Personalized Face Modeling for Improved Face Reconstruction and Motion Retargeting

Bindita Chaudhuri[1]*, Noranart Vesdapunt[2],
Linda Shapiro[1], and Baoyuan Wang[2]

[1] University of Washington
{bindita,shapiro}@cs.washington.edu
[2] Microsoft Cloud and AI
{noves,baoyuanw}@microsoft.com

Abstract. Traditional methods for image-based 3D face reconstruction and facial motion retargeting fit a 3D morphable model (3DMM) to the face, which has limited modeling capacity and fail to generalize well to in-the-wild data. Use of deformation transfer or multilinear tensor as a personalized 3DMM for blendshape interpolation does not address the fact that facial expressions result in different local and global skin deformations in different persons. Moreover, existing methods learn a single albedo per user which is not enough to capture the expression-specific skin reflectance variations. We propose an end-to-end framework that jointly learns a personalized face model per user and per-frame facial motion parameters from a large corpus of in-the-wild videos of user expressions. Specifically, we learn user-specific expression blendshapes and dynamic (expression-specific) albedo maps by predicting personalized corrections on top of a 3DMM prior. We introduce novel training constraints to ensure that the corrected blendshapes retain their semantic meanings and the reconstructed geometry is disentangled from the albedo. Experimental results show that our personalization accurately captures fine-grained facial dynamics in a wide range of conditions and efficiently decouples the learned face model from facial motion, resulting in more accurate face reconstruction and facial motion retargeting compared to state-of-the-art methods.

Keywords: 3D face reconstruction, face modeling, face tracking, facial motion retargeting

1 Introduction

With the ubiquity of mobile phones, AR/VR headsets and video games, communication through facial gestures has become very popular, leading to extensive research in problems like 2D face alignment, 3D face reconstruction and facial motion retargeting. A major component of these problems is to estimate the 3D face, i.e., face geometry, appearance, expression, head pose and scene lighting,

* This work was done when the author visited Microsoft.

from 2D images or videos. 3D face reconstruction from monocular images is ill-posed by nature, so a typical solution is to leverage a parametric 3D morphable model (3DMM) trained on a limited number of 3D face scans as prior knowledge [2,35,51,28,38,14,47,11,24]. However, the low dimensional space limits their modeling capacity as shown in [45,50,21] and scalability using more 3D scans is expensive. Similarly, the texture model of a generic 3DMM is learned in a controlled environment and does not generalize well to in-the-wild images. Tran et al. [50,49] overcomes these limitations by learning a non-linear 3DMM from a large corpus of in-the-wild images. Nevertheless, these reconstruction-based approaches do not easily support facial motion retargeting.

In order to perform tracking for retargeting, blendshape interpolation technique is usually adopted where the users' blendshapes are obtained by deformation transfer [43], but this alone cannot reconstruct expressions realistically as shown in [14,26]. Another popular technique is to use a multilinear tensor-based 3DMM [51,5,4], where the expression is coupled with the identity implying that same identities should share the same expression blendshapes. However, we argue that facial expressions are characterized by different skin deformations on different persons due to difference in face shape, muscle movements, age and other factors. This kind of user-specific local skin deformations cannot be accurately represented by a linear combination of predefined blendshapes. For example, smiling and raising eyebrows create different cheek folds and forehead wrinkle patterns respectively on different persons, which cannot be represented by simple blendshape interpolation and require correcting the corresponding blendshapes. Some optimization-based approaches [26,14,20,36] have shown that modeling user-specific blendshapes indeed results in a significant improvement in the quality of face reconstruction and tracking. However, these approaches are computationally slow and require additional preprocessing (e.g. landmark detection) during test time, which significantly limits real-time applications with in-the-wild data on the edge devices. The work [8] trains a deep neural network instead to perform retargeting in real-time on typical mobile phones, but its use of predefined 3DMM limits its face modeling accuracy. Tewari et al. [44] leverage in-the-wild videos to learn face identity and appearance models from scratch, but they still use expression blendshapes generated by deformation transfer.

Moreover, existing methods learn a single albedo map for a user. The authors in [17] have shown that skin reflectance changes with skin deformations, but it is not feasible to generate a separate albedo map for every expression during retargeting. Hence it is necessary to learn the static reflectance separately, and associate the expression-specific dynamic reflectance with the blendshapes so that the final reflectance can be obtained by interpolation similar to blendshape interpolation, as in [33]. Learning dynamic albedo maps in addition to static albedo map also helps to capture the fine-grained facial expression details like folds and wrinkles [34], thereby resulting in reconstruction of higher fidelity.

