometry of one character but in the style of another character. To address this challenge, we present the first implicit physics method trained on multiple identities, driven by both expression and style, allowing facial animation style transfer.

3 BACKGROUND

We first review the quasistatic, differentiable, active soft-body simulation and actuation networks, and then introduce our physics-based facial animation model. We refer the readers to previous work [Srinivasan et al. 2021; Yang et al. 2022] for more details.

Human bodies and faces are examples of active objects that can change their shape through internal actuation mechanisms. In facial simulation, the actuation mechanism is modeled using the actuation tensor field $\mathcal{A}$, which is defined over the material space $\mathcal{D}_M$ (undeformed facial space) and consists of local 3×3 symmetric matrices $\mathbf{A}$ that contain information about the contractile directions and magnitudes. Given the measurement of local deformation with deformation gradient $\mathbf{F} \in \mathbb{R}^{3 \times 3}$, the shape targeting model [Klár et al. 2020] is often used as the energy density function for simulation:

$$\Psi(\mathbf{F}, \mathbf{A}) = \min_{\mathbf{R} \in \mathrm{SO}(3)} ||\mathbf{F} - \mathbf{R}\mathbf{A}||_F^2. \tag{1}$$

Here, the rotation matrix $\mathbf{R}$ factors away the rotational difference between $\mathbf{F}$ and $\mathbf{A}$ and makes Ψ rotationally-invariant. The minimum of the total energy implicitly defines the sensitivity matrices of the simulation, which can be used to compute the gradient flowing into $\mathcal{A}$, enabling the training of generative actuation networks.

The influential work of Srinivasan et al. [2021] proposes an explicit *discretization-attached* actuation network replicating muscle actuation mechanisms and a latent activation space. Later, Yang et al. [2022] advances this approach by modeling actuation as a continuous function in the material space $\mathcal{D}_M$, parameterized by

L. Yang, G. Zoss, P. Chandran, P. Gotardo, M. Gross, B. Solenthaler, E. Sifakis, D. Bradley

a coordinate-based neural network $\mathcal{N}_A$, i.e., $\mathcal{A}(\mathbf{x}) = \mathcal{N}_A(\mathbf{x}; \mathbf{z})$, $\mathbf{x} \in \mathcal{D}_M$, for a sampled activation code $\mathbf{z}$ in the activation space. This eliminates the dependency on mesh discretization. The generated actuation field $\mathcal{A}$ is subsequently fed into the differentiable simulation to produce the deformed facial geometry. Such a model deviates from the purely geometry-based methods in that, after training, it generalizes better to novel and unseen physical effects. Besides, it requires no accurate knowledge of the underlying anatomy.

4 METHODOLOGY

In this section, we present our novel method. We build on top of the implicit, continuous actuation model of Yang et al. [2022], leveraging its discretization (topology) agnosticism, and propose several extensions to enable more efficient multi-identity learning, and thus controllability of both expression and individual style.

4.1 Dataset and Pre-Processing

Our training data consists of 3D facial animation sequences of nine identities performing various actions such as dialogue and random facial expressions, sourced from the work of Chandran et al. [2022a]. For each identity, we require a small amount of performance data, typically less than 30 seconds, and a collection of 20 static facial expressions. While these expressions are semantically consistent across identities, they embody unique styles, defining an individual blendshape basis for each identity. The performance sequences are projected into the respective blendshape basis of each identity, by optimizing for per-frame blendweight vectors, which our method will use as input.

The above collected data includes bone meshes, consisting of a skull and mandible, unique to each identity. For each frame, we also track the mandible position, utilizing the method introduced by Zoss et al. [2019]. Additionally, for every identity, we construct a simulation mesh. This mesh is discretized from the undeformed volumetric soft tissue between the facial surface and the bone, i.e., the material space, using regular finite hexahedral elements. We then embed the captured surface and bone meshes into the simulation mesh via trilinear interpolation, as shown in Fig. 1 (right).

The result of data processing is, for each identity, a sequence of facial surface meshes with bone geometry in *topological correspondence* and corresponding blendshape weights, plus a single *identity-specific* simulation mesh of hexahedral elements with facial geometry and bone attachments embedded. We assign six identities for training, while the rest are reserved for cross-identity assessments such as retargeting. For numerical reconstruction evaluation, three sequences, one for each of the three identities from the training set, are specifically designated (approximately 10 seconds).

4.2 Creation of Canonical Space and Mappings

Given that distinct identities each have their unique material spaces that define their actuation fields, we aim to develop a unified, aligned, and implicit actuation model applicable to all identities. For this, we introduce a canonical space $\mathcal{D}_C$, which is a 3D volume that contains the canonical facial surface and bone meshes, sculpted by the artist. Furthermore, we establish a mapping function $\phi^{id}(\cdot)$ per identity, transforming identity-specific material space $\mathcal{D}_M^{id}$ into $\mathcal{D}_C$. This

separates simulation and learning spaces, while still retaining a linkage between them. Concretely, the simulator first samples 3D spatial positions $\mathbf{x} \in \mathcal{D}_M^{id}$, which are subsequently warped as $\mathbf{X} \in \mathcal{D}_C$ by the mapping function ϕ^{id}. These $\mathbf{X}$ positions are used to query the actuation network, resulting in the actuation tensors $\mathcal{A}(\mathbf{X})$ required for the simulation. Consequently, the implicit actuation field, now defined on $\mathcal{D}_C$, becomes shared by different identities.

For each identity, we train a small MLP network to represent the mapping (deformation) $\mathbf{X} = \phi^{id}(\mathbf{x})$. We supervise the mapping using points on the surface meshes, thanks to the topological consistency. For the rest of the volume off-surface, we add an elastic regularization term to smooth the volumetric mapping, similar to related work [Park et al. 2021b; Wang et al. 2022]. The total mapping loss for a single identity is defined as:

$$\mathcal{L}_{map}^{id} = \frac{1}{|\mathbf{T}_C|}||\phi^{id}(\mathbf{T}_M) - \mathbf{T}_C|| + \frac{\lambda_e}{N_e}\sum_{i=1}^{N_e} \min_{\mathbf{R} \in SO(3)}||\nabla_{\mathbf{x}_i}\phi^{id}(\mathbf{x}_i) - \mathbf{R}||, \tag{2}$$

where $\mathbf{T}$ denotes the concatenated vertices of the facial surface and bone meshes, and its subscript indicates the corresponding space, e.g., C indicates that it lies in the canonical space $\mathcal{D}_C$, and M in the identity's material space $\mathcal{D}_M^{id}$. We uniformly sample N_e points within $\mathcal{D}_M^{id}$ for regularization, and λ_e is a weight. Using Eq. 2, ϕ^{id} is pre-trained and fixed for each identity during actuation training.

