Learning a model of facial shape and expression from 4D scans

TIANYE LI*†, University of Southern California and Max Planck Institute for Intelligent Systems
TIMO BOLKART*, Max Planck Institute for Intelligent Systems
MICHAEL J. BLACK, Max Planck Institute for Intelligent Systems
HAO LI, Pinscreen, University of Southern California, and USC Institute for Creative Technologies
JAVIER ROMERO†, Body Labs Inc.

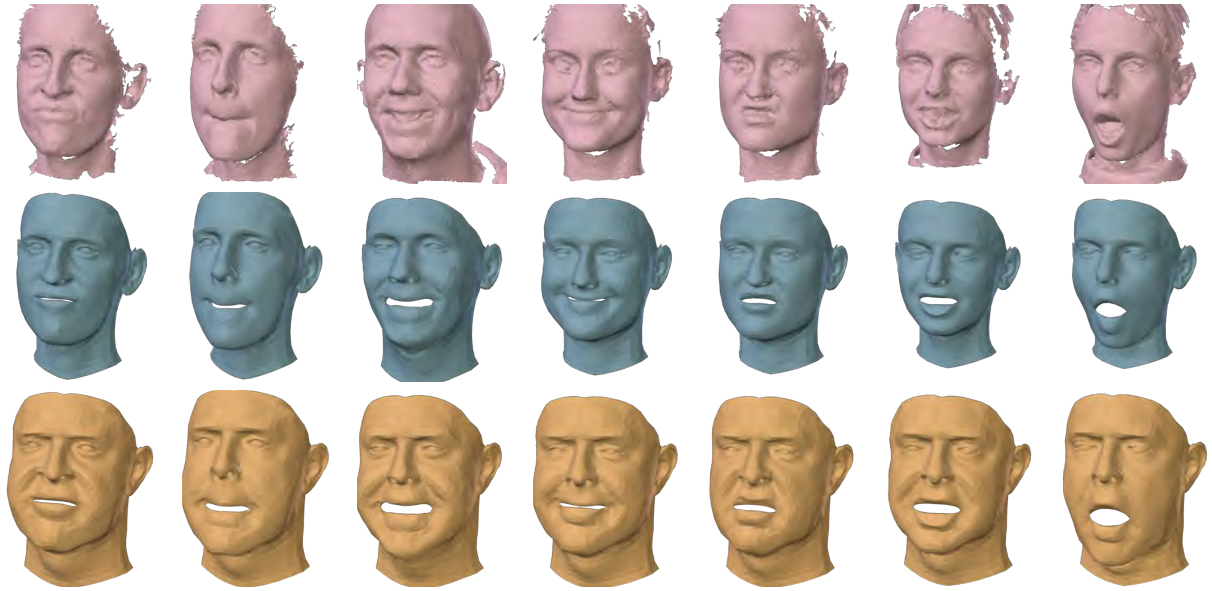

Fig. 1. **FLAME example.** Top: Samples of the D3DFACS dataset. Middle: Model-only registration. Bottom: Expression transfer to Beeler et al. [2011] subject using model only.

The field of 3D face modeling has a large gap between high-end and low-end methods. At the high end, the best facial animation is indistinguishable from real humans, but this comes at the cost of extensive manual labor. At the low end, face capture from consumer depth sensors relies on 3D face models that are not expressive enough to capture the variability in natural facial shape and expression. We seek a middle ground by learning a facial model from thousands of accurately aligned 3D scans. Our FLAME model (Faces Learned with an Articulated Model and Expressions) is designed to work with existing graphics software and be easy to fit to data. FLAME uses a linear shape space trained from 3800 scans of human heads. FLAME combines this linear shape space with an articulated jaw, neck, and eyeballs, pose-dependent corrective blendshapes, and additional global expression blendshapes. The pose and expression dependent articulations are learned from 4D face sequences in the D3DFACS dataset along with additional 4D sequences. We accurately register a template mesh to the scan sequences and make the D3DFACS registrations available for research purposes. In total the model is trained from over 33, 000 scans. FLAME is low-dimensional but more expressive than the FaceWarehouse model and the Basel Face Model. We compare FLAME to these models by fitting them to static 3D scans and 4D sequences using the same optimization method. FLAME is significantly more accurate and is available for research purposes (http://flame.is.tue.mpg.de).

CCS Concepts: • **Computing methodologies** → *Mesh models*;

Additional Key Words and Phrases: Face model, mesh registration, 4D registration, shape, facial expression, blend skinning, learning.

ACM Reference Format:
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, 6, Article 194 (November 2017), 17 pages. https://doi.org/10.1145/3130800.3130813

*Both authors contributed equally to the paper
†This research was performed while TL and JR were at the MPI for Intelligent Systems.

Authors' email addresses: tianyeli@usc.edu; {tbolkart, black}@tue.mpg.de; hao@hao-li.com; javier.romero@bodylabs.com.

0730-0301/2017/11-ART194
https://doi.org/10.1145/3130800.3130813

1 INTRODUCTION

This paper addresses a significant gap in the field of 3D face modeling. At one end of the spectrum are highly accurate, photo-realistic, 3D models of individuals that are learned from scans or images of

that individual and/or involve significant input from a 3D artist (e.g. [Alexander et al. 2009]). At the other end are simple generic face models that can be fit to images, video, or RGB-D data but that lack realism (e.g. [Li et al. 2013]). What is missing are generic 3D face models that are compact, can be fit to data, capture realistic 3D face details, and enable animation. Our goal is to move the "low end" models towards the "high end" by learning a model of facial shape and expression from 4D scans (sequences of 3D scans).

Early generic face models are built from limited numbers of 3D face scans of mostly young Europeans in a neutral expression [Blanz and Vetter 1999; Paysan et al. 2009]. More recently, the FaceWarehouse model [Cao et al. 2014] uses scans of 150 people with variation in age and ethnicity and with 20 different facial poses. While widely used, the limited amount of data constrains the range of facial shapes that the above models can express.

To address limitations of existing models, we exploit three heterogeneous datasets, using more than 33,000 3D scans in total. Our FLAME model (Faces Learned with an Articulated Model and Expressions) is factored in that it separates the representation of identity, pose, and facial expression, similar to models of the human body [Anguelov et al. 2005; Loper et al. 2015]. To keep the model simple, computationally efficient, and compatible with existing game and rendering engines, we define a vertex-based model with a relatively low polygon count, articulation, and blend skinning. Specifically FLAME includes a learned shape space of identity variations, an articulated jaw and neck, and eyeballs that rotate. Additionally we learn pose-dependent blendshapes for the jaw and neck from examples. Finally, we learn "expression" blendshapes to capture non-rigid deformations of the face.

We train the identity shape space from the heads of roughly 4000 CAESAR body scans [Robinette et al. 2002] spanning a wide range of ages, ethnicities, and both genders. To model pose and expression variation we use over 400 4D face capture sequences from the D3DFACS dataset [Cosker et al. 2011] and additional 4D sequences that we captured, spanning more expression variation. All the model parameters are learned from data to minimize 3D reconstruction error. To make this possible we perform a detailed temporal registration of our template mesh to all the scans (CAESAR and 4D).

The CAESAR dataset has been widely used for modeling 3D body shape [Allen et al. 2003, 2006; Bogo et al. 2015; Chen et al. 2011; Hirshberg et al. 2012; Loper et al. 2015; Pishchulin et al. 2017] but not explicitly for face modeling and existing body models built from CAESAR do not capture facial articulation or expression. Here we take an approach similar to the SMPL body model [Loper et al. 2015] but apply it to the face, neck, and head. SMPL is a parameterized blend-skinned body model that combines an identity shape space, articulated pose, and pose-dependent corrective blendshapes. SMPL does not model facial motion and we go beyond it to learn expression blendshapes.

Given that faces are relatively low-resolution in full body scans, the task of precisely registering the scans is both critical and difficult. To achieve accurate registration a form of co-registration [Hirshberg et al. 2012] is used in which we jointly build a face model and use it to align the raw data. Given registrations we build a facial shape model and show that the resulting identity shape space is richer than that of the Basel Face Model (BFM) [Paysan et al. 2009] and the FaceWarehouse model.

To the best of our knowledge, FaceWarehouse is the only publicly available 3D face database with a large number of facial expression that comes together with template meshes aligned to raw scan data (from a depth sensor). The D3DFACS dataset has much higher quality scans but does not contain aligned meshes. Registering such 4D data presents yet another challenge. To do so we use co-registration and image texture to obtain high quality alignment from a sequence of 3D scans with texture; this is similar to work on full bodies [Bogo et al. 2014]. Including eyeballs in the model also improves alignment for the eye region, particularly the eye lids. The registration and model learning process is fully automatic.

In a departure from previous work, we do not tie the expression blendshapes to facial action units (FACS) [Ekman and Friesen 1978]. Instead we learn the blendshapes with a global linear model that captures correlations across the face. FACS models are overcomplete in that multiple settings can produce the same shape; this complicates solving for the parameters from data. The FLAME model, in contrast, uses an orthonormal expression space, which is further factored into identity and pose. We argue that this is advantageous for fitting to noisy, partial, or sparse data. Other types of sparse rigs can be built on top of, or derived from, our representation.

Unlike most previous models, we model the head and neck together. This allows the head to rotate relative to the neck and we learn pose-dependent blendshapes to capture how the neck deforms during rotation. This captures effects like the protrusion of neck tendons during rotation, increasing realism.

Our key contribution is a statistical head model that is significantly more accurate and expressive than existing head and face models, while remaining compatible with standard graphics software. In contrast to existing models, FLAME explicitly models head pose and eyeball rotation. Additionally we provide a detailed quantitative comparison between, and analysis of, different models. We make our trained models publicly available for research purposes [FLAME 2017]. The release comprises female and male models along with software to animate and use the model. Furthermore, we make the temporal registration of the D3DFACS dataset publicly available [FLAME 2017] for research purposes, enabling others to train new models.

2 RELATED WORK

Generic face models: Blanz and Vetter [1999] propose the first generic 3D face model learned from scan data. They define a linear subspace to represent shape and texture using principal component analysis (PCA) and show how to fit the model to data. The model is built from head scans of 200 young, mostly Caucasian adults, all in a roughly neutral expression. The model has had significant impact because it was available for research purposes as the Basel Face Model (BFM) [Paysan et al. 2009]. Booth et al. [2017; 2016] learn a linear face model from almost 10,000 facial scans of more diverse subjects in a neutral expression.

To additionally model variations in facial expression, Amberg et al. [2008] combine a PCA model of neutral face shape with a PCA space learned on the residual vectors of expressions from the

neutral shape. The recently published Face2Face framework [Thies et al. 2015] uses a similar model combining linear identity and expression models with an additional linear albedo model to capture appearance. Yang et al. [2011] build several PCA models, one per facial expression, while Vlasic et al. [2005] use a multilinear face model; i.e. a tensor-based model that jointly represents the variations of facial identity and expression. The limited data used to train these methods constrains the range of facial shapes that they can express. Since the identity space of our method is trained from much richer data, our model is more flexible and more able to capture person-specific facial shapes. Tensor-based models assume that facial expressions can be captured by a small number of discrete poses that correspond between people. In contrast, our expression space is trained from sequences of 3D scans. It is unclear how to extend existing tensor methods to deal with the complexity and variability of our temporal data.

Modeling facial motion locally is inspired both by animation and the psychology community where the idea of the Facial Action Coding System (FACS) [Ekman and Friesen 1978] is popular. To capture localized facial details, Neumann et al. [2013] and Ferrari et al. [2015] use sparse linear models. Brunton et al. [2014] use a large number of localized multilinear wavelet models. For animation, facial rigs use localized, hand crafted, blendshapes to give the animator full control. These rigs, however, suffer from significant complexity and redundancy, with overlapping blendshapes. This makes them ill suited as a model to fit to data since they afford multiple solutions for the same shape.