To address these issues, we introduce a novel end-to-end framework that leverages a large corpus of in-the-wild user videos to jointly learn personalized face modeling and face tracking parameters. Specifically, we design a modeling

network which learns geometry and reflectance corrections on top of a 3DMM prior to generate user-specific expression blendshapes and dynamic (expression-specific) albedo maps. In order to ensure proper disentangling of the geometry from the albedo, we introduce the face parsing loss inspired by [57]. Note that [57] uses parsing loss in a fitting based framework whereas we use it in a learning based framework. We also ensure that the corrected blendshapes retain their semantic meanings by restricting the corrections to local regions using attention maps and by enforcing a blendshape gradient loss. We design a separate tracking network which predicts the expression blendshape coefficients, head pose and scene lighting parameters. The decoupling between the modeling and tracking networks enables our framework to perform reconstruction as well as retargeting (by tracking one user and transferring the facial motion to another user's model). Our main contributions are:

1. We propose a deep learning framework to learn user-specific expression blendshapes and dynamic albedo maps that accurately capture the complex user-specific expression dynamics and high-frequency details like folds and wrinkles, thereby resulting in photorealistic 3D face reconstruction.
2. We bring two novel constraints into the end-to-end training: face parsing loss to reduce the ambiguity between geometry and reflectance and blendshape gradient loss to retain the semantic meanings of the corrected blendshapes.
3. Our framework jointly learns user-specific face model and user-independent facial motion in disentangled form, thereby supporting motion retargeting.

2 Related Work

Face Modeling: Methods like [53,19,32,41,25,30,48] leverage user images captured with varying parameters (e.g. multiple viewpoints, expressions etc.) at least during training with the aim of user-specific 3D face reconstruction (not necessarily retargeting). Monocular video-based optimization techniques for 3D face reconstruction [13,14] leverage the multi-frame consistency to learn the facial details. For single image based reconstruction, traditional methods [59] regress the parameters of a 3DMM and then learn corrective displacement [19,22,18] or normal maps [40,37] to capture the missing details. Recently, several deep learning based approaches have attempted to overcome the limited representation power of 3DMM. Tran et al. [50,49] proposed to train a deep neural network as a non-linear 3DMM. Tewari et al. [45] proposed to learn shape and reflectance correctives on top of the linear 3DMM. In [44], Tewari et al. learn new identity and appearance models from videos. However, these methods use expression blendshapes obtained by deformation transfer [43] from a generic 3DMM to their own face model and do not optimize the blendshapes based on the user's identity. In addition, these methods predict a single static albedo map to represent the face texture, which fail to capture adequate facial details.

Personalization: Optimization based methods like [26,20,7] have demonstrated the need to optimize the expression blendshapes based on user-specific facial dy-

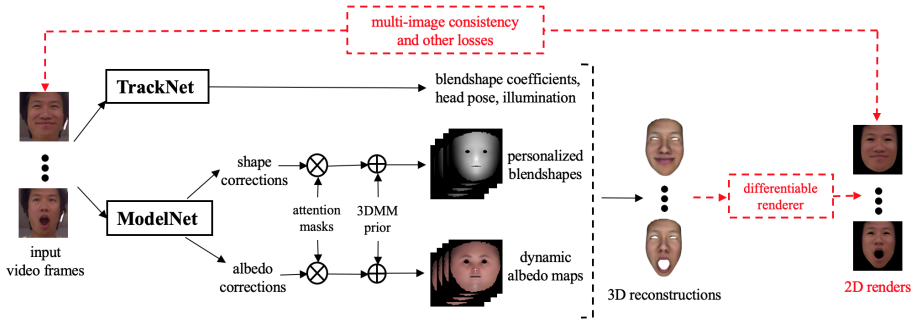

Fig. 1: Our end-to-end framework. Our framework takes frames from in-the-wild video(s) of a user as input and generates per-frame tracking parameters via the *TrackNet* and personalized face model via the *ModelNet*. The networks are trained together in an end-to-end manner (marked in red) by projecting the reconstructed 3D outputs into 2D using a differentiable renderer and computing multi-image consistency losses and other regularization losses.

namics. These methods alternately update the blendshapes and the corresponding coefficients to accurately fit some example poses in the form of 3D scans or 2D images. For facial appearance, existing methods either use a generic texture model with linear or learned bases or use a GAN [15] to generate a static texture map. But different expressions result in different texture variations, and Nagano et al. [33] and Olszewski et al. [34] addressed this issue by using a GAN to predict the expression-specific texture maps given the texture map in neutral pose. However, the texture variations with expression also vary from person to person. Hence, hallucinating an expression-specific texture map for a person by learning the expression dynamics of other persons is not ideal. Besides, these methods requires fitted geometry as a preprocessing step, thereby limiting the accuracy of the method by the accuracy of the geometry fitting mechanism.