To make the model generalize better, we train the canonical actuation function $\mathcal{A}(\cdot)$ to learn a tensor field that is agnostic to the geometry of each identity's face. The actual simulation uses a warped actuation field defined as:

$$\tilde{\mathcal{A}}^{id}(\mathbf{x}) = \mathbf{R}_{\phi^{id}}(\mathbf{x})^{\top}\mathcal{A}(\phi^{id}(\mathbf{x}))\mathbf{R}_{\phi^{id}}(\mathbf{x}), \tag{3}$$

where $\mathbf{R}_{\phi^{id}}$ is the rotational component of the Jacobian of ϕ^{id} at $\mathbf{x}$. While a single canonical field $\mathcal{A}(\cdot)$ is trained for all identities, each one has their own, identity-adapted $\tilde{\mathcal{A}}^{id}(\cdot)$. The intuition for this idea is that the learned canonical contractile direction should also be the same (aligned) for all identity within $\mathcal{D}_C$. For more details, please refer to the supplementary material.

4.3 Generation of Physical Constraints

Our method trains a unified implicit actuation network that defines the actuation tensor field solely in the shared canonical space $\mathcal{D}_C$, while also accommodating individual styles. Learning within $\mathcal{D}_C$ eases the burden to adapt to different material spaces. This streamlines its ability to discern common features across various identities and improves the transferability of the learned actuation mechanism. To integrate individual nuances, we assign a distinct style code to each identity. Optimized alongside the network, this style code facilitates the utilization of all training data from all identities to train a singular network, thereby enhancing its generalization to unseen blendweight vectors originating from any actor's blendshape basis.

As shown in the pipeline, Fig. 2 (center), given the sampled expression code and the style code, two MLPs are used to map them to higher dimensional latent space, which are concatenated to get the activation code $\mathbf{z}$. This code is used as the input to the implicit actuation network $\mathcal{N}_A$ to control the canonical actuation tensor field $\mathcal{A}$ through modulation. The modulation code $\mathbf{m}$ is generated
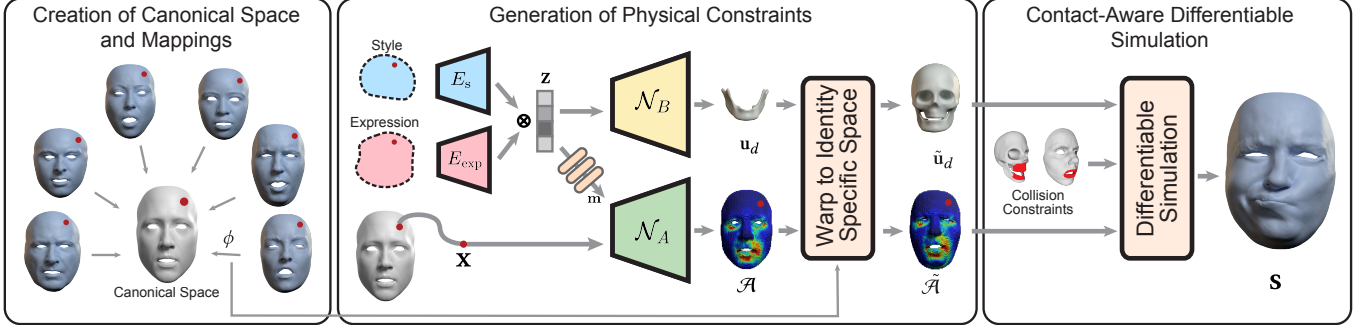
Fig. 2. We illustrate our pipeline, consisting of the creation of a canonical space for training, our style- and expression-conditioned network for generating multi-identity physical constraints, and the contact-aware differentiable simulation.

from $\mathbf{z}$ by a tiny MLP network. $\mathcal{A}$ is then warped to the identity-specific material space $\mathcal{D}_M^{id}$ by the mapping function ϕ^{id}, as $\tilde{\mathcal{A}}^{id}$. At the same time, $\mathbf{z}$ is the direct input to another network $\mathcal{N}_B$ to get the jaw transformation from the rest state. Analogous to actuation learning, we train $\mathcal{N}_B$ within the canonical jaw coordinate system. This allows it to be applied to any identity's jaw $\tilde{\mathbf{u}}_d^{id}$ in the identity's jaw coordinate system. Finally, the generated physical constraints are fed into the differentiable simulator to get the embedded facial geometry $\mathbf{s}$. During training, we optimize for one style code per input identity, and use the per-frame blendweights as the expression code, together with the per-frame reconstructed geometry as the target of the simulation. After training, the model can be driven with any expression and style code, for any of the seen identities. The objective for training our system consists of three terms. The first term is to minimize the positional and normal difference between the captured and embedded facial geometry.

$$\mathcal{L}_{\text{geo}} = ||\mathbf{s} - \mathbf{s}_{\text{captured}}|| + ||1 - \mathbf{n} \cdot \mathbf{n}_{\text{captured}}||. \quad (4)$$

The second term is to regularize the actuation intensity to be as little actuated as possible,

$$\mathcal{L}_{\text{act}} = \frac{1}{N_a} \sum_{i=1}^{N_a} ||\mathbf{A}_i - \mathbf{I}||, \quad (5)$$

where $\mathbf{I}$ is the identity matrix, and we sample N_a actuation tensors. This term can also stabilize the simulation and therefore the training process. The third term is to enforce the smoothness by constraining the Lipschitz constant of the network to be small. Since the output of the network is a tensor, penalizing the gradient using auto differentiation is not trivial. Liu et al. [2022] propose a more effective way by defining the smoothness energy solely based on the weights $\mathbf{W}$ of the network regardless of the inputs. The loose Lipschitz bound c of a L-layer fully-connected network can be penalized directly in a differentiable manner by augmenting each layer with a Lipschitz weight normalization layer [Liu et al. 2022] that has a trainable Lipschitz bound c_i. Therefore the third loss is defined as:

$$\mathcal{L}_{\text{lip}} = \prod_{i=1}^{L} \text{softplus}(c_i), \quad (6)$$

The overall loss is the weighted sum of the three terms:

$$\mathcal{L} = \mathcal{L}_{\text{geo}} + \lambda_{\text{act}}\mathcal{L}_{\text{act}} + \lambda_{\text{lip}}\mathcal{L}_{\text{lip}}. \quad (7)$$

where λ_{act} and λ_{lip} are the weights for the two regularization terms. For more details, please refer to the supplementary material.