Because generic face models are often quite coarse, several methods augment coarse face shape with additional higher-frequency details. Dutreve et al. [2011], Shi et al. [2014], and Li et al. [2015] add actor specific fine-scale details by defining a wrinkle displacement map from training images. Garrido et al. [2013] build an actor specific blendshape model with the rest-pose shape created from a binocular stereo reconstruction and expressions from an artist generated blendshape model. All these methods are non-generic as they require offline actor-specific preprocessing [Dutreve et al. 2011; Li et al. 2015] or an actor-specific initial 3D mesh.

Cao et al. [2015] use a probability map to model person-specific features such as wrinkles on top of a personalized blendshape model. In their later work [Garrido et al. 2016], they use a generic model to estimate a coarse face shape of an actor, and build personalized high-frequency face rigs by relating high-frequency details to the low-resolution parameters of the personalized blendshape model. Xu et al. [2014] decompose facial performance in a multi-resolution way to transfer details from one mesh to another. They use pre-defined expression blendshapes and do not learn a model. The methods above could be applied as a refinement to add additional facial details to FLAME with a displacement or normal map.

Kozlov et al. [2017] add non-rigid dynamics to facial animation by using "blend materials" to control physical simulation of dynamics; they do not learn the model from scans. Still other work takes collections of images from the Internet and uses a variety of methods, including shape from shading, to extract a person specific 3D shape [Kemelmacher-Shlizerman and Seitz 2011]. They animate the face using 3D flow, warping, and a texture synthesis approach driven by a video sequence [Suwajanakorn et al. 2014, 2015].

Alexander et al. [2009] generate a personalized facial rig for an actress using high-resolution facial scanning and track a facial performance of this actress using a semi-automatic animation system. Wu et al. [2016] combine an anatomical subspace with a local patch-based deformation subspace to realistically model the facial performance of three actors. Similar to our work, the jaw has a rotational degree of freedom, but their method uses personalized subspaces to capture shape details and therefore is not applicable to arbitrary targets.

Personalized blendshape models are often used for facial performance capture. Such personalized rigs typically require a user-specific calibration or training procedure [Li et al. 2010; Weise et al. 2011]. Bouaziz et al. [2013] use an identity PCA model along with deformation transfer [Sumner and Popović 2004]. Cao et al. [2014] generate personalized blendshapes using a multilinear face model based on their FaceWarehouse database. Ichim et al. [2015] generate personalized blendshapes from images of the neutral rest pose and facial motion recordings. These methods either use artist-designed generic blendshapes as initialization, or low resolution local expressions designed to resemble FACS action units (FaceWarehouse).

A key step in building our model is the alignment, or registration, of a template face mesh to 3D scan data. Generic shape alignment is a vast field (e.g. [Bronstein et al. 2008; Davies et al. 2008]), which we do not summarize here. We focus on methods for aligning 3D meshes to scan data to build articulated shape and pose models. There have been many approaches for aligning static face scans [Amberg et al. 2007; Salazar et al. 2014] but few methods focus on 4D data (sequences of 3D meshes). Approaches like Vlasic et al. [2005] rely on manual key points; such approaches do not scale to deal with thousands of scans. Beeler et al. [2011] use repeated anchor frames with the same expression to prevent drift. They register high-resolution meshes with great detail but do so only for three actors, and demonstrate results only qualitatively on several hundred frames; here our automated method generalizes to tens of thousands of frames. Cosker et al. [2011] describe a method to align the D3DFACS dataset using an active appearance model. They do not evaluate the accuracy of the alignment in 3D and do not make the aligned data available. Our approach to face registration uses co-registration [Hirshberg et al. 2012], which has previously only been used with full bodies.

We note that most previous methods have ignored the eyes in the alignment process. This biases the eyelids to explain the noisy geometry of the eyeballs, and creates substantial photometric errors in the eye region. Consequently we add eyeballs to our mesh and show that this helps the alignment process.

3 MODEL FORMULATION

FLAME adapts the SMPL body model formulation [Loper et al. 2015] to heads. The SMPL body model neither models facial pose (articulation of jaw or eyes) nor facial expressions. Extending SMPL makes our model computationally efficient and compatible with existing game engines. We use a consistent notation with SMPL.

In SMPL, geometric deformations are due to the intrinsic shape of the subject, or deformations related to pose changes in the kinematic tree. With faces, however, many deformations are due to muscle

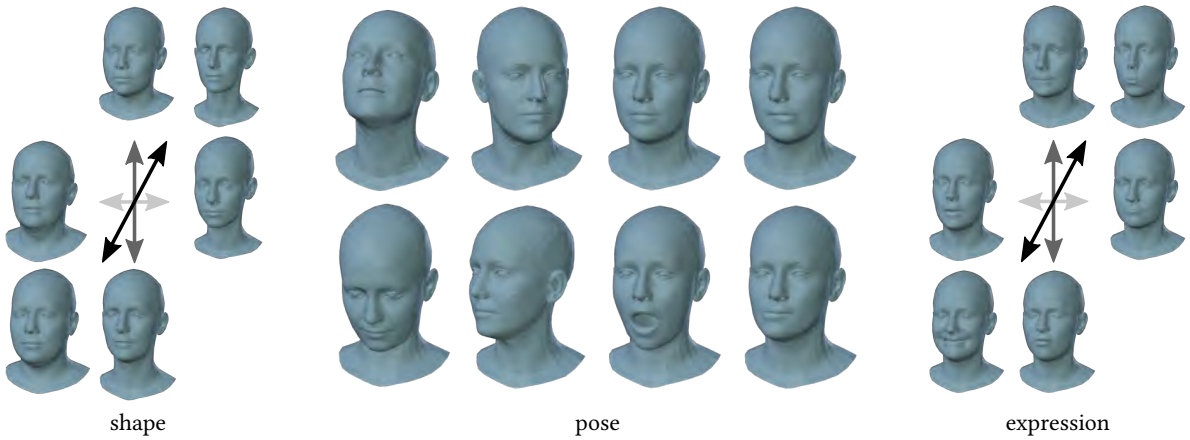

Fig. 2. Parametrization of our model (female model shown). Left: Activation of the first three shape components between −3 and +3 standard deviations. Middle: Pose parameters actuating four of the six neck and jaw joints in a rotational manner. Right: Activation of the first three expression components between −3 and +3 standard deviations.

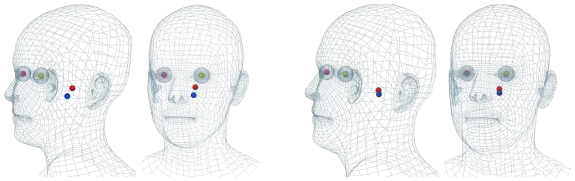

Fig. 3. Joint locations of the female (left) and male (right) FLAME models. Pink/yellow represent right/left eyes. Red is the neck joint and blue the jaw.

activation, which are not related to any articulated pose change. We therefore extend SMPL with additional expression blendshapes as shown in Figure 2. Note that in several experiments we show just the face region for comparison to other methods but FLAME models the face, full head, and neck.

FLAME uses standard vertex based linear blend skinning (LBS) with corrective blendshapes, with $N = 5023$ vertices, $K = 4$ joints (neck, jaw, and eyeballs as shown in Figure 3), and blendshapes, which will be learned from data. FLAME is described by a function $M(\vec{\beta},\vec{\theta},\vec{\psi}) : \mathbb{R}^{|\vec{\beta}|\times|\vec{\theta}|\times|\vec{\psi}|} \rightarrow \mathbb{R}^{3N}$, that takes coefficients describing shape $\vec{\beta} \in \mathbb{R}^{|\vec{\beta}|}$, pose $\vec{\theta} \in \mathbb{R}^{|\vec{\theta}|}$, and expression $\vec{\psi} \in \mathbb{R}^{|\vec{\psi}|}$ and returns N vertices. Each pose vector $\vec{\theta} \in \mathbb{R}^{3K+3}$ contains $K + 1$ rotation vectors ($\in \mathbb{R}^3$) in axis-angle representation; i.e. one three-dimensional rotation vector per joint plus the global rotation.

The model consists of a template mesh, $\overline{\mathbf{T}} \in \mathbb{R}^{3N}$, in the "zero pose" $\vec{\theta}^*$, a shape blendshape function, $B_S(\vec{\beta};\mathcal{S}) : \mathbb{R}^{|\vec{\beta}|} \rightarrow \mathbb{R}^{3N}$, to account for identity related shape variation, corrective pose blendshapes, $B_P(\vec{\theta};\mathcal{P}) : \mathbb{R}^{|\vec{\theta}|} \rightarrow \mathbb{R}^{3N}$, to correct pose deformations that cannot be explained solely by LBS, and expression blendshapes, $B_E(\vec{\psi};\mathcal{E}) : \mathbb{R}^{|\vec{\psi}|} \rightarrow \mathbb{R}^{3N}$, that capture facial expressions. A standard skinning function $W(\overline{\mathbf{T}},\mathbf{J},\vec{\theta},\mathcal{W})$ is applied to rotate the vertices of $\overline{\mathbf{T}}$ around joints $\mathbf{J} \in \mathbb{R}^{3K}$, linearly smoothed by blendweights $\mathcal{W} \in \mathbb{R}^{K\times N}$. Figure 2 visualizes the parametrization of FLAME, showing the degrees of freedom in shape (left), pose (middle), and expression (right).

More formally, the model is defined as

$$M(\vec{\beta},\vec{\theta},\vec{\psi}) = W(T_P(\vec{\beta},\vec{\theta},\vec{\psi}),\mathbf{J}(\vec{\beta}),\vec{\theta},\mathcal{W}), \tag{1}$$

where

$$T_P(\vec{\beta},\vec{\theta},\vec{\psi}) = \overline{\mathbf{T}} + B_S(\vec{\beta};\mathcal{S}) + B_P(\vec{\theta};\mathcal{P}) + B_E(\vec{\psi};\mathcal{E}) \tag{2}$$

denotes the template with added shape, pose, and expression offsets.

Since different face shapes imply different joint locations, the joints are defined as a function of the face shape $\mathbf{J}(\vec{\beta};\mathcal{J},\overline{\mathbf{T}},\mathcal{S}) = \mathcal{J}(\overline{\mathbf{T}} + B_S(\vec{\beta};\mathcal{S}))$, where $\mathcal{J}$ is a sparse matrix defining how to compute joint locations from mesh vertices. This joint regression matrix will be learned from training examples below. Figure 3 illustrates the learned location of the joints, which vary automatically with head shape.

Shape blendshapes: The variations in shape of different subjects are modeled by linear blendshapes as

$$B_S(\vec{\beta};\mathcal{S}) = \sum_{n=1}^{|\vec{\beta}|} \beta_n \mathbf{S}_n, \tag{3}$$

where $\vec{\beta} = [\beta_1,\cdots,\beta_{|\vec{\beta}|}]^T$ denotes the shape coefficients, and $\mathcal{S} = [\mathbf{S}_1,\cdots,\mathbf{S}_{|\vec{\beta}|}] \in \mathbb{R}^{3N\times|\vec{\beta}|}$ denotes the orthonormal shape basis, which will be learned below with PCA. The training of the shape space is described in Section 6.3.

Pose blendshapes: Let $R(\vec{\theta}) : \mathbb{R}^{|\vec{\theta}|} \rightarrow \mathbb{R}^{9K}$ be a function from a face/head/eye pose vector $\vec{\theta}$ to a vector containing the concatenated elements of all the corresponding rotation matrices. The pose blendshape function is defined as

$$B_P(\vec{\theta};\mathcal{P}) = \sum_{n=1}^{9K} \left(R_n(\vec{\theta}) - R_n(\vec{\theta}^*)\right)\mathbf{P}_n, \tag{4}$$

where $R_n(\vec{\theta})$ and $R_n(\vec{\theta}^*)$ denote the n-th element of $R(\vec{\theta})$, and $R(\vec{\theta}^*)$, respectively. The vector $\mathbf{P}_n \in \mathbb{R}^{3N}$ describes the vertex offsets from the rest pose activated by R_n, and the pose space $\mathcal{P} = [\mathbf{P}_1, \cdots, \mathbf{P}_{9K}] \in \mathbb{R}^{3N \times 9K}$ is a matrix containing all pose blendshapes. While the pose blendshapes are linear in R, they are non-linear with respect to $\vec{\theta}$ due to the non-linear mapping from $\vec{\theta}$ to rotation matrix elements. Details on how to compute the pose parameters from data are described in Section 6.1.