Face Tracking and Retargeting: Traditional face tracking and retargeting methods [52,3,27] generally optimize the face model parameters with occasional correction of the expression blendshapes using depth scans. Recent deep learning based tracking frameworks like [47,53,8,23] either use a generic face model and fix the model during tracking, or alternate between tracking and modeling until convergence. We propose to perform joint face modeling and tracking with novel constraints to disambiguate the tracking parameters from the model.

3 Methodology

3.1 Overview

Our network architecture, as shown in Fig. 1, has two parts: 1) *ModelNet* which learns to capture the user-specific facial details and 2) *TrackNet* which learns to capture the user-independent facial motion. The networks are trained together in an end-to-end manner using multi-frame images of different identities, i.e., multiple images $\{I_1, \ldots, I_N\}$ of the same person sampled from a video in

each mini-batch. We leverage the fact that the person's facial geometry and appearance remain unchanged across all the frames in a video, whereas the facial expression, head pose and scene illumination change on a per-frame basis. The *ModelNet* extracts a common feature from all the N images to learn a user-specific face shape, expression blendshapes and dynamic albedo maps (Section 3.2). The *TrackNet* processes each of the N images individually to learn the image-specific expression blendshape coefficients, pose and illumination parameters (Section 3.3). The predictions of *ModelNet* and *TrackNet* are combined to reconstruct the 3D faces and then projected to the 2D space using a differentiable renderer in order to train the network in a self-supervised manner using multi-image photometric consistency, landmark alignment and other constraints. During testing, the default settings can perform 3D face reconstruction. However, our network architecture and training strategy allow simultaneous tracking of one person's face using *TrackNet* and modeling another person's face using *ModelNet*, and then retarget the tracked person's facial motion to the modeled person or an external face model having similar topology as our face model.

3.2 Learning Personalized Face Model

Our template 3D face consists of a mean (neutral) face mesh S_0 having 12K vertices, per-vertex colors (converted to 2D mean albelo map R_0 using UV coordinates) and 56 expression blendshapes $\{S_1, \ldots, S_{56}\}$. Given a set of expression coefficients $\{w_1, \ldots, w_{56}\}$, the template face shape can be written as $\bar{S} = w_0 S_0 + \sum_{i=1}^{56} w_i S_i$ where $w_0 = (1 - \sum_{i=1}^{56} w_i)$. Firstly, we propose to learn an identity-specific corrective deformation $\Delta_0^{\bar{S}}$ from the identity of the input images to convert $\bar{S}$ to identity-specific shape. Then, in order to better fit the facial expression of the input images, we learn corrective deformations Δ_i^S for each of the template blendshapes S_i to get identity-specific blendshapes. Similarly, we learn a corrective albedo map Δ_0^R to convert R_0 to identity-specific static albedo map. In addition, we also learn corrective albedo maps Δ_i^R corresponding to each S_i to get identity-specific dynamic (expression-specific) albedo maps.

In our *ModelNet*, we use a shared convolutional encoder E^{model} to extract features F_n^{model} from each image $I_n \in \{I_1, \ldots, I_N\}$ in a mini-batch. Since all the N images belong to the same person, we take an average over all the F_n^{model} features to get a common feature F^{model} for that person. Then, we pass F^{model} through two separate convolutional decoders, D_S^{model} to estimate the shape corrections Δ_0^S and Δ_i^S, and D_R^{model} to estimate the albedo corrections Δ_0^R and Δ_i^R. We learn the corrections in the UV space instead of the vertex space to reduce the number of network parameters and preserve the contextual information.

User-specific expression blendshapes A naive approach to learn corrections on top of template blendshapes based on the user's identity would be to predict corrective values for all the vertices and add them to the template blendshapes. However, since blendshape deformation is local, we want to restrict the corrected deformation to a similar local region as the template deformation. To do this, we first apply an attention mask over the per-vertex corrections and then add it to the template blendshape. We compute the attention mask A_i corresponding

to the blendshape S_i by calculating the per-vertex euclidean distances between S_i and S_0, thresholding them at 0.001, normalizing them by the maximum distance, and then converting them into the UV space. We also smooth the mask discontinuities using a small amount of Gaussian blur following [33]. Finally, we multiply A_i with Δ_i^S and add it to S_i to obtain a corrected S_i. Note that the masks are precomputed and then fixed during network operations. The final face shape is thus given by:

$$S = w_0 S_0 + \mathcal{F}(\Delta_0^S) + \sum_{i=1}^{56} w_i [S_i + \mathcal{F}(A_i \Delta_i^S)] \tag{1}$$

where $\mathcal{F}(\cdot)$ is a sampling function for UV space to vertex space conversion.