4.4 Contact-Aware Differentiable Simulation

Our simulation framework consists of three energy terms: shape targeting, bone attachment, and contact energies. The first two terms are based on Projective Dynamics [Bouaziz et al. 2014], where the local constraint can be generally represented as follows:

$$E_i(\mathbf{u}) = \min_{\mathbf{y}_i} \frac{\omega_i}{2} ||\mathbf{G}_i \mathbf{S}_i \mathbf{u} - \mathbf{B}_i \mathbf{y}_i||_F^2 \quad \text{s.t.} \quad C_i(\mathbf{y}_i) = 0, \quad (8)$$

where $\mathbf{u}$ denotes the simulation vertices, ω_i is a weight coefficient, and $\mathbf{y}_i$ an auxiliary variable, embodying the target position. $\mathbf{S}_i$ is a selection matrix choosing DOFs involved in E_i. $\mathbf{G}_i$ and $\mathbf{B}_i$ are designed to facilitate the distance measure. The total energy $E(\mathbf{u})$ is the sum of all these local constraints. This type of energy is solved using a local-global scheme. In the local step, we solve for all the $\mathbf{y}_i$ in parallel. In the global step, we solve the following linear system derived by setting $\nabla E = 0$:

$$\left(\sum_i \omega_i \mathbf{S}_i^\top \mathbf{G}_i^\top \mathbf{G}_i \mathbf{S}_i \right) \mathbf{u} = \sum_i \omega_i \mathbf{S}_i^\top \mathbf{G}_i^\top \mathbf{B}_i \mathbf{y}_i \quad (9)$$

For the collision model, we adopt the IPC model [Li et al. 2020] for its effectiveness, robustness, and versatility. Without adding some inequality constraints explicitly, the IPC model exploits a smoothly clamped barrier potential to penalize collision:

$$B_i(d_i) = \begin{cases} -\left(d_i - \hat{d}\right)^2 \ln\left(\frac{d_i}{\hat{d}}\right), & 0 < d_i < \hat{d} \\ 0, & d_i \geq \hat{d} \end{cases} \quad (10)$$

where d_i is the unsigned distance between the given pair i of surface primitives, i.e., vertex-triangle or edge-edge, and $\hat{d}$ is a user-defined tolerance of the collision resolution. As long as we ensure the collision-free state at the beginning of the simulation, using such potential forestalls any interpenetration afterwards. Our idea is to use this barrier energy in our quasistatic facial simulation

framework by solving the following optimization problem:

$$\min_{\mathbf{u}} E(\mathbf{u}) + B(\mathbf{u}), \tag{11}$$

where $B(\mathbf{u})$ is the sum of all the barrier potentials for all the collision pairs constructed from $\mathbf{u}$. However, solving this optimization is not easy. $B(\mathbf{u})$ is a highly nonlinear energy, and not PD-based. For consistency, we project $B(\mathbf{u})$ to its local hyperparaboloid centered at $\hat{\mathbf{u}}$, i.e., from the last iteration, using Taylor expansion:

$$\hat{B}(\mathbf{u}) = B(\hat{\mathbf{u}}) + \nabla B(\hat{\mathbf{u}})^{\top}(\mathbf{u} - \hat{\mathbf{u}}) + \frac{1}{2}(\mathbf{u} - \hat{\mathbf{u}})^{\top}\nabla^2 B(\hat{\mathbf{u}})(\mathbf{u} - \hat{\mathbf{u}}). \tag{12}$$

After projection, we add $\nabla \hat{B} = 0$ into the global step of Eq. 9, giving the following new linear system:

$$\left(\nabla^2 B + \underbrace{\sum_i \omega_i \mathbf{S}_i^{\top}\mathbf{G}_i^{\top}\mathbf{G}_i\mathbf{S}_i}_{\mathbf{K}}\right)\mathbf{u} = \nabla^2 B\hat{\mathbf{u}} - \nabla B + \sum_i \omega_i \mathbf{S}_i^{\top}\mathbf{G}_i^{\top}\mathbf{B}_i\mathbf{y}_i, \tag{13}$$

where we remove the explicit evaluation of ∇B and $\nabla^2 B$ for simplicity. We refer the readers to literature [Li et al. 2020; Yang et al. 2022] for the details on the construction of the hessians. In Eq. 13, $\mathbf{K}$ is a Laplacian-like positive definite symmetric matrix, that remains fixed throughout the simulation. $\nabla^2 B$, on the other hand, constantly changes but only affects a small portion of the left-hand side. To maintain positive definiteness of the linear system, we project $\nabla^2 B$ to the cone of positive semidefinite matrices, as in Li et al. [2020]. This linear system can then be efficiently solved using the preconditioned conjugate gradient method, with prefactorized $\mathbf{K}$ serving as the preconditioner. In addition, we use continuous collision detection to ensure that the system remains penetration-free at all iterations, and line search to guarantee convergence when necessary. To improve the performance, we mark out the collision-prone regions (e.g. the lips) on the embedded facial surface and only consider those regions when constructing B, as shown in Fig. 2. Since we use embedded simulation for its efficiency, the finer collision proxy $\mathbf{p}$ for constructing B is linearly embedded in the coarser simulation mesh $\mathbf{u}$, i.e., $\mathbf{p} = \mathbf{W}\mathbf{u}$. Consequently, we get $\nabla^2 B(\mathbf{u}) = \mathbf{W}^{\top}\nabla^2 B(\mathbf{p})\mathbf{W}$. The linear mapping also guarantees the linear trajectories of $\mathbf{p}$, making it possible to use CCD in our optimization framework.