Expression blendshapes: Similar to the shape blendshapes, the expression blendshapes are modeled by linear blendshapes as

$$B_E(\vec{\psi}; \mathcal{E}) = \sum_{n=1}^{|\vec{\psi}|} \vec{\psi}_n \mathbf{E}_n, \tag{5}$$

where $\vec{\psi} = [\psi_1, \cdots, \psi_{|\vec{\psi}|}]^T$ denotes the expression coefficients, and $\mathcal{E} = [\mathbf{E}_1, \cdots, \mathbf{E}_{|\vec{\psi}|}] \in \mathbb{R}^{3N \times |\vec{\psi}|}$ denotes the orthonormal expression basis. The SMPL model does not have anything equivalent to these expression blendshapes, which are not driven by pose. The training of the expression space is described in Section 6.2.

Template shape: Note that the shape, pose, and expression blendshapes are all displacements from a template mesh $\overline{\mathbf{T}}$. We begin with a generic face template mesh and then learn the $\overline{\mathbf{T}}$ from scans along with the rest of the model. We also learn the blend weights, $\mathcal{W}$, as described below.

4 TEMPORAL REGISTRATION

Statistically modeling facial shape requires all training shapes to be in full vertex correspondence. Given sequences of 3D scans, for each scan, i, the registration process computes an aligned template $\mathbf{T}_i \in \mathbb{R}^{3N}$. The registration pipeline alternates between registering meshes while regularizing to a FLAME model and training a FLAME model from the registrations as shown in Figure 4. This alternating registration is similar to that used for human bodies [Bogo et al. 2014].

4.1 Initial model

The alternating registration process requires an initial FLAME model. As described in Section 3, FLAME consists of parameters for shape $\{\overline{\mathbf{T}}, \mathcal{S}\}$, pose $\{\mathcal{P}, \mathcal{W}, \mathcal{J}\}$, and expression $\mathcal{E}$, that require an initialization, which we then refine to fit registered scan data.

Shape: To get an initial head shape space, we extract the head region from the full-body registrations of SMPL [Loper et al. 2015] to the CAESAR dataset. We refine the mesh structure of the full-body SMPL template and adjust the topology to contain holes for the mouth and eyes. We then use deformation transfer [Sumner and Popović 2004], between the SMPL full-body shape registrations and our refined template, to get full-body registrations with the refined head template. Using these registered head templates, we compute the initial shape blend shapes, representing identity, by applying PCA to the vertices.

To make the registration process more stable, and to increase the visual quality of our model, we add eyeballs to our shape model. To initialize the eyes, we place the left eyeball using the eye region model of Woods et al. [2016] and regress its geometric center given a set of vertices around the left eye. Finally, we apply the same regressor to the equivalent (i.e. mirrored) set of vertices around the right eye.

Pose: The blendweights $\mathcal{W}$ and joint regressor $\mathcal{J}$ are initialized with weights defined manually by an artist. The initial vertices for the eyeball joint regressors are manually selected to result in joints close to the eyeball geometric center.

Expression: To initialize the expression parameters $\mathcal{E}$, we establish a correspondence through mesh registration between our head template and the artist generated FACS-based blendshape model of Li et al. [2013]. We then use deformation transfer, to transfer the expression blendshapes to our model. Although this initial expression basis does not conform to our requirements of orthogonality and expression realism, it is useful for bootstrapping the registration process.

4.2 Single-frame registration

The data to which we align our mesh includes 3D scan vertices, multi-view images (two for D3DFACS, three for our sequences), and camera calibrations. To align a sequence of an individual, we compute a personalized template and texture map of resolution 2048×2048 pixels as described later in Section 4.3.

Our model-based registration of a face scan consists of three steps.

Model-only: First, we estimate the model coefficients $\{\vec{\beta}, \vec{\theta}, \vec{\psi}\}$ that best explain the scan by optimizing

$$E(\vec{\beta}, \vec{\theta}, \vec{\psi}) = E_D + \lambda_L E_L + E_P, \tag{6}$$

with the data term

$$E_D = \lambda_D \sum_{\mathbf{v}_s} \rho\left(\min_{\mathbf{v}_m \in M(\vec{\beta}, \vec{\theta}, \vec{\psi})} \|\mathbf{v}_s - \mathbf{v}_m\| \right), \tag{7}$$

that measures the scan-to-mesh distance of the scan vertices $\mathbf{v}_s$ and the closest point in the surface of the model. The weight λ_D controls the influence of the data term. A Geman-McClure robust penalty function [Geman and McClure 1987], ρ, gives robustness to outliers in the scan.

The objective E_L denotes a landmark term, measuring the L2-norm distance between image landmarks and corresponding vertices on the model template, projected into the image using the known camera calibration. We use CMU Intraface [Xiong and la Torre 2013] to fully automatically predict 49 landmarks (Figure 5 left) in all multi-view camera images. We manually define the corresponding 49 landmarks in our template (see Figure 5 right). The weight λ_L describes the influence of the landmark term.

The prior term

$$E_P = \lambda_{\vec{\theta}} E_{\vec{\theta}} + \lambda_{\vec{\beta}} E_{\vec{\beta}} + \lambda_{\vec{\psi}} E_{\vec{\psi}} \tag{8}$$

regularizes the pose coefficients $\vec{\theta}$, shape coefficients $\vec{\beta}$, and expression coefficients $\vec{\psi}$ to be close to zero by penalizing their squared values.

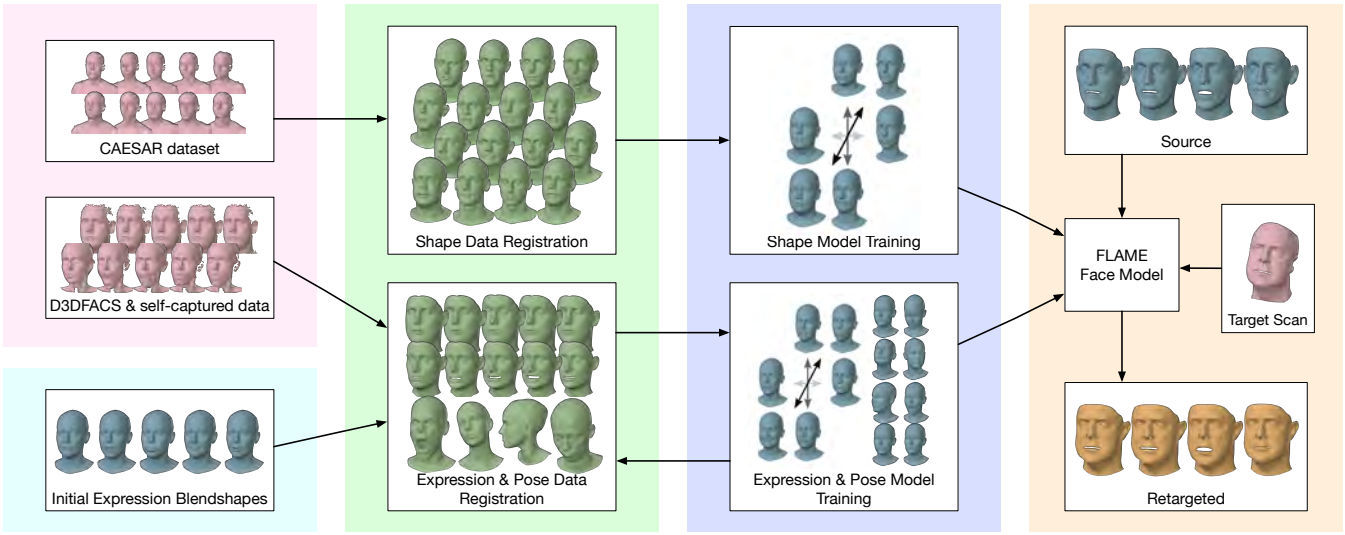

Fig. 4. Overview of the face registration, model training, and application to expression transfer.

Coupled: Second, we allow the optimization to leave the model space by optimizing

$$E(\mathbf{T}, \vec{\beta}, \vec{\theta}, \vec{\psi}) = E_D + E_C + E_R + E_P, \tag{9}$$

with respect to the model parameters $\{\vec{\beta}, \vec{\theta}, \vec{\psi}\}$ and the vertices of the template mesh $\mathbf{T}$, which is allowed to deform. In contrast to the model-only registration, E_D now measures the scan-to-mesh distance from the scan to the aligned mesh $\mathbf{T}$. The coupling term E_C constrains $\mathbf{T}$ to be close to the current statistical model by penalizing edge differences between $\mathbf{T}$ and the model $M(\vec{\beta}, \vec{\theta}, \vec{\psi})$ as

$$E_C = \sum_e \lambda_e \left\| \mathbf{T}_e - M(\vec{\beta}, \vec{\theta}, \vec{\psi})_e \right\|, \tag{10}$$

where $\mathbf{T}_e$ and $M(\vec{\beta}, \vec{\theta}, \vec{\psi})_e$ are the edges of $\mathbf{T}$ and $M(\vec{\beta}, \vec{\theta}, \vec{\psi})$, respectively, and λ_e denotes an individual weight assigned to each edge. The coupling uses edge differences to spread the coupling influence on single points across its neighbors. The optimization is performed simultaneously over $\mathbf{T}$ and model parameters in order to recover possible model errors in the first stage. The regularization term for each vertex $\mathbf{v}_k \in \mathbb{R}^3$ in $\mathbf{T}$ is the discrete Laplacian approximation [Kobbelt et al. 1998]

$$E_R = \frac{1}{N} \sum_{k=1}^{N} \lambda_k \|U(\mathbf{v}_k)\|^2, \tag{11}$$

with $U(\mathbf{v}) = \frac{\sum_{\mathbf{v}_r \in \mathcal{N}(\mathbf{v})} \mathbf{v}_r - \mathbf{v}}{|\mathcal{N}(\mathbf{v})|}$, where $\mathcal{N}(\mathbf{v})$ denotes the set of vertices in the one-ring neighborhood of $\mathbf{v}$. The regularization term avoids foldovers in the registration and hence makes the registration approach robust to noise and partial occlusions. The weight λ_k for each vertex allows for more regularization in noisy scan regions.

Texture-based: Third, we include a texture term E_T to obtain

$$E(\mathbf{T}, \vec{\beta}, \vec{\theta}, \vec{\psi}) = E_D + E_C + \lambda_T E_T + E_R + E_P, \tag{12}$$

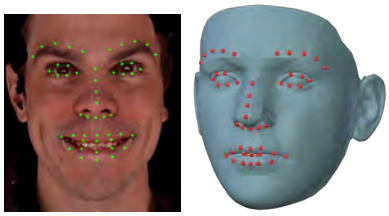

Fig. 5. Predicted 49 landmarks from the CMU Intraface landmark tracker [Xiong and la Torre 2013] (left) and the same landmarks defined on our topology (right).

where E_T measures the photometric error between real image I and the rendered textured image $\hat{I}$ of $\mathbf{T}$ from all V views as

$$E_T = \sum_{l=0}^{3} \sum_{v=1}^{V} \|\Gamma(I_l^{(v)}) - \Gamma(\hat{I}_l^{(v)})\|_F^2, \tag{13}$$

where $\|\mathbf{X}\|_F$ denotes the Frobenius norm of $\mathbf{X}$. Ratio of Gaussian filters, Γ [Bogo et al. 2014], help minimize the influence of lighting changes between real and rendered images. Further, as photometric errors are only meaningful for small displacements, a multi-level pyramid with four resolution levels are used during optimization to increase the spatial extent of the photometric error. The image I of resolution level l from view v is denoted by $I_l^{(v)}$.