User-specific dynamic albedo maps We use one static albedo map to represent the identity-specific neutral face appearance and 56 dynamic albedo maps, one for each expression blendshape, to represent the expression-specific face appearance. Similar to blendshape corrections, we predict 56 albedo correction maps in the UV space and add them to the static albedo map after multiplying the dynamic correction maps with the corresponding UV attention masks. Our final face albedo is thus given by:

$$R = R_0^t + \Delta_0^R + \sum_{i=1}^{56} w_i [A_i \Delta_i^R] \tag{2}$$

where R_0^t is the trainable mean albedo initialized with the mean albedo R_0 from our template face similar to [44].

3.3 Joint Modeling and Tracking

The *TrackNet* consists of a convolutional encoder E^{track} followed by multiple fully connected layers to regress the tracking parameters $\mathbf{p}_n = (\mathbf{w}_n, \mathbf{R}_n, \mathbf{t}_n, \gamma_n)$ for each image I_n. The encoder and fully connected layers are shared over all the N images in a mini-batch. Here $\mathbf{w}_n = (w_0^n, \ldots, w_{56}^n)$ is the expression coefficient vector and $\mathbf{R}_n \in SO(3)$ and $\mathbf{t}_n \in \mathbb{R}^3$ are the head rotation (in terms of Euler angles) and 3D translation respectively. $\gamma_n \in \mathbb{R}^{27}$ are the 27 Spherical Harmonics coefficients (9 per color channel) following the illumination model of [44].

Training Phase: We first obtain a face shape S_n and albedo R_n for each I_n by combining S (equation 1) and R (equation 2) from the *ModelNet* and the expression coefficient vector $\mathbf{w}_n$ from the *TrackNet*. Then, similar to [15,44], we transform the shape using head pose as $\tilde{S}_n = \mathbf{R}_n S_n + \mathbf{t}_n$ and project it onto the 2D camera space using a perspective camera model $\Phi : \mathbb{R}^3 \rightarrow \mathbb{R}^2$. Finally, we use a differentiable renderer $\mathcal{R}$ to obtain the reconstructed 2D image as $\hat{I}_n = \mathcal{R}(\tilde{S}_n, \mathbf{n}_n, R_n, \gamma_n)$ where $\mathbf{n}_n$ are the per-vertex normals. We also mark 68 facial landmarks on our template mesh which we can project onto the 2D space using Φ to compare with the ground truth 2D landmarks.

Testing Phase: The *ModelNet* can take a variable number of input images of a person (due to our feature averaging technique) to predict a personalized face

model. The *TrackNet* executes independently on one or more images of the same person given as input to *ModelNet* or a different person. For face reconstruction, we feed images of the same person to both the networks and combine their outputs as in the training phase to get the 3D faces. In order to perform facial motion retargeting, we first obtain the personalized face model of the target subject using *ModelNet*. We then predict the facial motion of the source subject on a per-frame basis using the *TrackNet* and combine it with the target face model. It is important to note that the target face model can be any external face model with semantically similar expression blendshapes.

3.4 Loss Functions

We train both the *TrackNet* and the *ModelNet* together in an end-to-end manner using the following loss function:

$$L = \lambda_{\text{ph}} L_{\text{ph}} + \lambda_{\text{lm}} L_{\text{lm}} + \lambda_{\text{pa}} L_{\text{pa}} + \lambda_{\text{sd}} L_{\text{sd}} + \lambda_{\text{bg}} L_{\text{bg}} + \lambda_{\text{reg}} L_{\text{reg}} \tag{3}$$

where the loss weights λ_* are chosen empirically and their values are given in the supplementary material[3].

Photometric and Landmark Losses: We use the $l_{2,1}$ [49] loss to compute the multi-image photometric consistency loss between the input images I_n and the reconstructed images $\hat{I}_n$. The loss is given by

$$L_{\text{ph}} = \sum_{n=1}^{N} \frac{\sum_{q=1}^{Q} ||M_n(q) * [I_n(q) - \hat{I}_n(q)]||_2}{\sum_{q=1}^{Q} M_n(q)} \tag{4}$$

where M_n is the mask generated by the differentiable renderer (to exclude the background, eyeballs and mouth interior) and q ranges over all the pixels Q in the image. In order to further improve the quality of the predicted albedo by preserving high-frequency details, we add the image (spatial) gradient loss having the same expression as the photometric loss with the images replaced by their gradients. Adding other losses as in [15] resulted in no significant improvement. The landmark alignment loss L_{lm} is computed as the l_2 loss between the ground truth and predicted 68 2D facial landmarks.