4.5 Implementation Details

For the simulator during training, we detach our collision model for speed-up, and use Projective Dynamics [Bouaziz et al. 2014] as our forward solver, and differentiable Projective Dynamics [Du et al. 2022] as backward solver, where we adopt similar optimization strategies as in Yang et al. [2022]. During inference, we use our proposed solver with the collision model attached. For the actuation network, we use a similar architecture as Yang et al. [2022] but with two main differences. First, we replace all the SIREN layers except the first with GeLU layers, since we found SIREN layers to produce unstable and noisy results that were extremely sensitive to initialization. Second we apply a tanh activation function for the modulating code $\mathbf{m}$ that directly modulates the actuation network backbone, and can be used for T-SNE visualization, as shown in Fig. 4. For the visualization of actuation values, we use the Frobenius distance between the actuation matrix and the identity matrix. For

all experiments shown in Section 5, we use PyTorch [Paszke et al. 2019] on the NVIDIA RTX 2080Ti with the simulator integrated as a layer. For more details, please refer to the supplementary material.

Table 1. Numerical evaluation of different models on the test set.

Methods	Vertex Error (mm) ↓
Linear Blendshape Rig	0.5549
Model-N (w.o. CS)	0.4119
Model-CS (w. CS)	0.4105
Model-CSL (w. CS + Lip)	0.4055
Model-CSW (w. CS + Warp)	0.4020
Model-CSWL (w. CS + Warp + Lip) [Ours]	**0.3849**
Model-CSWL (Displacement Only)	0.4313
Model-CSWL (Single-Identity)	0.4218
[Yang et al. 2022]	0.4143

5 EXPERIMENTS

We start by evaluating our model learned on multiple identities, and then demonstrate several applications like style transfer, animation retargeting and other physical effects.

5.1 Model Evaluation

Multi-identity Model. In this section, we quantitatively and qualitatively evaluate our multi-identity model. For quantitative evaluation we use a reconstruction error metric, which is the average Euclidean distance between the ground truth and the reconstructed mesh vertices. As outlined in Section 4.1, three sequences, one for each of the three identities from the training set, are allocated for this purpose. However, it is important to note that the reconstruction accuracy only reflects the performance of the model on within-identity animation tasks, where ground-truth performance data is available. Cross-identity animation tasks including retargeting and style transfer provide a more comprehensive test of the model's generalization capabilities, for which we provide qualitative comparisons.

There are three main components in our model that we can evaluate, including the canonical space, the warp operation, and the Lipschitz regularization. First, we evaluate our canonical space and the warp operation. The Canonical space is expected to let the model learn a more unified muscle structure, and thus ease the burden to adapt to different material spaces. The warp operation is further introduced for better generalization. For evaluation, we train three models with different simplifications. The first (Model-N) is a basic multi-identity model trained on separate material spaces, without the canonical space. The second (Model-CS) uses the canonical space for multi-identity learning, but without a per-identity warp function. Lastly, Model-CSW employs the canonical space and warp operation for multi-identity learning. Table 1 shows the corresponding errors when reconstructing the test set. Here, Model-CSW exhibits the best performance of these 3 models, and Model-N the worst.

In order to show the benefits of our Lipschitz regularization, we further train two more models, Model-CSL and Model-CSWL that are equipped with the Lipschitz regularization on top of Model-CS, and Model-CSW. As shown in Table 1, the effect of the warp operation is consistent while the Lipschitz regularization makes the model generalize to the test set even better. Our proposed full model (Model-CSWL) achieves the lowest error on the test set.
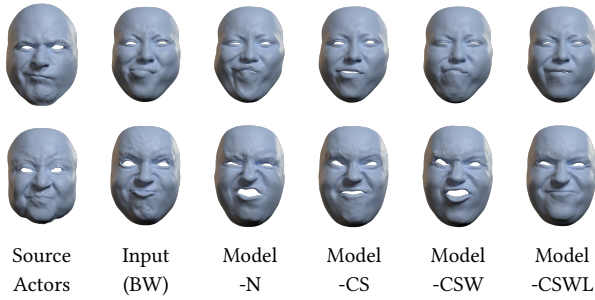
Fig. 3. Qualitative evaluation of our model components on two retargeting examples. The input is the unseen blendweight (BW) vector of a source actor (col 1), illustrated with simple blending on the target identity's blendshape model (col 2). Among the model variants, our proposed model (Model-CSWL) achieves the most natural results with fewer artifacts.

Fig. 5. We compare our model trained on a single identity (S.) and multi-identities (M.) to the model of Yang et al. [2022], with both SIREN and GeLU activations. Our multi-identity model with GeLU exhibits less artifacts.

To further evaluate the Lipschitz regularization, we analyze our model's ability to disentangle expression and style with and without regularization. To this end, we employ T-SNE to visualize a comprehensive set of actuation modulating codes $\mathbf{m}$, generated by pairing the expression blendweight vectors of the entire training dataset (across all identities) with each learned style code. Successful disentanglement would mean that these modulation codes should cluster by identity style, whether the expression blendweights come from the particular identity or another identity. As demonstrated in Fig. 4, Model-CSWL exhibits superior style and expression disentanglement compared to Model-CSW, where points drifting outside clusters or misallocated to incorrect clusters can be observed.

Qualitative comparisons of these models, in terms of cross-identity animation quality, are depicted in Fig. 3. The full model, Model-CSWL, generates more natural results, while the other methods exhibit more expression mismatches and artifacts.

Comparison with Single-identity Model. We also compare our model to our single-identity variant, both quantitatively and qualitatively. To ensure a fair comparison, we train three single-identity models—each for a test identity—with the same specifications as our multi-identity model. The key distinction is the data source: single models use data from one identity, while the multi-identity model uses data from several. As indicated in Table 1, our multi-identity
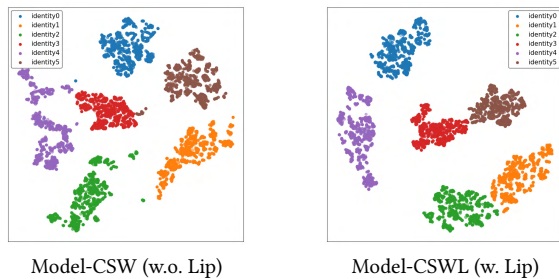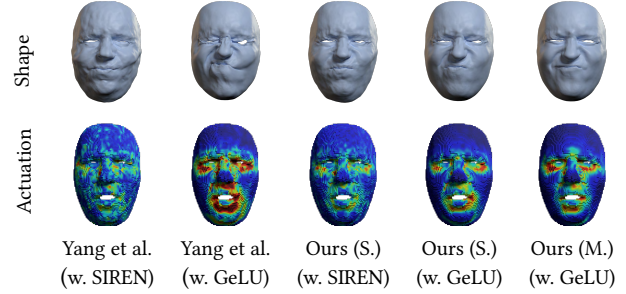
model has a lower error on the test set. The visualization of the error points, collected from the averaged vertex error per frame, is shown on the right half of the Fig. 6. Note that the error points are biased towards the upper part of the diagonal dotted line. We also compare cross-identity animation results. As shown on the left half of Fig. 6, our multi-identity model achieves better shape matching and more natural results. These experiments show that when equipped with more data from other identities, our model generalizes better and is more effective, especially when it comes to cross-identity animation. It is worth noting that our single-identity model is adapted from the multi-identity model, and thus has the two main differences from Yang et al. [2022] that we mentioned in Section 4.5. Namely, it replaces all but the first SIREN layers with GeLU layers and adds a tanh activation after modulation code. As shown in Fig. 5, these changes largely alleviate artifacts visible in the method of Yang et al. [2022], while achieving comparable accuracy (see Table 1).