4.3 Sequential registration

Our temporal registration approach uses a personalization phase that builds a personalized template for each subject in the database, which is then kept constant during tracking the facial performance.

Personalization: We assume that each captured sequence begins with a neutral pose and expression. During personalization, we use a coupled registration (Equation 9) and we average the results

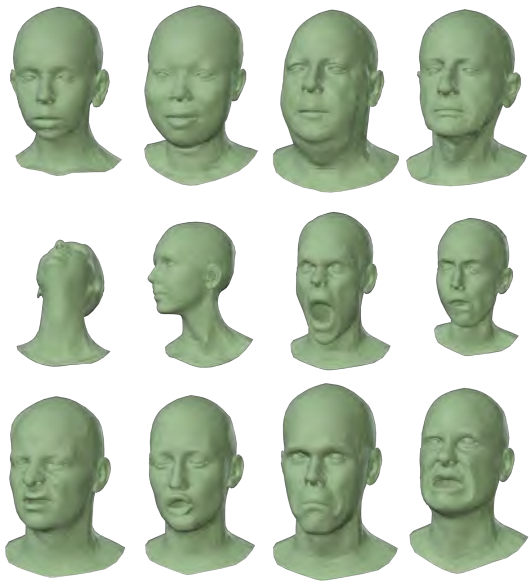

Fig. 6. Sample registrations. Top: shape data extracted from the CAESAR body database. Middle: sample registrations of the self captured pose data with head rotations around the neck (left) and mouth articulations (right). Bottom: samples registrations of the expression data from D3DFACS (left) and self captured sequences (right). The supplementary document shows further registrations.

$\mathbf{T}_i$ across multiple sequences to get a personalized template for each subject. We randomly select one of the $\mathbf{T}$ for each subject to generate a personalized texture map that is used later for texture-based registration. This personalization increases the stability of the registration, and improves the performance of the optimization, as it significantly reduces the number of parameters being optimized in each step.

Sequence fitting: During sequence fitting, we replace the generic model template $\overline{\mathbf{T}}$ in M (Equation 1) by the personalized template, and fix the $\vec{\beta}$ to zero. For each frame, we initialize the model parameters from the previous frame and use the single-frame registration 4.2. Given the registered sequences, we train a new FLAME model as described below and then iterate the registration procedure. We stop after four iterations as the visual improvement, compared to the registrations after three iterations, is only minor.

5 DATA

FLAME is trained from two large publicly available datasets and our self-captured sequences.

Our capture setup: For our self-captured sequences we use a multi-camera active stereo system (3dMD LLC, Atlanta). The capture system consists of three pairs of stereo cameras, three color cameras, three speckle projectors, and three white light LED panels. The system generates 3D meshes with an average of 45K vertices at 60fps. The color images are used to create a UV texture map for each frame and we use them to find image-based facial landmarks.

Training data: The identity shape parameters $\{\overline{\mathbf{T}}, \mathcal{S}\}$ are trained on the 3800 registered heads from the US and European CAESAR body scan database [Robinette et al. 2002]. The CAESAR database contains 2100 female and 1700 male static full-body scans, capturing large variations in shape (see Figure 6 top). The CAESAR scans are registered with a full-body SMPL model combined with our revised head template using a two-step registration approach. First, the global shape is initialized by a model-only registration with the initial model, followed by a coupled refinement (Section 4.2). The shape parameters are then trained on these registrations.

Training the pose parameters $\{\mathcal{P}, \mathcal{W}, \mathcal{J}\}$ requires training data that represent the full range of possible head motions, i.e. neck and jaw motions. As neither CAESAR, nor the existing 3D face databases, provide sufficient head pose articulation, we captured neck rotation and jaw motions of 10 subjects (see Figure 6 middle) to fill this gap. The jaw and mouth sequences are registered as described in Section 4. The head rotation sequences are registered using a coupled alignment, where only the vertices in the neck region are allowed to leave the model space, coupled to the model, while all other vertices stay in model space. This adds robustness to inevitable large facial occlusions when the head is turned. Overall, the pose parameters are trained on about 8000 registered heads.

The expression model, $\mathcal{E}$, uses two sources of training data, namely registrations of D3DFACS [Cosker et al. 2011] and self-captured sequences. All motion sequences are fully automatically registered with the registration approach described in Section 4, leading to a total number of $69,000$ registered frames (see Figure 6 bottom). In these 3D sequences, neighboring frames can be very similar. For efficiency in training, we consequently sample a subset of $21,000$ registered frames to train the model.

Test data: FLAME is evaluated quantitatively on three datasets. First we use the neutral scans of the BU-3DFE [Yin et al. 2006] database with its 3D face scans of 100 subjects with a large variety in ethnicity. Second we use self-captured sequences of seven subjects, performing different facial expressions, including the six prototypical expressions, talking sequences, and different facial action units. Note that the training and test subjects are fully disjoint. Third, we the 347 registered frames of the Beeler et al. [2011] sequence.

Implementation details: The registration framework is written in Python, using Numpy and Scikit-learn [Pedregosa et al. 2011] to compute PCA. All other model parameters are optimized by a gradient-based dogleg method [Nocedal and Wright 2006], where all gradients are computed using Chumpy [Loper and Black 2014] for automatic differentiation.

Parameter settings: Our registrations are obtained by a bootstrapping framework that alternates between model training and registration. During each iteration, we choose the parameters as follows:

We generally choose $\lambda_{\vec{\beta}} = \lambda_{\vec{\theta}} = 0.03$. For model-only registration, we set $\lambda_D \in \{100, 300\}$, and $\lambda_l = 0.002$, for coupled and texture based registration, we choose $\lambda_k = 10.0$. The coupling to the model varies depending on the regions of the face to deal with noise. We set $\lambda_e = 3.0$ for the face region (Figure 7) and $\lambda_e = 30.0$ and for all other vertices. For coupled registration, we further use $\lambda_D = 1000$, for texture-based registration $\lambda_D = 700$ and $\lambda_T = 0.1$. For

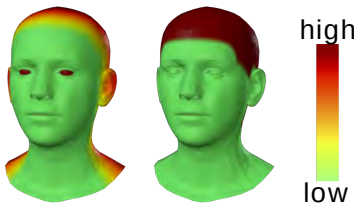

Fig. 7. Head regions with higher coupling edge weight (left) and higher Laplacian weight (right).

the third iteration, λ_e is reduced to 1.0 in the face region, for the fourth iteration to 0.3. For the fourth iteration, we further choose $\lambda_k = 100.0$ for the non-facial regions shown in the right of Figure 7.

A high coupling weight effectively prevents vertices leaving the model space and hence increases the robustness to noise. As the noise within a scan differs for different regions, i.e. it is significantly higher in hair regions, we use higher coupling weights for the back of the head, back of the neck, and the eyeballs (Figure 7 left). For regions like the forehead, a high coupling weight prevents the registration from effectively capturing the motion (e.g. when raising the eyebrows). A higher Laplacian weight (Figure 7 left) however, adds some smoothness and hence lowers the influence of noise, while allowing tangential motion to be captured.

Performance: Our registration takes about 155 s for one frame. (model-only (Eq. 6): 25 s; coupled (Eq. 9): 50 s; texture-based (Eq. 12): 80 s) on a single thread on a quad-core 3.2 GHz Intel Core i5 with 32 GB RAM

6 MODEL TRAINING

Given registered datasets for identity (Figure 6 top), pose (Figure 6 middle), and expression (Figure 6 bottom), the goal of training FLAME is to decouple shape, pose, and expression variations to compute the set of parameters $\Phi = \{\overline{\mathbf{T}}, \mathcal{S}, \mathcal{P}, \mathcal{E}, \mathcal{W}, \mathcal{J}\}$. To achieve this decoupling, the pose parameters $\{\mathcal{P}, \mathcal{W}, \mathcal{J}\}$, expression parameters $\mathcal{E}$, and shape parameters $\{\overline{\mathbf{T}}, \mathcal{S}\}$ are optimized one at a time using an iterative optimization approach that minimizes the reconstruction error of the training data. We use gender specific models Φ_f for female, and Φ_m for male, respectively.

6.1 Pose parameter training

There are two types of pose parameters in our model. First, there are parameters specific to each subject (indexed by $i \in \{1, \cdots P_{\text{subj}}\}$) such as personalized rest-pose templates $\mathbf{T}_i^P$ and person specific joints $\mathbf{J}_i^P$. Second, there are parameters spanning across subjects such as blendweights $\mathcal{W}$ and the pose blendshapes $\mathcal{P}$. The joint regressor $\mathcal{J}$ is learned to regress person specific joints $\mathbf{J}_i^P$ of all subjects from the personalized rest-pose templates $\mathbf{T}_i^P$.

The optimization of these parameters is done by alternating between solving for the pose parameters $\vec{\theta}_j$ of each registration j, optimizing the subject specific parameters $\{\mathbf{T}_i^P, \mathbf{J}_i^P\}$, and optimizing the global parameters $\{\mathcal{W}, \mathcal{P}, \mathcal{J}\}$. The objective function being optimized consists of a data term E_D that penalizes the squared Euclidean reconstruction error of the training data, a regularization term $E_{\mathcal{P}}$ that penalizes the Frobenius norm of the pose blendshapes, and a regularization term $E_{\mathcal{W}}$ that penalizes large deviations of the blendweights from their initialization. The weighting of the regularizers $\{E_{\mathcal{P}}, E_{\mathcal{W}}\}$ is a tradeoff between closely resembling the training data and keeping the parameters general. Hence, the regularizers prevent FLAME from overfitting to the training data, and make it more general. The method and objectives used for the optimization of joint regressors, pose and shape parameters are described in more detail by the SMPL body model [Loper et al. 2015], as we adapted their approach to represent pose and shape for FLAME.

In absence of a subject specific template $\mathbf{T}_i^P$, the initial estimation of the pose coefficients $\vec{\theta}$ while training the pose space is done using an initial average template. To be robust with respect to large variations in shape, this is done by minimizing the edge differences between the template and each registration.

To avoid $\mathbf{T}_i^P$ and $\mathbf{J}_i^P$ being affected by strong facial expressions, expression effects are removed when solving for $\mathbf{T}_i^P$ and $\mathbf{J}_i^P$. This is done by jointly solving for pose $\vec{\theta}$ and expression parameters $\vec{\psi}$ for each registration, subtracting B_E (Equation 5), and solving for $\mathbf{T}_i^P$ and $\mathbf{J}_i^P$ on those residuals.

6.2 Expression parameter training

Training the expression space $\mathcal{E}$ requires expressions to be decoupled from pose and shape variations. This is achieved by first solving for the pose parameters $\vec{\theta}_j$ of each registration, and removing the pose influence by applying the inverse transformation entailed by $M(\vec{0}, \vec{\theta}, \vec{0})$ (Equation 1); where $\vec{0}$ is a vector of zero-valued coefficients. We call this step "unposing" and call the vertices resulting from unposing the registration j as $\mathbf{V}_j^U$. As we want to model expression variations from a neutral expression, we assume that a registration defining the neutral expression is given for each subject. Let $\mathbf{V}_i^{NE}$ denote the vertices of the neutral expression of subject i, also unposed. To decouple the expression variations from the shape variations, we compute expression residuals $\mathbf{V}_j^U - \mathbf{V}_{s(j)}^{NE}$ for each registration j, where $s(j)$ is the subject index j. We then compute the expression space $\mathcal{E}$ by applying PCA to these expression residuals.

6.3 Shape parameter training

Training the shape parameters consists of computing template $\overline{\mathbf{T}}$ and shape blendshapes $\mathcal{S}$ for the registrations in the shape dataset. Similarly as before, effects of pose and expression are removed from all training data, to ensure the decoupling of pose, expression, and shape. The template $\overline{\mathbf{T}}$ is then computed as the mean of these expression- and pose-normalized registrations, the shape blendshapes $\mathcal{S}$ are formed by the first $|\vec{\beta}|$ principal components computed using PCA.