Face Parsing Loss: The photometric and landmark loss constraints are not strong enough to overcome the ambiguity between shape and albedo in the 2D projection of a 3D face. Besides, the landmarks are sparse and often unreliable especially for extreme poses and expressions which are difficult to model because of depth ambiguity. So, we introduce the face parsing loss given by:

$$L_{\text{pa}} = \sum_{n=1}^{N} ||I_n^{\text{pa}} - \hat{I}_n^{\text{pa}}||_2 \tag{5}$$

where I_n^{pa} is the ground truth parsing map generated using the method in [29] and $\hat{I}_n^{\text{pa}}$ is the predicted parsing map generated as $\mathcal{R}(\tilde{S}_n, \mathbf{n}_n, T)$ with a fixed precomputed UV parsing map T.

[3] https://homes.cs.washington.edu/~bindita/personalizedfacemodeling.html

Shape Deformation Smoothness Loss: We employ Laplacian smoothness on the identity-specific corrective deformation to ensure that our predicted shape is locally smooth. The loss is given as:

$$L_{\text{sd}} = \sum_{v=1}^{V} \sum_{u \in \mathcal{N}_v} ||\Delta_0^S(v) - \Delta_0^S(u)||_2^2 \tag{6}$$

where V is the total number of vertices in our mesh and $\mathcal{N}_v$ is the set of neighboring vertices directly connected to vertex v.

Blendshape Gradient Loss: Adding free-form deformation to a blendshape, even after restricting it to a local region using attention masks, can change the semantic meaning of the blendshape. However, in order to retarget tracked facial motion of one person to the blendshapes of another person, the blendshapes of both the persons should have semantic correspondence. We introduce a novel blendshape gradient loss to ensure that the deformation gradients of the corrected blendshapes are similar to those of the template blendshapes. The loss is given by:

$$L_{\text{bg}} = \sum_{i=1}^{56} ||\mathbf{G}_{S_0 \rightarrow (S_i + \Delta_i^S)} - \mathbf{G}_{S_0 \rightarrow S_i}||_2^2 \tag{7}$$

where $\mathbf{G}_{a \rightarrow b}$ denotes the gradient of the deformed mesh b with respect to original mesh a. Details about how to compute $\mathbf{G}$ can be found in [26].

Tracking Parameter Regularization Loss: We use sigmoid activation at the output of the expression coefficients and regularize the coefficients using l_1 loss (L_{reg}^w) to ensure sparse coefficients in the range $[0, 1]$. In order to disentangle the albedo from the lighting, we use a lighting regularization loss given by:

$$L_{\text{reg}}^{\gamma} = ||\gamma - \gamma_{\text{mean}}||_2 + \lambda_{\gamma}||\gamma||_2 \tag{8}$$

where the first term ensures that the predicted light is mostly monochromatic and the second term restricts the overall lighting. We found that regularizing the illumination automatically resulted in albedo consistency, so we don't use any additional albedo loss. Finally, $L_{\text{reg}} = L_{\text{reg}}^w + L_{\text{reg}}^{\gamma}$.

4 Experimental Setup

Datasets: We train our network using two datasets: 1) VoxCeleb2 [9] and 2) ExpressiveFaces. We set aside 10% of each dataset for testing. VoxCeleb2 has more than 140k videos of about 6000 identities collected from internet, but the videos are mostly similar. So, we choose a subset of 90k videos from about 4000 identities. The images in VoxCeleb2 vary widely in pose but lack variety in expressions and illumination, so we add a custom dataset (ExpressiveFaces) to our training, which contains 3600 videos of 3600 identities. The videos are captured by the users using a hand-held camera (typically the front camera of a mobile phone) as they perform a wide variety of expressions and poses in

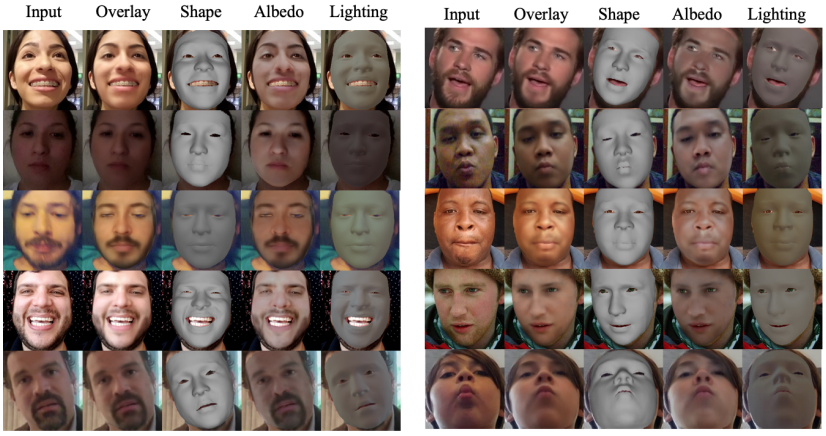

Fig. 2: Qualitative results of our method. Our modeling network accurately captures high-fidelity facial details specific to the user, thereby enabling the tracking network to learn user-independent facial motion. Our network can handle a wide variety of pose, expression, lighting conditions, facial hair and makeup etc. Refer to supplementary material for more results for images and videos.