Comparison with Other Models. As reported in Table 1, our model performs better than a linear blendshape rig that uses the same blendshape basis as ours. It also outperforms our displacement variant, which bypasses physics completely and directly learns the offset field. More importantly, our model supports style transfer and physical effects.

Collision Model. The IPC model plays a critical role in helping achieve the *collision-free* and *converged* results, as shown in Fig. 7.

Fig. 4. Our model can disentangle identity style vs. expression, as shown by a T-SNE plot of actuation modulation codes $\mathbf{m}$. With Lipschitz regularization (right) achieves better disentanglement than without (left).
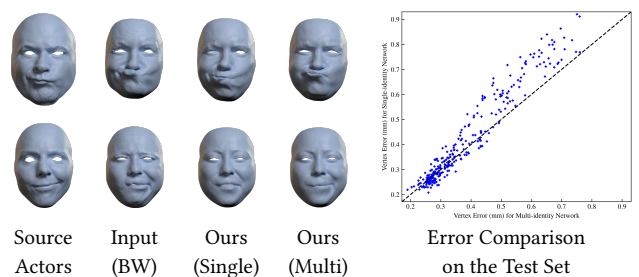

Fig. 6. Comparison between single-identity and multi-identity models on a retargeting example (left), and visualization of reconstruction errors (right). Our multi-identity model achieves lower errors and more natural shapes.

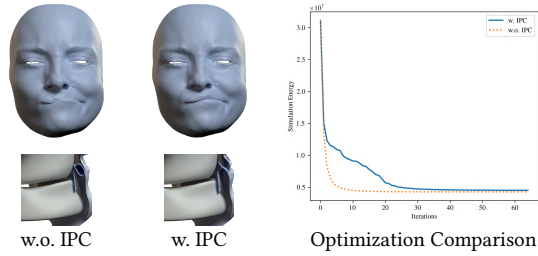w.o. IPC   w. IPC   Optimization Comparison

Fig. 7. We juxtapose our method, with and without IPC. The left side illustrates a visual comparison, with the mouth cut-away displayed in the bottom row. The right side depicts the optimization curves.

Note that previous works commonly use the level-set method for consistency with the framework of Projective Dynamics. However, as shown in Li et al. [2020], this method is not robust and prone to penetration. In addition, we find that the IPC model can support various applications. As shown in Fig. 16, the IPC model's natural differentiability could be used to correct data. Second, it naturally supports the simulation of friction effects, as shown in Fig. 13.

5.2 Style Transfer

As demonstrated in Fig. 8, our approach enables style transfer between different identities by utilizing the style code of a source identity with the geometry of a target identity. Notably, our method also facilitates style interpolation, as shown in Fig. 9, a capability made possible by the Lipschitz regularization technique [Liu et al. 2022]. It's worth mentioning that our shared canonical space, $\mathcal{D}_C$, plays an essential role in this capability. Fig. 10 offers a comparison between our method and a model devoid of the canonical space (Model-N). The results indicate that our approach ensures consistent actuation across different identities, while Model-N fails to deliver similar consistency and produces unnatural and inconsistent results. This is because, for the same style code, different identities in Model-N query the actuation tensors within their respective material spaces, leading to inconsistent actuation results. In contrast, our model successfully maintains actuation uniformity due to its learning within the shared canonical space, which supports transferability across different identities.

5.3 Retargeting

Our method can be used for facial animation retargeting thanks to the enhanced generalization from multi-identity learning, as shown in Fig. 11. Retargeting is achieved by using the expression blendweights of one identity but the style and geometry of another identity. Note that the lip penetration problems that are common in conventional methods [Chandran et al. 2022a] can be naturally solved with our method, as shown in Fig. 12. In addition, our model can allow other physical effects on top of the retargeting results, as shown in Section 5.4.

5.4 Various Physical Effects

Friction. Our model effectively simulates friction effects in the quasistatic setting. To enable this, for each frame, we initialize the

Style 1   Style 2   Style 3   Style 4   Style 5   Style 6

Fig. 8. Here we demonstrate the application of Style Transfer. We can transfer the expression style from any source identity to any target identity, while keeping the expression code fixed.

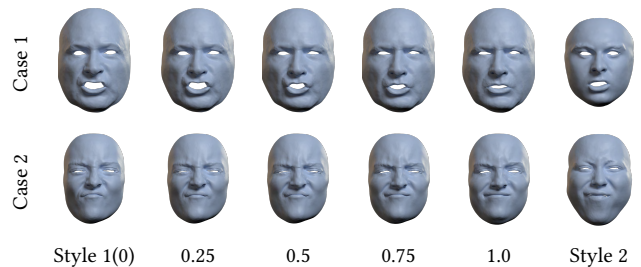Style 1(0)   0.25   0.5   0.75   1.0   Style 2

Fig. 9. We can smoothly interpolate the style code from one style to another, demonstrated here for two examples where the style is slowly changed to that of identity 2, visualized on the geometry of identity 1.