6.4 Optimization structure

The training of FLAME is done iteratively by solely optimizing pose, expression, or shape parameters, while keeping the other parameters fixed. Due to the high capacity and flexibility of the expression space formulation, pose blendshapes should be trained before expression parameters in order to avoid expression overfitting.

7 RESULTS

We evaluate the quality of our sequence registration process and the FLAME models learned from these registrations. Comparisons to Basel Face Model and FaceWarehouse model show that FLAME is significantly more expressive. Additionally we show how FLAME can be used to fit 2D image data and for expression transfer. Please see the **supplementary video** for more details.

Visualization: We use a common color coding to present all results throughout the entire document. Input data such as static or dynamic 3D face scans are shown in a light red color. Meshes that are within the space of a statistical model, obtained by model-only registration (Section 4.2) or by sampling the latent space of a model, are shown in blue. For comparison, we use the same color to visualize results of FLAME, Basel Face Model, or FaceWarehouse model. Meshes, obtained by leaving the shape space in a coupled or texture-based alignment (Section 4.2) are visualized in light green.

FLAME is a fully articulated head model (see Figure 2). Nevertheless, most training and test scans only capture the face region. To facilitate comparison between methods, in such cases we show registrations of similar facial regions only. For comparisons to scans with clean outer boundary and without holes (e.g. Figures 16), we use the background of the scan images to mask the region of interest. For scans with noisy outline and holes (e.g. Figure 11) we use a common pre-defined vertex mask to visualize all registrations.

7.1 Registration quality

Registration process: Our registration process contains three steps: a model-only fit, a coupled fit, and a texture-based refinement. Figure 8 visualizes the registration results of each optimization step. The model-only step serves the initialization of the expression, but it is unable to capture all personalized details. After coupled alignment, the registration tightly fits the surface of the scan but the synthesized texture reveals misalignments at the mouth, nose, and eyebrows. While the texture-based registration slightly raises the geometric error across the face, it visually improves the registration around the mouth, nose, and eyebrow regions while reducing the sliding within the surface.

Note, we do not explicitly model lighting for the synthesized image, which causes visual differences compared to the original image due to cast shadows (e.g. seen at the cheeks). Using a Ratio of Gaussians for filtering alleviates the influence of lighting changes in our optimization setup.

Alternating registration: Figure 9 shows representative results for each of the alternating registration iterations. While the registration is unable to capture the facial expressions properly in the first iteration, after more iterations, the quality of the registration improves.

Quantitative evaluation: Figure 10 (top) visualizes the median per-vertex distance to the scan. The distance is measured across all 69,000 registered frames of the D3DFACS database and our self captured sequences (top) and the 347 registered frames of the Beeler et al. [2011] sequence.

For the registered training data (Figure 10 top), within the face region (excluding the eyeballs), 60% of the vertices have a median distance less than 0.2mm, 90% are closer than 0.5mm. Visible regions of higher distance are mostly caused by missing data (at the neck, below the chin, or at the ears) or noise in the scans (at the eyebrows, around the eyes). As described in Section 5, our registration framework uses higher Laplacian weights in non-face regions to increase the robustness to noise and partial occlusions in the scans. While not causing visual artifacts in the registrations, this transition between the face and non-face part causes a slightly enlarged error at the boundary of the mask, noticeable at the forehead.

The goal of our registration framework is to fully-automatically register a large set of sequences (> 600) from different sources (i.e. D3DFACS and self-captured sequences). For robustness to self-cast shadows and lighting changes, the influence of the photometric error (Equation 12) has a low weight ($w_T = 0.1$). Due to this, our registrations are not entirely free of within-surface drift, especially in regions without salient features (i.e. forehead, cheeks, neck). The bottom of Figure 10 evaluates the within-surface drift of our registration on the publicly available Beeler et al. sequence. While the distance between our registrations and the Beeler et al. scans is small (bottom left), measuring the distance between our registrations and their ground-truth registration reveals some within-surface drift (bottom right). Note, since the Beeler et al. data are with uniform lighting, one could use our registration method with a higher weighted photometric error, potentially further lowering the drift error.

Qualitative evaluation: Figure 11 shows sample registrations of the D3DFACS dataset (top) and our self-captured sequences (bottom). For all sequences, the distance between our registration and the scan surface is small, and our registration captures the expression. Note that our registration is able to track even subtle motions such as eye blinks well as can be seen in top row of Figure 11.

7.2 Model quality

A good statistical model should ideally be compact and generalize well to new data, while staying specific to the object class of the model. A common way to quantify these attributes is to measure the compactness, generalization, and specificity [Davies et al. 2008, Chapter 9.2] of the model. These measurements have previously been used to evaluate statistical models of various classes of objects, including 3D faces (e.g. [Bolkart and Wuhrer 2015; Booth et al. 2017; Brunton et al. 2014]). These evaluations provide a principled way to determine the model dimensions that preserve a large amount of the data variability without overfitting to the training data.

Compactness: A statistical model should describe the training data with few parameters. Compactness measures the amount of variability present in the training data, that is captured by the model. The compactness for a given number of k components is $C(k) = \sum_{i=1}^{k} \lambda_i / \sum_{i=1}^{rank(\mathbf{D})} \lambda_i$, where λ_i is the i-th eigenvalue of the data covariance matrix $\mathbf{D}$. The compactness of FLAME is independently evaluated for identity and expression, by computing $C(k)$ for a varying number of components.

Generalization: A statistical model ideally generalizes from the samples in the training data to arbitrary valid samples of the same class of objects. Generalization measures the ability of the model to

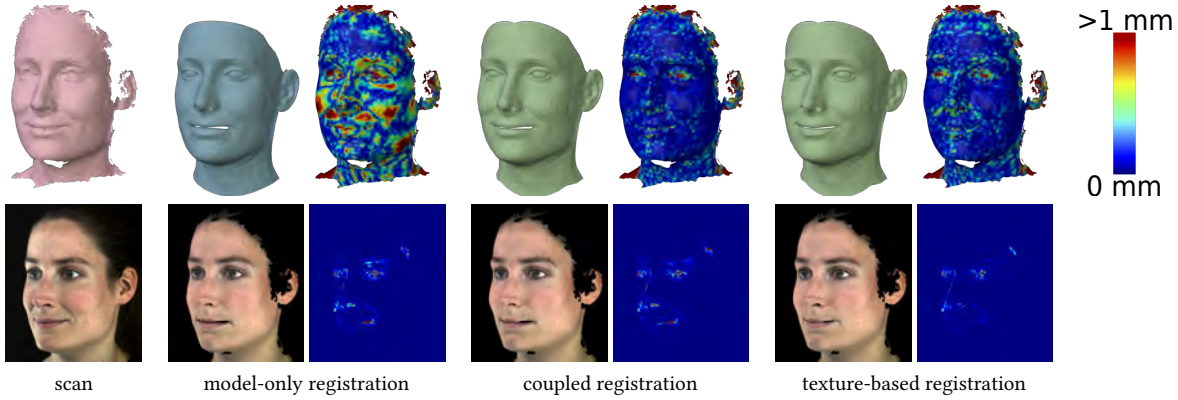

Fig. 8. Results of the model-only, coupled, and texture-based registration steps for one scan. Top: scan, registrations, and scan-to-mesh distance for each registration visualized color-coded on the scan. Bottom: original texture image, synthesized texture image for each step, and the corresponding photometric errors.

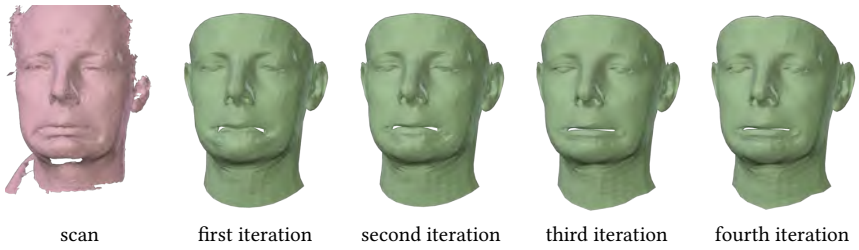

Fig. 9. Results of each iteration of the alternating registration approach.

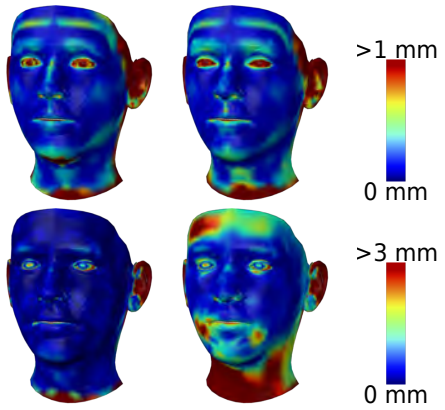

Fig. 10. Median per-vertex distance between registration and the scan surface. Top: Distance measure across all frames of all female (left) and male (right) training sequences. Bottom: Distance measure across all registered frames for the Beeler et al. [2011] sequence (left) and the ground-truth error (right) measuring the within-surface drift. The supplementary video shows the full registration sequence.

represent unseen shapes of the same object class. The generalization ability is commonly quantified by fitting the model with a varying number of components to data excluded from the model training, and measuring the fitting error. The identity space of FLAME is evaluated on the neutral BU-3DFE data, registered using a coupled alignment. The expression space is evaluated on self-captured test sequences, registered with the texture-based registration framework. During evaluation of the identity space, i.e. for a varying number of identity shape components, the number of expression components is fixed to 100. For evaluation of the expression space, the number of shape parameters is fixed to 300, accordingly. For each model-fit, the average vertex distance to the registration is reported as fitting error.

Specificity: A statistical model is required to be specific to the modeled class of objects, by only representing valid samples of this object class. To evaluate the specificity of the identity and expression space, we randomly draw 1000 samples from a Gaussian distribution for a varying number of identity or expression coefficients, and reconstruct the sample shape using Equation 1. The specificity error is measured as the average distance to the closest training shape. For identity space evaluation, the expression parameters are kept at zero; for expression evaluation, the identity parameters are zero, accordingly.

Quantitative evaluation: Figure 12 shows compactness, generalization, and specificity, independently evaluated for the identity and expression space. With 90 identity components our model captures 98% of the data variability, and with 300 components effectively 100%. The generalization error gradually declines for up to 300