both indoor and outdoor environments. We sample the videos at 10fps to avoid multiple duplicate frames, randomly delete frames with neutral expression and pose based on a threshold on the expression and pose parameters predicted by [8], and then crop the face and extract ground truth 2D landmarks using [8]. The cropped faces are resized to 224×224 and grouped into mini-batches, each of N images chosen randomly from different parts of a video to ensure sufficient diversity in the training data. We set $N = 4$ during training and $N = 1$ during testing (unless otherwise mentioned) as evaluated to work best for real-time performance in [44].

Implementation Details: We implemented our networks in Tensorflow and used TF mesh renderer [16] for differentiable rendering. During the first stage of training, we train both *TrackNet* and *ModelNet* in an end-to-end manner using equation 3. During the second stage of training, we fix the weights of *TrackNet* and fine-tune the *ModelNet* to better learn the expression-specific corrections. The fine-tuning is done using the same loss function as before except the tracking parameter regularization loss since the tracking parameters are now fixed. This training strategy enables us to tackle the bilinear optimization problem of optimizing the blendshapes and the corresponding coefficients, which is generally solved through alternate minimization by existing optimization-based methods. For training, we use a batch size of 8, learning rates of 10^{-4} (10^{-5} during second stage) and Adam optimizer for loss minimization. Training takes $\sim$20 hours on a single Nvidia Titan X for the first stage, and another $\sim$5 hours for the second stage. The encoder and decoder together in *ModelNet* has an architecture similar to U-Net [39] and the encoder in *TrackNet* has the same architecture as ResNet-18 (details in the supplementary). Since our template mesh contains 12264 vertices, we use a corresponding UV map of dimensions 128×128.

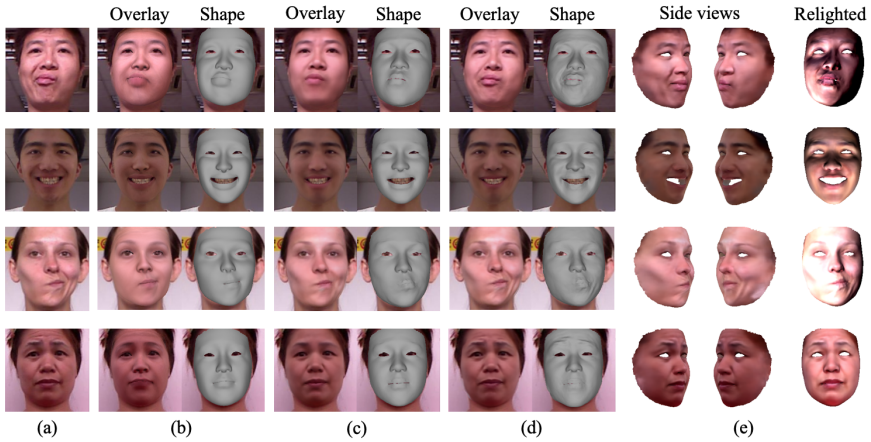

Fig. 3: Importance of personalization. (a) input image, (b) reconstruction using 3DMM prior only, (c) reconstruction after adding only identity-based corrections, i.e. Δ_0^S and Δ_0^R in eq. (2) and (3) respectively, (d) reconstruction after adding expression-specific corrections, (e) results of (d) with different viewpoints and illumination.

5 Results

We evaluate the effectiveness of our framework using both qualitative results and quantitative comparisons. Fig. 2 shows the personalized face shape and albedo, scene illumination and the final reconstructed 3D face generated from monocular images by our method. Learning a common face shape and albedo from multiple images of a person separately from the image-specific facial motion helps in successfully decoupling the tracking parameters from the learned face model. As a result, our tracking network have the capacity to represent a wide range of expressions, head pose and lighting conditions. Moreover, learning a unified model from multiple images help to overcome issues like partial occlusion, self-occlusion, blur in one or more images. Fig. 3 shows a gallery of examples that demonstrate the effectiveness of personalized face modeling for better reconstruction. Fig. 7a shows that our network can be efficiently used to perform facial motion retargeting to another user or to an external 3D model of a stylized character in addition to face reconstruction.