Fig. 10. We show a style transfer comparison between our model and the model without canonical space (Model-N) for 5 identities. The actuation field is inconsistent across identities in Model-N for the same style.

simulation mesh and compute the *lagged* sliding basis from the previous frame. This basis aids in formulating the smoothed static
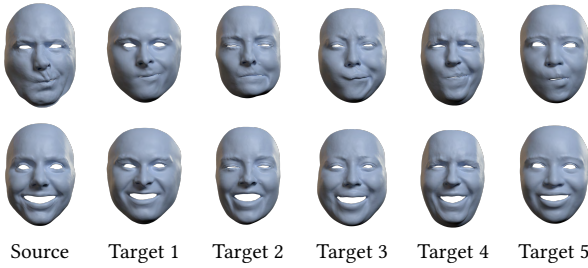
Fig. 11. Here we show animation retargeting for 2 expressions from different source actors retargeted to 5 different target identities (in their own style).
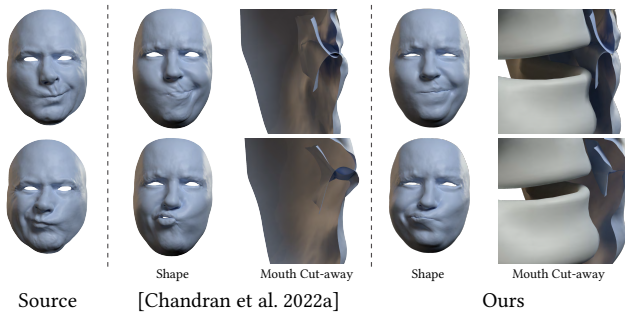

Fig. 12. Comparing to the retargeting method of Chandran et al. [2022a], our method naturally resolves lip intersections in the retarget result.
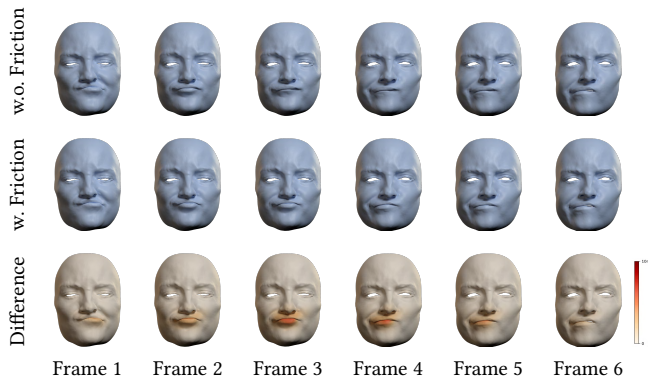

Fig. 13. Our IPC contact model allows to simulate friction effects, shown here for the mouth region.
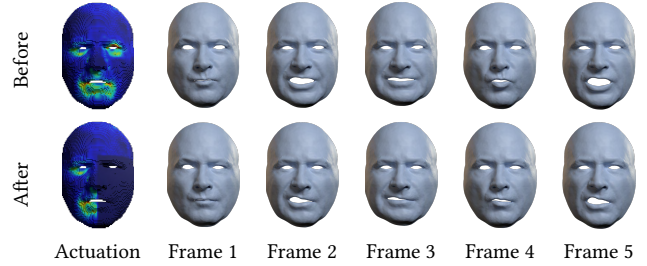

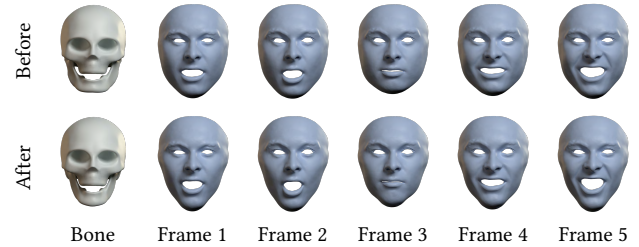Fig. 14. Our model can simulate artistic effects like facial paralysis.

Fig. 15. A physical effect that our model can achieve is osteotomy. Here we shrink the mandible bone and the facial animation naturally matches.
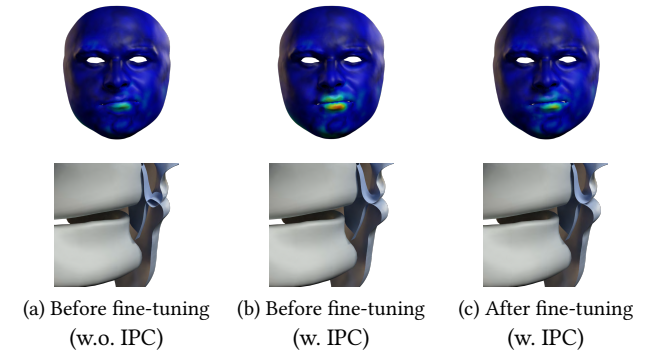

(a) Before fine-tuning (w.o. IPC) (b) Before fine-tuning (w. IPC) (c) After fine-tuning (w. IPC)

Fig. 16. Our collision model supports differentiable collision handling, allowing us to fine-tune the network for better fitting to captured data.

friction energy for the current frame, as in [Li et al. 2020]. Our solver remains compatible, the only difference is to add the hessian and gradient from the friction energy into the optimization loop. As shown in Fig. 13, the lower lip first slips inside and then pops out when no friction is applied, whereas it consistently moves along with the upper lip when friction is imposed upon it.

Paralysis. Our method also allows to add artistic effects like partial facial paralysis, as shown in Fig. 14. Note that the paralyzed part is not completely fixed, but naturally deforms along with the other parts, *e.g.*, the mandible movement, which is in contrast with conventional methods, where the paralyzed part is more or less fixed, as in Chandran et al. [2022a].

Osteotomy. Our model could also be used to simulate osteotomy, where we change the shape of the mandible and replay the actuation. As shown in Fig. 15, our model is able to achieve natural and realistic animation results.

Differentiable Collision. Our collision model naturally supports differentiability. Since the captured data is prone to lip penetration, this can affect training results, as shown in Fig. 16 (a). When we turn on the collision model during testing, the penetration is corrected but increases the error, as shown in Fig. 16 (b). After we fine-tune the network with the collision model integrated, the error is reduced, as shown in Fig. 16 (c).

6 CONCLUSION

We propose a data-driven physical model that leverages multi-identity learning and can transfer performance and style across subjects. Our framework generalizes from animation examples seen on any of the trained identities to create an even richer and more expressive physics-based model for all subjects, and allows for manipulation of expression and style. We inherit some of the design limitations of prior work on data-driven face models [Srinivasan et al. 2021; Yang et al. 2022], such as being limited to the material constitutive model of shape targeting [Klár et al. 2020] as opposed to more biomechanically accurate models. Due to hardware constraints and the computational demands of simulation, we use a small finite set of identities in our training set, thus we do not claim to span the space of human face identities and styles as exhaustively as we cover the space of expressions. That said, by virtue of the Lipschitz regularization, we retain some ability to interpolate between the identities in the training set. Finally, we have not aggressively optimized our pipeline in terms of efficiency of training, focusing rather on quality and generalization than speed of training.