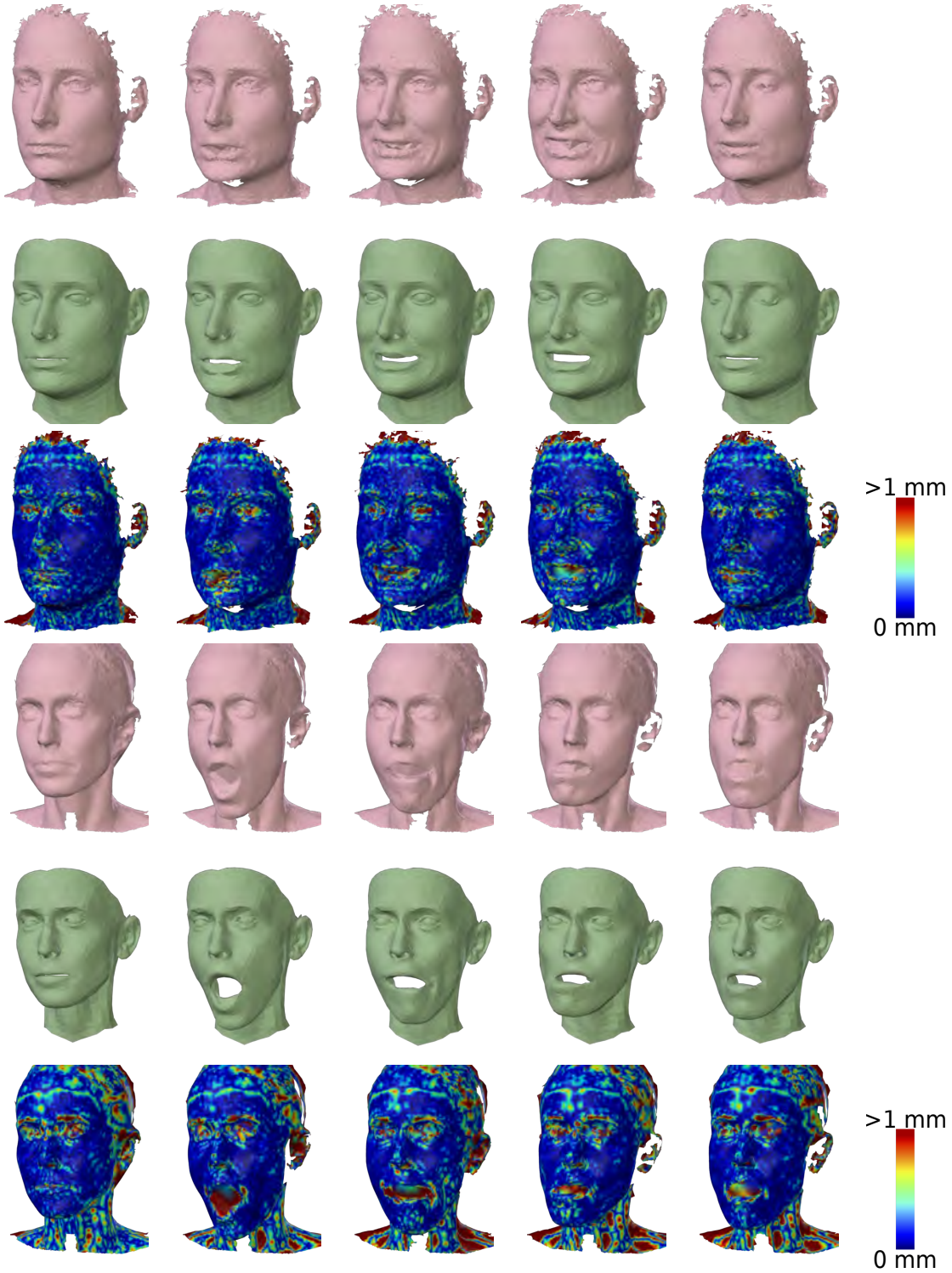

Fig. 11. Sample frames, registrations, and scan-to-mesh distance of one sequences of the D3DFACS database (top) and one sequence of our self-captured sequence (bottom). The supplementary document shows further registrations.

identity components, while specificity not increase significantly. Consequently, we use models with 90 and 300 identity components throughout our evaluations. We denote these with *FLAME 90* and *FLAME 300*, respectively. For expression, we choose 100 components, representing 98% of the data variability.

Qualitative evaluation: Figure 13 qualitatively evaluates the influence of a varying number of identity components for fitting the neutral BU-3DFE face scans (the supplementary document shows more samples). The error measures, for each scan vertex, the distance to the closest point in the surface of the registration. While *FLAME 49* fits the global shape of the scan well, it is unable to capture localized person specific details. Increasing the number of components increases the ability of the model to reconstruct localized details. *FLAME 300* leads to registrations with an error that is close to zero millimeters in most facial regions.

FLAME models head and jaw motions as joint rotations. Figure 14 shows the influence of the trained pose blendshapes. The pose blendshapes recreate realistic neck details when turning the head and stretch the cheeks when opening the mouth. The learned pose blendshapes result in significantly more realism than LBS.

7.3 Comparison to state-of-the-art

We compare FLAME to the Basel Face Model (BFM) [Paysan et al. 2009] and FaceWarehouse model (FW) [Cao et al. 2014]. We evaluate the ability of each model to account for unseen data by fitting them to static and dynamic 3D data not part of the training; in all cases we use the same model-fitting framework. BFM is trained from 200 neutral expression shapes and all 199 identity components are available. FW is learned from 150 shapes but the model only includes 50 identity components plus 46 expression components.

Identity: The identity space is evaluated by fitting the models to the neutral BU-3DFE scans, initializing with the landmarks provided with the database. For a fair comparison to FW and BFM, FLAME is constrained to use comparable dimensions. Consequently we only make use of 49 FLAME shape components for comparison to FW and 198 components for comparison with BFM (we subtract one component since we select the appropriate gender). We further show the expressiveness of FLAME with 90 and 300 components.

Figure 15 shows the cumulative scan-to-mesh distance computed over all model-fits to the neutral BU-3DFE scans. With the same number of parameters, for *FLAME 49*, 74% of the scan vertices have distance lower than 0.5mm, compared to 69% for *BFM 50* or 67% by FW. Compared to BFM with all components, for *FLAME 198*, 94% of the vertices have a distance less than 0.5mm, compared to 92% for *BFM Full*. With 300 components, *FLAME 300* fits 96% of the vertices with a distance of less than 0.5mm.

Figure 16 compares the models visually (the supplementary document shows more examples). Compared to FLAME, BFM introduces high-frequency details that make the fits look more realistic. Nevertheless, the comparison with the scans reveals that these details are hallucinated and spurious, as they come from people in the dataset, rather than from the scans. While lower-resolution and less detailed, FLAME is actually more accurate. Note, since FLAME contains modeled eyeballs, the eye region looks more realistic than the closed surface of BFM or the empty space of FW.

Expression: The ability to capture real facial expressions is evaluated by fitting FW and FLAME to our self-captured high-resolution dynamic test sequences (see Section 5). For comparison, we first compute a personalized shape space for each model per sequence by only optimizing the identity parameters, keeping the expression fixed to a neutral expression. For the rest of the sequence, only the expression and pose are optimized, initialized by landmarks, while the identity parameters are kept fixed. To remove one source of error caused by noisy landmarks, we register all test sequences with our texture-based registration framework and extract the same set of landmarks as shown in Figure 5. As for the identity evaluation, we constrain FLAME to be of comparable dimension to FW for a fair comparison. We use 49 components for identity and as for FW, 46 components for expression and pose; i.e. we use 43 components for expression, and 3 degrees of freedom for the jaw rotation.

Figure 17 compares the median of the per-vertex distance to the scans, measured across all registered frames of the test data. For FW, 50% of all vertices in the face region have a distance lower than 1.0mm, compared to 67% for *FLAME 49*, 73% for *FLAME 90*, and 75% for *FLAME 300*. With the same number of parameters, FLAME fits the data closer than FW.

Figure 18 visualizes examples from this experiment. While FW is able to perform the expression for the first sequence (top row), FLAME gives a more natural looking result with a lower error. For the second sequence (bottom row), FW is unable to reconstruct the widely open mouth. As FLAME models the mouth opening with a rotation, it better fits this extreme expression. As Figure 17 shows, if we used more components, FLAME would significantly outperform FW.

7.4 Shape reconstruction from images

FLAME is readily usable to reconstruct 3D faces from single 2D images. For comparison to FaceWarehouse, we fit both models to 2D image landmarks by optimizing the L2-norm distance between image landmarks and corresponding model vertices, projected into the image using the known camera calibration. Unlike other facial landmarks, the face contour does not correspond to specific 3D points. Therefore, the correspondences are updated based on the silhouette of the projected 3D face as described in Cao et al. [2014]. The input landmarks are manually labeled in the same format as in FaceWarehouse. As in Section 7.3, we use 49 components for identity and 46 components for expression and pose (43 for expression and 3 for jaw pose), for a fair comparison.

Figure 19 shows the 2D landmark fitting using both models. FLAME better reconstructs the identity and expression. To quantify the error in the fit shown in Figure 19, we measure the distance from the fitted mesh to the ground truth scan. Due to the challenges in estimating depth from merely 2D landmarks, we firstly rigidly align the fitted mesh to scan using precomputed 3D landmarks, and then measure the distances. For qualitative comparison, we further show the fitted mesh from a novel view for better comparison to the ground truth scan. As shown in Figure 19, FLAME has lower

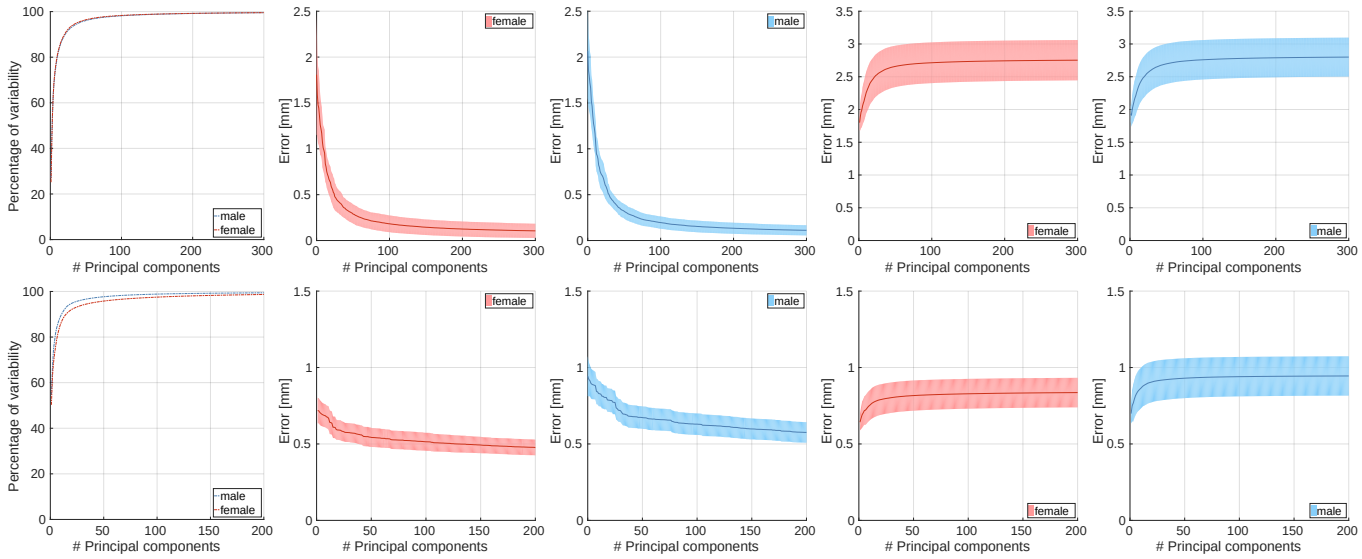

Fig. 12. Quantitative evaluation of identity shape space (top) and expression space (bottom) of the female and male FLAME models. From left to right: compactness, generalization female, generalization male, specificity female, and specificity male.

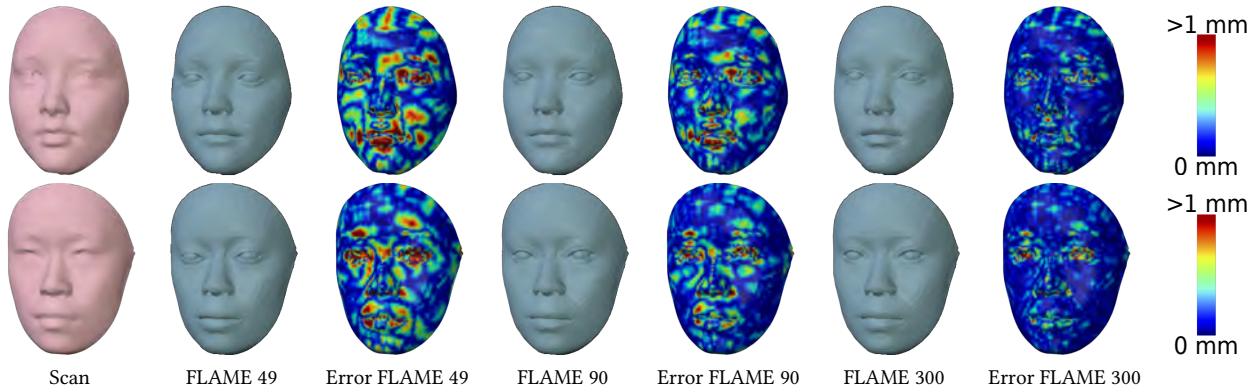

Fig. 13. Expressiveness of the FLAME identity space for fitting neutral scans of the BU-3DFE face database with a varying number of identity components. The supplementary document shows further examples.