5.1 Importance of Personalized Face Model

Importance of personalized blendshapes: Modeling the user-specific local geometry deformations while performing expressions enable our modeling network to accurately fit the facial shape of the input image. Fig. 4a shows examples of how the same expression can look different on different identities and how the corrected blendshapes capture those differences for more accurate reconstruction than with the template blendshapes. In the first example, the extent of opening of the mouth in the *mouth open* blendshape is adjusted according to the user expression. In the second example, the mouth shape of the *mouth funnel* blendshape is corrected.

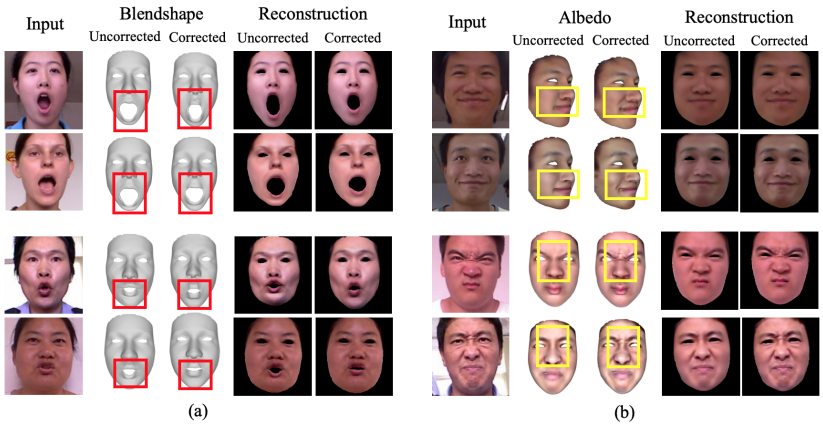

Fig. 4: Visualization of corrected blendshapes and albedo. The corrections are highlighted. (a) Learning user-specific blendshapes corrects the mouth shape of the blendshapes, (b) Learning user-specific dynamic albedo maps captures the high-frequency details like skin folds and wrinkles.

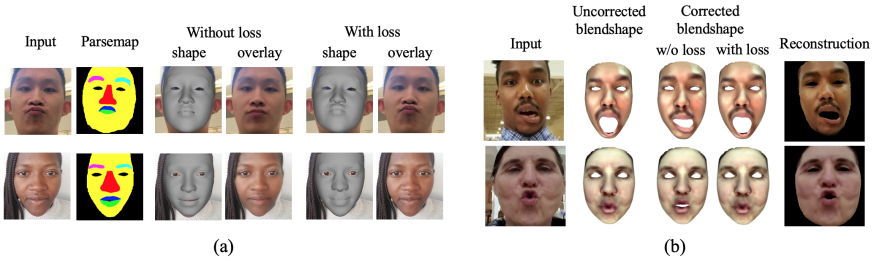

Fig. 5: Importance of novel training constraints. (a) importance of face parsing loss in obtaining accurate geometry decoupled from albedo, (b) importance of blendshape gradient loss in retaining the semantic meaning of *mouth open* (row 1) and *kiss* (row 2) blendshapes after correction.

Importance of dynamic textures: Modeling the user-specific local variations in skin reflectance while performing expressions enable our modeling network to generate a photorealistic texture for the input image. Fig. 4b shows examples of how personalized dynamic albedo maps help in capturing the high-frequency expression-specific details compared to static albedo maps. In the first example, our method accurately captures the folds around the nose and mouth during smile expression. In the second example, the unique wrinkle patterns between the eyebrows of the two users are correctly modeled during a disgust expression.

5.2 Importance of Novel Training Constraints

Importance of parsing loss: The face parse map ensures that each face part of the reconstructed geometry is accurate as shown in [57]. This prevents the albedo to compensate for incorrect geometry, thereby disentangling the albedo from the geometry. However, the authors of [57] use parse map in a geometry

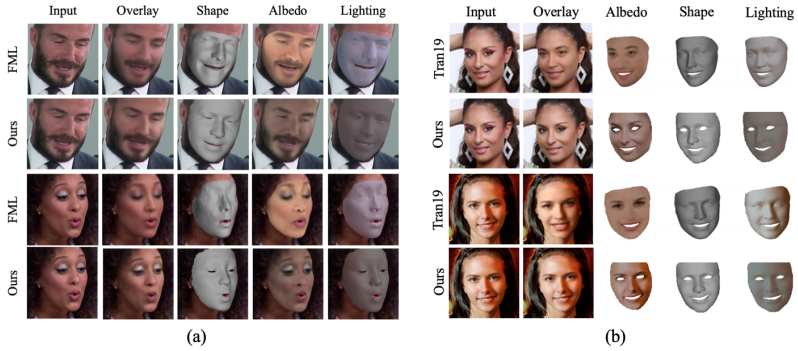

Fig. 6: Visual comparison with (a) FML [44], (b) Non-linear 3DMM [49].