ACKNOWLEDGMENTS

REFERENCES

Kfir Aberman, Yijia Weng, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. 2020. Unpaired Motion Style Transfer from Video to Animation. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 64.

ShahRukh Athar, Zexiang Xu, Kalyan Sunkavalli, Eli Shechtman, and Zhixin Shu. 2022. Rignerf: Fully controllable neural 3d portraits. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*. 20364–20373.

Ziqian Bai, Zhaopeng Cui, Xiaoming Liu, and Ping Tan. 2021. Riggable 3d face reconstruction via in-network optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 6216–6225.

Thambiraja Balamurugan, Ikhsanul Habibie, Sadegh Aliakbarian, Darren Cosker, Christian Theobalt, and Justus Thies. 2022. Imitator: Personalized Speech-driven 3D Facial Animation. arXiv.

Michael Bao, Matthew Cong, Stéphane Grabli, and Ronald Fedkiw. 2018. High-Quality Face Capture Using Anatomical Muscles. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (12 2018). http://arxiv.org/abs/1812.02836

Vincent Barrielle, Nicolas Stoiber, and Cédric Cagniart. 2016. BlendForces: A Dynamic Framework for Facial Animation. *Computer Graphics Forum* (2016). https://doi.org/10.1111/cgf.12836

Kiran S. Bhat, Rony Goldenthal, Yuting Ye, Ronald Mallet, and Michael Koperwas. 2013. High Fidelity Facial Animation Capture and Retargeting with Contours. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Association for Computing Machinery, 7–14.

Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. 2014. Projective dynamics: fusing constraint projections for fast simulation. *ACM Transactions on Graphics* 33, 4 (7 2014), 1–11. https://doi.org/10.1145/2601097.2601116

Sofien Bouaziz, Yangang Wang, and Mark Pauly. 2013. Online Modeling for Realtime Facial Animation. 32, 4, Article 40 (2013).

Chen Cao, Qiming Hou, and Kun Zhou. 2014. Displaced Dynamic Expression Regression for Real-Time Facial Tracking and Animation. 33, 4, Article 43 (2014).

Prashanth Chandran, Derek Bradley, Markus Gross, and Thabo Beeler. 2020. Semantic Deep Face Models. (2020), 345–354.

Prashanth Chandran, Loïc Ciccone, Markus Gross, and Derek Bradley. 2022a. Local Anatomically-Constrained Facial Performance Retargeting. *ACM Trans. Graph.* 41, 4, Article 168 (jul 2022).

Prashanth Chandran, Gaspard Zoss, Markus Gross, Paulo Gotardo, and Derek Bradley. 2022b. Facial Animation with Disentangled Identity, Motion using Transformers. (2022).

Bindita Chaudhuri, Noranart Vesdapunt, and Baoyuan Wang. 2019. Joint face detection and facial motion retargeting for multiple faces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9719–9728.

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*. Association for Computing Machinery, Article 16, 9 pages.

Timothy Costigan, Mukta Prasad, and Rachel McDonnell. 2014. Facial Retargeting Using Neural Networks. Association for Computing Machinery, 31–38.

D. Cudeiro, T. Bolkart, C. Laidlaw, A. Ranjan, and M. J. Black. 2019. Capture, Learning, and Synthesis of 3D Speaking Styles. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10093–10103.

Daniel K. Dorda, Daniel Peter, Dominik Borer, Niko Benjamin Huber, Irena Sailer, Markus Gross, Barbara Solenthaler, and Bernhard Thomaszewski. 2022. Differentiable Simulation for Outcome-Driven Orthognathic Surgery Planning. In *Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation*.

Tao Du, Kui Wu, Pingchuan Ma, Sebastien Wah, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. 2022. DiffPD: Differentiable Projective Dynamics. *ACM Transactions on Graphics* 41, 2 (4 2022), 1–21. https://doi.org/10.1145/3490168

Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, et al. 2020. 3d morphable face models—past, present, and future. *ACM Transactions on Graphics (ToG)* 39, 5 (2020), 1–38.

Xun Huang and Serge Belongie. 2017. Arbitrary Style Transfer in Real-Time With Adaptive Instance Normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Alexandru-Eugen Ichim, Petr Kadleček, Ladislav Kavan, and Mark Pauly. 2017. Phace: Physics-based Face Modeling and Animation. *ACM Transactions on Graphics* 36, 4 (7 2017), 1–14. https://doi.org/10.1145/3072959.3073664

Alexandru-Eugen Ichim, Ladislav Kavan, Merlin Nimier-David, and Mark Pauly. 2016. Building and Animating User-Specific Volumetric Face Rigs. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Eurographics Association, 107–117.

Deok-Kyeong Jang, Soomin Park, and Sung-Hee Lee. 2022. Motion Puzzle: Arbitrary Motion Style Transfer by Body Part. *ACM Trans. Graph.* 41, 3, Article 33 (2022).

Yongcheng Jing, Xiao Liu, Yukang Ding, Xinchao Wang, Errui Ding, Mingli Song, and Shilei Wen. 2020. Dynamic Instance Normalization for Arbitrary Style Transfer. In *AAAI*.

Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song. 2019. Neural Style Transfer: A Review. *IEEE Transactions on Visualization and Computer Graphics*.

Petr Kadleček and Ladislav Kavan. 2019. Building Accurate Physics-based Face Models from Data. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 2, 2 (7 2019), 1–16. https://doi.org/10.1145/3340256

D. Kim, T. Kuang, Y. L. Rodrigues, J. Gateno, S. G. F. Shen, X. Wang, H. Deng, P. Yuan, D. M. Alfi, M. A. K. Liebschner, and J. J. Xia. 2019. A New Approach of Predicting Facial Changes following Orthognathic Surgery using Realistic Lip Sliding Effect. In *Proc. MICCAI*.

Paul Hyunjin Kim, Yeongho Seol, Jaewon Song, and Junyong Noh. 2011. Facial Retargeting by Adding Supplemental Blendshapes. In *Pacific Graphics Short Papers*, Bing-Yu Chen, Jan Kautz, Tong-Yee Lee, and Ming C. Lin (Eds.). The Eurographics Association.