3D error, suggesting that FLAME may provide a better prior for estimating 3D facial shape from 2D image features.

7.5 Expression transfer

FLAME can easily be used to synthesize new motion sequences, e.g. by transferring the facial expression from a source actor to a target actor, while preserving the subject-specific details of the target face. This transfer is performed in three steps. First, the source sequence is registered with the proposed registration framework (Section 4.3) to compute the pose and expression coefficients $\{\vec{\theta}_s, \vec{\psi}_s\}$ for each frame of the source sequence. Second, a coupled registration (Section 4.2) is used to compute a personalized template $\mathbf{T}_t$ for the target scan. Finally, replacing the average model template $\overline{\mathbf{T}}$ by the personalized target template $\mathbf{T}_t$ results in a personalized FLAME model $M_t(\vec{\beta}, \vec{\theta}, \vec{\psi})$ of the target actor. The result of the expression transfer is then the model reconstruction $M_t(\vec{0}, \vec{\theta}_s, \vec{\psi}_s)$ using Equation 1.

Figure 20 shows the expression transfer between two subjects in our test dataset, while Figure 1 shows transfer to a high-resolution scan from Beeler et al. [2011]. The **supplemental material** shows additional results.

7.6 Discussion

While FLAME moves closer to custom head models in realism, it still lacks the detail needed for high-quality animation. Fine-scale details such as wrinkles and pores are subject-specific and hence (i.e. due to the missing inter-subject correspondence) are not well modeled by a generic face model. A different approach (e.g. via deep learning)

Fig. 14. Influence of the pose blendshapes for different actuations of the neck and yaw joints in a rotational manner. Visualization of FLAME without (top) and with (bottom) activated pose blendshapes.

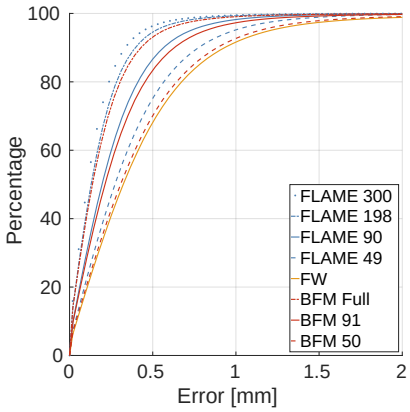

Fig. 15. Cumulative scan-to-mesh distance computed over all model-fits of the neutral BU-3DFE scans.

could be used to infer high-frequency and non-linear details, but this is beyond the scope of this work.

The surface-based decoupling of shape, pose, and expression variations (Sec. 6) requires a lot of diverse training data (Sec. 5). Exploiting anatomical constraints, i.e. by using a rigid stabilization method [Beeler and Bradley 2014], could further improve the decoupling, but this would require a significant amount of work to handle the large amounts of training data as reasoning about the underlying skull is needed.

Here we learned expression blendshapes and showed that they capture real facial expressions better than those of FaceWarehouse. We argue that these capture important correlations across the face and result in natural looking expressions. Still animators may prefer more semantic, or localized, controls. Consequently one could learn a mapping from our space to semantic attributes as shown in other works [Allen et al. 2003; Hasler et al. 2009; Vlasic et al. 2005] or train a localized space as proposed by Neumann et al. [2013] from our provided expression registration, and replace the global expression space of FLAME with a local one.

Here we found that modeling the eyes improved alignment and the final model; we plan to do something similar for mouths by explicitly modeling them. FLAME is connected to a neck, which has the same topology as the SMPL body model. In the future we will combine the models, which will enable us to capture both the body and face together. Since we have eyes in the model, we also plan to integrate eye tracking.

One could also personalize our model to a particular actor, restricting the expression space based on past performance. Our model could also be fit to sparse marker data, enabling facial performance capture using standard methods. Future work should also fit the model to images and video sequences by replacing simpler models in standard methods. Finally, images can be used to add more shape detail from shading cues as in recent work [Garrido et al. 2016].

8 CONCLUSION

Here we trained a new model of the face from around 33, 000 3D scans from the CAESAR body dataset, the D3DFACS dataset, and self captured sequences. To do so, we precisely aligned a template mesh to all static and dynamic scans and will make the alignments of the D3DFACS dataset available for research purposes. We defined the FLAME model using a PCA space for identity shape, simple rotational degrees of freedom and linear blend skinning for the neck, jaw, and eyeballs, corrective blendshapes for these rotations, and global expression blendshapes. We show that the learned model is significantly more expressive and realistic than the popular FaceWarehouse model and the Basel Face Model. We compare the models by fitting to static 3D scans and dynamic 3D sequences of novel subjects using the same optimization method. While significantly more accurate, FLAME has many fewer vertices, which also makes it more appropriate for real-time applications. Unlike over-complete representations associated with standard manual blendshapes, ours are easier to optimize because they are orthogonal. The model is designed to be compatible with existing rendering systems and is available for research purposes [FLAME 2017].

ACKNOWLEDGMENTS

We thank T. Alexiadis, A. Keller, and J. Márquez for help with the data acquisition, S. Saito and C. Laidlaw for support with the evaluation, Y. Huang, A. Osman, and N. Mahmood for helpful discussions, T. Zaman for the voice recording, and A. Quiros-Ramirez for help with the project page. We further thank D. Cosker for his advice and for allowing us to publish D3DFACS registrations.

REFERENCES

O. Alexander, M. Rogers, W. Lambeth, M. Chiang, and P. Debevec. 2009. The Digital Emily Project: Photoreal Facial Modeling and Animation. In *SIGGRAPH 2009 Courses*. 12:1–12:15.

B. Allen, B. Curless, and Z. Popović. 2003. The space of human body shapes: Reconstruction and parameterization from range scans. In *Transactions on Graphics (Proceedings of SIGGRAPH)*, Vol. 22. 587–594.

B. Allen, B.n Curless, Z. Popović, and A. Hertzmann. 2006. Learning a Correlated Model of Identity and Pose-dependent Body Shape Variation for Real-time Synthesis. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '06)*. 147–156.

B. Amberg, R. Knothe, and T. Vetter. 2008. Expression Invariant 3D Face Recognition with a Morphable Model. In *International Conference on Automatic Face Gesture Recognition*. 1–6.

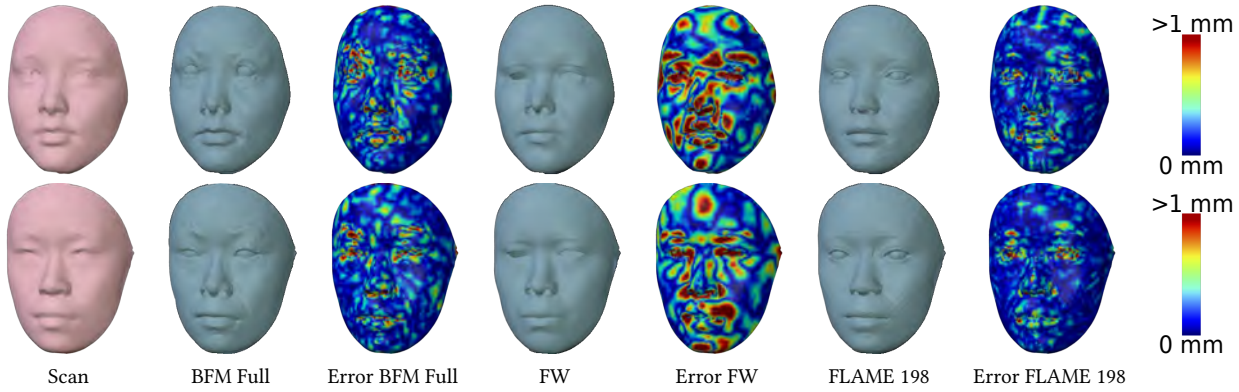

Fig. 16. Comparison of Basel Face Model (BFM) [Paysan et al. 2009], FaceWarehouse model [Cao et al. 2014] and FLAME for fitting neutral scans of the BU-3DFE database. The supplementary document shows further examples.

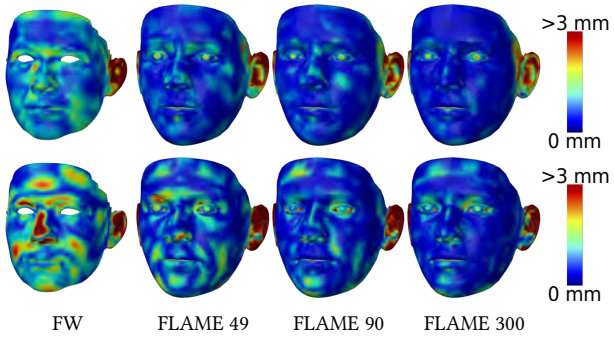

Fig. 17. Median per-vertex distance between registration to the scan surface, measured across all frames of the test data. Top: female data. Bottom: male data.

B. Amberg, S. Romdhani, and T. Vetter. 2007. Optimal Step Nonrigid ICP Algorithms for Surface Registration. In *Conference on Computer Vision and Pattern Recognition*. 1–8.

D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. 2005. SCAPE: Shape Completion and Animation of People. *Transactions on Graphics (Proceedings of SIGGRAPH)* 24, 3 (2005), 408–416.

T. Beeler and D. Bradley. 2014. Rigid Stabilization of Facial Expressions. *Transactions on Graphics (Proceedings of SIGGRAPH)* 33, 4 (2014), 44:1–44:9.

T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. Beardsley, C. Gotsman, R. W. Sumner, and M. Gross. 2011. High-quality passive facial performance capture using anchor frames. *Transactions on Graphics (Proceedings of SIGGRAPH)* 30, 4 (2011), 75:1–75:10.

V. Blanz and T. Vetter. 1999. A morphable model for the synthesis of 3D faces. In *SIGGRAPH*. 187–194.

F. Bogo, M. J. Black, M. Loper, and J. Romero. 2015. Detailed Full-Body Reconstructions of Moving People from Monocular RGB-D Sequences. In *International Conference on Computer Vision*. 2300–2308.

F. Bogo, J. Romero, M. Loper, and M.J. Black. 2014. FAUST: Dataset and Evaluation for 3D Mesh Registration. In *Conference on Computer Vision and Pattern Recognition*. 3794–3801.

T. Bolkart and S. Wuhrer. 2015. A Groupwise Multilinear Correspondence Optimization for 3D Faces. In *International Conference on Computer Vision*. 3604–3612.

J. Booth, A. Roussos, A. Ponniah, D. Dunaway, and S. Zafeiriou. 2017. Large Scale 3D Morphable Models. *International Journal of Computer Vision* (2017), 1–22.

J. Booth, A. Roussos, S. Zafeiriou, A. Ponniahy, and D. Dunaway. 2016. A 3D Morphable Model Learnt from 10,000 Faces. In *Conference on Computer Vision and Pattern Recognition*. 5543–5552.

S. Bouaziz, Y. Wang, and M. Pauly. 2013. Online modeling for realtime facial animation. *Transactions on Graphics (Proceedings of SIGGRAPH)* 32, 4 (2013), 40:1–40:10.

A. Bronstein, M. Bronstein, and R. Kimmel. 2008. *Numerical Geometry of Non-Rigid Shapes* (1 ed.). Springer Publishing Company, Incorporated.

A. Brunton, T. Bolkart, and S. Wuhrer. 2014. Multilinear wavelets: A statistical shape space for human faces. In *European Conference on Computer Vision*. 297–312.

C. Cao, D. Bradley, K. Zhou, and T. Beeler. 2015. Real-time High-fidelity Facial Performance Capture. *Transactions on Graphics (Proceedings of SIGGRAPH)* 34, 4 (2015), 46:1–46:9.

C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou. 2014. FaceWarehouse: A 3D facial expression database for visual computing. *Transactions on Visualization and Computer Graphics* 20, 3 (2014), 413–425.

Y. Chen, D. Robertson, and R. Cipolla. 2011. A Practical System for Modelling Body Shapes from Single View Measurements. In *British Machine Vision Conference*. 82.1–82.11.