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, 7 (2021), 45–55.

Gergely Klár, Andrew Moffat, Ken Museth, and Eftychios Sifakis. 2020. Shape Targeting: A Versatile Active Elasticity Constitutive Model. In *ACM SIGGRAPH 2020 Talks* (Virtual Event, USA) *(SIGGRAPH '20)*. Association for Computing Machinery, New York, NY, USA, Article 59, 2 pages. https://doi.org/10.1145/3388767.3407379

Avisek Lahiri, Vivek Kwatra, Christian Frueh, John Lewis, and Chris Bregler. 2021. LipSync3D: Data-Efficient Learning of Personalized 3D Talking Faces From Video Using Pose and Lighting Normalization. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

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*, Sylvain Lefebvre and Michela Spagnuolo (Eds.). The Eurographics Association. https://doi.org/10.2312/egst.20141042

Hao Li, Jihun Yu, Yuting Ye, and Chris Bregler. 2013. Realtime Facial Animation with On-the-Fly Correctives. 32, 4, Article 42 (2013).

Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy R Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M Kaufman. 2020. Incremental potential contact: intersection-and inversion-free, large-deformation dynamics. *ACM Trans. Graph.* 39, 4 (2020), 49.

Hsueh-Ti Derek Liu, Francis Williams, Alec Jacobson, Sanja Fidler, and Or Litany. 2022. Learning Smooth Neural Functions via Lipschitz Regularization. *arXiv preprint arXiv:2202.08345* (2022).

Ko-Yun Liu, Wan-Chun Ma, Chun-Fa Chang, Chuan-Chang Wang, and Paul Debevec. 2011. A framework for locally retargeting and rendering facial performance. *Computer Animation and Virtual Worlds* 22, 2-3 (2011), 159–167.

Lucio Moser, Chinyu Chien, Mark Williams, Jose Serra, Darren Hendler, and Doug Roble. 2021. Semi-Supervised Video-Driven Facial Animation Transfer for Production. 40, 6, Article 222 (2021).

Jun-yong Noh and Ulrich Neumann. 2001. Expression Cloning. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*. Association for Computing Machinery, 277–288.

Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. 2021b. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5865–5874.

Soomin Park, Deok-Kyeong Jang, and Sung-Hee Lee. 2021a. Diverse Motion Stylization for Multiple Style Domains via Spatial-Temporal Graph-Based Generative Model. *Proc. ACM Comput. Graph. Interact. Tech.* 4, 3, Article 36 (2021).

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*. 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Trong-Thang Pham, Nhat Le, Tuong Do, Hung Nguyen, Erman Tjiputra, Quang D. Tran, and Anh Nguyen. 2023. Style Transfer for 2D Talking Head Animation. arXiv:2303.09799 [cs.CV]

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, 4, Article 154 (2017).

Alexander Richard, Michael Zollhöfer, Yandong Wen, Fernando de la Torre, and Yaser Sheikh. 2021. MeshTalk: 3D Face Animation From Speech Using Cross-Modality Disentanglement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 1173–1182.

Yeongho Seol, J.P. Lewis, Jaewoo Seo, Byungkuk Choi, Ken Anjyo, and Junyong Noh. 2012. Spacetime Expression Cloning for Blendshapes. 31, 2, Article 14 (apr 2012).

Yeongho Seol, Jaewoo Seo, Hyunjin Kim, J.P. Lewis, and Junyong Noh. 2011. Artist Friendly Facial Animation Retargeting. 30 (12 2011), 162.

Eftychios Sifakis, Igor Neverov, and Ronald Fedkiw. 2005. Automatic determination of facial muscle activations from sparse motion capture marker data. *ACM Transactions on Graphics* 24, 3 (7 2005), 417–425. https://doi.org/10.1145/1073204.1073208

Jaewon Song, Byungkuk Choi, Yeongho Seol, and Junyong Noh. 2011. Characteristic Facial Retargeting. *Comput. Animat. Virtual Worlds* 22, 2–3 (2011), 187–194.

Sangeetha Grama Srinivasan, Qisi Wang, Junior Rojas, Gergely Klár, Ladislav Kavan, and Eftychios Sifakis. 2021. Learning active quasistatic physics-based models from data. *ACM Transactions on Graphics* 40, 4 (8 2021), 1–14. https://doi.org/10.1145/3450626.3459883

Robert W. Sumner and Jovan Popović. 2004. Deformation Transfer for Triangle Meshes. 23, 3 (2004), 399–405.

Supasorn Suwajanakorn, Steven M. Seitz, and Ira Kemelmacher-Shlizerman. 2017. Synthesizing Obama: Learning Lip Sync from Audio. *ACM Trans. Graph.* 36, 4, Article 95 (2017).

T. Tao, X. Zhan, Z. Chen, and M. van de Panne. 2022. Style-ERD: Responsive and Coherent Online Motion Style Transfer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 6583–6593.

Daoye Wang, Prashanth Chandran, Gaspard Zoss, Derek Bradley, and Paulo Gotardo. 2022. Morf: Morphable radiance fields for multiview neural head modeling. In *ACM SIGGRAPH 2022 Conference Proceedings*. 1–9.

Feng Xu, Jinxiang Chai, Yilong Liu, and Xin Tong. 2014. Controllable High-Fidelity Facial Performance Transfer. *ACM Trans. Graph.* 33, 4, Article 42 (2014).

Lingchen Yang, Byungsoo Kim, Gaspard Zoss, Baran Gözcü, Markus Gross, and Barbara Solenthaler. 2022. Implicit neural representation for physics-driven actuated soft bodies. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–10.

Juyong Zhang, Keyu Chen, and Jianmin Zheng. 2022. Facial Expression Retargeting From Human to Avatar Made Easy. *IEEE Transactions on Visualization and Computer Graphics* 28, 2 (2022), 1274–1287.

Yang Zhou, Xintong Han, Eli Shechtman, Jose Echevarria, Evangelos Kalogerakis, and Dingzeyu Li. 2020. MakeItTalk: Speaker-Aware Talking-Head Animation. *ACM Trans. Graph.* 39, 6, Article 221 (2020).

Gaspard Zoss, Thabo Beeler, Markus Gross, and Derek Bradley. 2019. Accurate markerless jaw tracking for facial performance capture. *ACM Transactions on Graphics* 38, 4 (7 2019), 1–8. https://doi.org/10.1145/3306346.3323044