D. Cosker, E. Krumhuber, and A. Hilton. 2011. A FACS valid 3D dynamic action unit database with applications to 3D dynamic morphable facial modeling. In *International Conference on Computer Vision*. 2296–2303.

R. Davies, C. Twining, and C. Taylor. 2008. *Statistical Models of Shape: Optimisation and Evaluation*. Springer.

L. Dutreve, A. Meyer, and S. Bouakaz. 2011. Easy acquisition and real-time animation of facial wrinkles. *Computer Animation and Virtual Worlds* 22, 2-3 (2011), 169–176.

P. Ekman and W. Friesen. 1978. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press.

C. Ferrari, G. Lisanti, S. Berretti, and A. Del Bimbo. 2015. Dictionary Learning based 3D Morphable Model Construction for Face Recognition with Varying Expression and Pose. In *International Conference on 3D Vision*. 509–517.

FLAME. 2017. http://flame.is.tue.mpg.de. (2017).

P. Garrido, L. Valgaerts, C. Wu, and C. Theobalt. 2013. Reconstructing detailed dynamic face geometry from monocular video. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 32, 6 (2013), 158:1–158:10.

P. Garrido, M. Zollhöfer, D. Casas, L. Valgaerts, K. Varanasi, P. Perez, and C. Theobalt. 2016. Reconstruction of Personalized 3D Face Rigs from Monocular Video. *Transactions on Graphics (Presented at SIGGRAPH 2016)* 35, 3 (2016), 28:1–28:15.

S. Geman and D. E. McClure. 1987. Statistical methods for tomographic image reconstruction. *Proceedings of the 46th Session of the International Statistical Institute, Bulletin of the ISI* 52 (1987).

N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel. 2009. A Statistical Model of Human Pose and Body Shape. *Computer Graphics Forum* (2009).

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

A. E. Ichim, S. Bouaziz, and M. Pauly. 2015. Dynamic 3D Avatar Creation from Hand-held Video Input. *Transactions on Graphics (Proceedings of SIGGRAPH)* 34, 4 (2015), 45:1–45:14.

I. Kemelmacher-Shlizerman and S. M. Seitz. 2011. Face Reconstruction in the Wild. In *International Conference on Computer Vision*. 1746–1753.

L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel. 1998. Interactive Multi-resolution Modeling on Arbitrary Meshes. In *SIGGRAPH*. 105–114.

Y. Kozlov, D. Bradley, M. Bächer, B. Thomaszewski, T. Beeler, and M. Gross. 2017. Enriching Facial Blendshape Rigs with Physical Simulation. *Computer Graphics Forum* (2017).

H. Li, T. Weise, and M. Pauly. 2010. Example-based facial rigging. *Transactions on Graphics (Proceedings of SIGGRAPH)* 29, 4 (2010), 32:1–32:6.

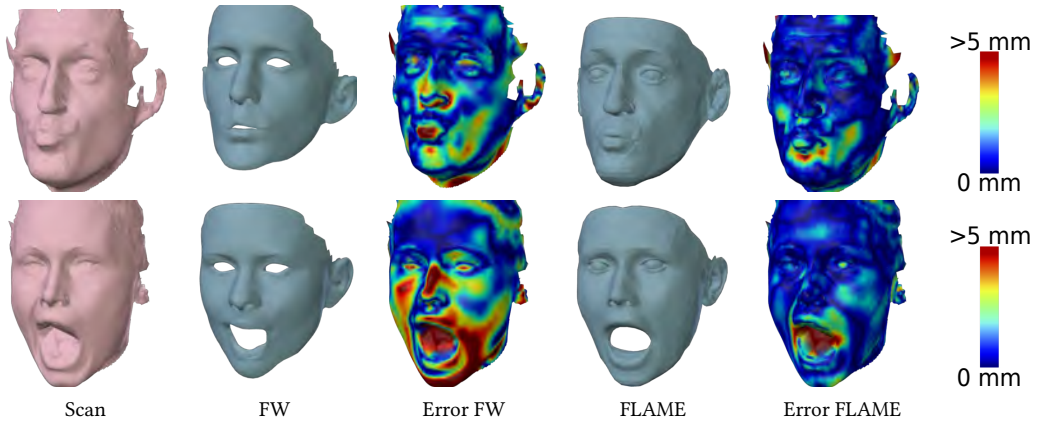

Fig. 18. Reconstruction quality from high-resolution motion sequences compared to FaceWarehouse (FW). Intermediate frames of three motion sequences. FLAME is restricted to have the same number of parameters as FW.

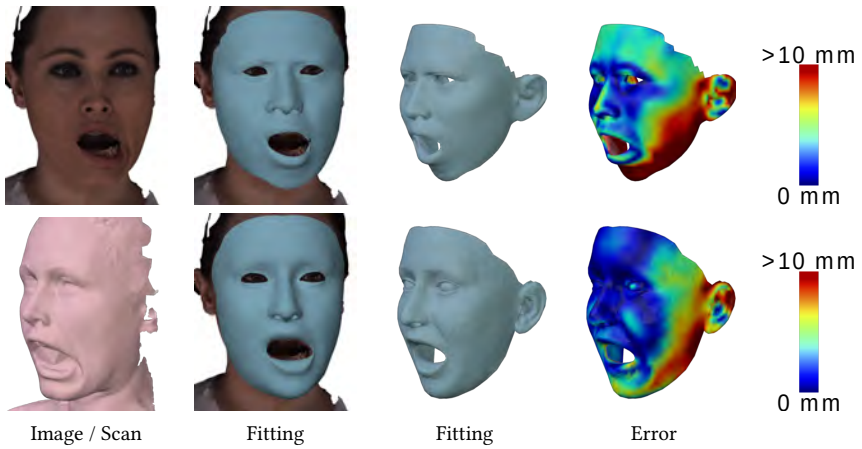

Fig. 19. Comparison of FaceWarehouse model (top) and FLAME (bottom) for 3D face fitting from single 2D image. Note, that the scan (pink) is only used for evaluation. The supplementary document shows further examples.

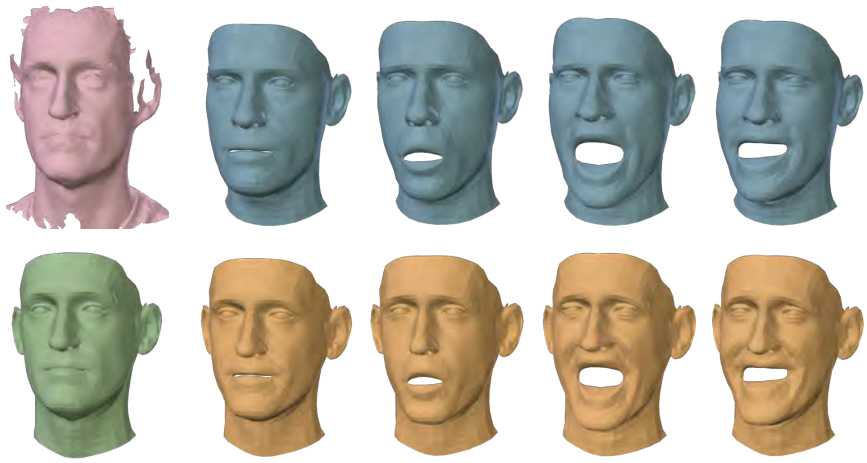

Fig. 20. Expression transfer from a source sequence (blue) to a static target scan (pink). The aligned personalized template for the scan is shown in green, the transferred expression in yellow. The supplementary document shows further examples.

H. Li, J. Yu, Y. Ye, and C. Bregler. 2013. Realtime facial animation with on-the-fly correctives. *Transactions on Graphics (Proceedings of SIGGRAPH)* 32, 4 (2013), 42:1–42:10.

J. Li, W. Xu, Z. Cheng, K. Xu, and R. Klein. 2015. Lightweight wrinkle synthesis for 3D facial modeling and animation. *Computer-Aided Design* 58 (2015), 117–122.

M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. 2015. SMPL: A Skinned Multi-person Linear Model. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 34, 6 (2015), 248:1–248:16.

M. M. Loper and M. J. Black. 2014. OpenDR: An Approximate Differentiable Renderer. In *European Conference on Computer Vision*. 154–169.

T. Neumann, K. Varanasi, S. Wenger, M. Wacker, M. Magnor, and C. Theobalt. 2013. Sparse Localized Deformation Components. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 32, 6 (2013), 179:1–179:10.

J. Nocedal and S. J. Wright. 2006. *Numerical Optimization*. Springer.

P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. 2009. A 3D Face Model for Pose and Illumination Invariant Face Recognition. In *International Conference on Advanced Video and Signal Based Surveillance*. 296–301.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

L. Pishchulin, S. Wuhrer, T. Helten, C. Theobalt, and B. Schiele. 2017. Building Statistical Shape Spaces for 3D Human Modeling. *Pattern Recognition* 67, C (July 2017), 276–286.

K. Robinette, S. Blackwell, H. Daanen, M. Boehmer, S. Fleming, T. Brill, D. Hoeferlin, and D. Burnsides. 2002. *Civilian American and European Surface Anthropometry Resource (CAESAR) Final Report*. Technical Report AFRL-HE-WP-TR-2002-0169. US Air Force Research Laboratory.

A. Salazar, S. Wuhrer, C. Shu, and F. Prieto. 2014. Fully automatic expression-invariant face correspondence. *Machine Vision and Applications* 25, 4 (2014), 859–879.

F. Shi, H.-T. Wu, X. Tong, and J. Chai. 2014. Automatic Acquisition of High-fidelity Facial Performances Using Monocular Videos. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 33, 6 (2014), 222:1–222:13.

R.W. Sumner and J. Popović. 2004. Deformation transfer for triangle meshes. *Transactions on Graphics (Proceedings of SIGGRAPH)* 23, 3 (2004), 399–405.

S. Suwajanakorn, I. Kemelmacher-Shlizerman, and S. M. Seitz. 2014. Total Moving Face Reconstruction. In *European Conference on Computer Vision*. 796–812.

S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman. 2015. What Makes Tom Hanks Look Like Tom Hanks. In *International Conference on Computer Vision*. 3952–3960.

J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt. 2015. Real-time Expression Transfer for Facial Reenactment. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 34, 6 (2015), 183:1–183:14.

D. Vlasic, M. Brand, H. Pfister, and J. Popović. 2005. Face transfer with multilinear models. *Transactions on Graphics (Proceedings of SIGGRAPH)* 24, 3 (2005), 426–433.

T. Weise, S. Bouaziz, H. Li, and M. Pauly. 2011. Realtime performance-based facial animation. *Transactions on Graphics (Proceedings of SIGGRAPH)* 30, 4 (2011), 77:1–77:10.

E. Wood, T. Baltrušaitis, L.-P. Morency, P. Robinson, and A. Bulling. 2016. A 3D morphable eye region model for gaze estimation. In *European Conference on Computer Vision*. 297–313.

C. Wu, D. Bradley, M. Gross, and T. Beeler. 2016. An Anatomically-constrained Local Deformation Model for Monocular Face Capture. *Transactions on Graphics (Proceedings of SIGGRAPH)* 35, 4 (2016), 115:1–115:12.

X. Xiong and F. De la Torre. 2013. Supervised descent method and its applications to face alignment. In *Conference on Computer Vision and Pattern Recognition*. 532–539.

F. Xu, J. Chai, Y. Liu, and X. Tong. 2014. Controllable High-fidelity Facial Performance Transfer. *Transactions on Graphics (Proceedings of SIGGRAPH)* 33, 4 (2014), 42:1–42:11.

F. Yang, J. Wang, E. Shechtman, L. Bourdev, and D. Metaxas. 2011. Expression flow for 3D-aware face component transfer. *Transactions on Graphics (Proceedings of SIGGRAPH)* 30, 4 (2011), 60:1–10.

L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato. 2006. A 3D Facial Expression Database for Facial Behavior Research. In *International Conference on Automatic Face and Gesture Recognition*. 211–216.