fitting framework which, unlike our learning framework, does not generalize well to in-the-wild images. Besides, due to the dense correspondence of parse map compared to the sparse 2D landmarks, parsing loss (a) provides a stronger supervision on the geometry, and (b) is more robust to outliers. We demonstrate the effectiveness of face parsing loss in Fig. 5a. In the first example, the kiss expression is correctly reconstructed with the loss, since the 2D landmarks are not enough to overcome the depth ambiguity. In the second example, without parsing loss the albedo tries to compensate for the incorrect geometry by including the background in the texture. With loss, the nose shape, face contour and the lips are corrected in the geometry, resulting in better reconstruction.

Importance of blendshape gradient loss: Even after applying attention masks to restrict the blendshape corrections to local regions, our method can distort a blendshape such that it loses its semantic meaning, which is undesirable for retargeting purposes. We prevent this by enforcing the blendshape gradient loss, as shown in Fig. 5b. In the first example, without gradient loss, the *mouth open* blendshape gets combined with *jaw left* blendshape after correction in order to minimize the reconstruction loss. With gradient loss, the reconstruction is same but the *mouth open* blendshape retains its semantics after correction. Similarly in the second example, without gradient loss, the *kiss* blendshape gets combined with the *mouth funnel* blendshape, which is prevented by the loss.

5.3 Visual Comparison with State-of-the-art Methods

3D face reconstruction: We test the effectiveness of our method on VoxCeleb2 test set to compare with FML results [44] as shown in Fig. 6a. In the first example, our method captures the mouth shape and face texture better. The second example shows that our personalized face modeling can efficiently model complex expressions like kissing and complex texture like eye shadow better than FML. We also show the visual comparisons between our method and Non-linear 3DMM [49] on the AFLW2000-3D dataset [58] in Fig. 6b. Similar to FML, Non-linear 3DMM fails to accurately capture the subtle facial details.

Face tracking and retargeting: By increasing the face modeling capacity and decoupling the model from the facial motion, our method performs superior face

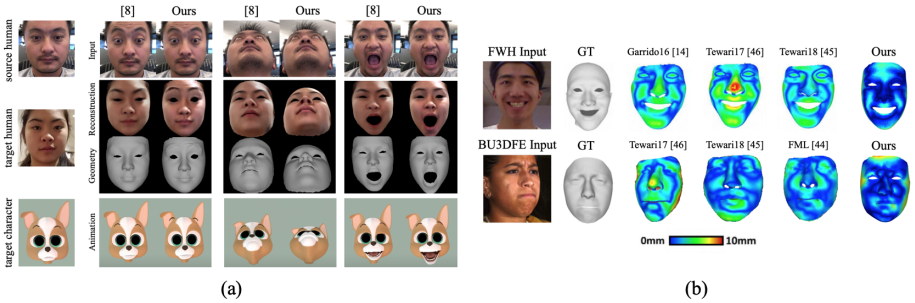

Fig. 7: (a) Tracking comparison with [8]. (b) 3D reconstruction error maps.

tracking and retargeting compared to a recent deep learning based retargeting approach [8]. Fig. 7a shows some frames from a user video and how personalization helps in capturing the intensity of the expressions more accurately.

5.4 Quantitative Evaluation

3D face reconstruction: We compute 3D geometry reconstruction error (root mean squared error between a predicted vertex and ground truth correspondence point) to evaluate our predicted mesh on 3D scans from BU-3DFE [55] and Facewarehouse (FWH) [6]. For BU-3DFE we use both the views per scan as input, and for FWH we do not use any special template to start with, unlike Asian face template used by FML. Our personalized face modeling and novel constraints together result in lower reconstruction error compared to state-of-the-art methods (Table 2a and Fig. 7b). The optimization-based method [14] obtains 1.59mm 3D error for FWH compared to our 1.68mm, but is much slower (120s/frames) compared to our method (15.4ms/frame). We also show how each component of our method helps in improving the overall output in Table 1. For photometric error, we used 1000 images of CelebA [31] (referred as CelebA*) dataset to be consistent with [44].

Face tracking: We evaluate the tracking performance of our method using two metrics: 1) Normalized Mean Error (NME), defined as an average Euclidean distance between the 68 predicted and ground truth 2D landmarks normalized by the bounding box dimension, on AFLW2000-3D dataset [58], and 2) Area under the Curve (AUC) of the cumulative error distribution curve for 2D landmark error [10] on 300VW video test set [42]. Table 2b shows that we achieve lower landmark error compared to state-of-the-art methods although our landmarks are generated by a third-party method. We also outperform existing methods on video data (Table 2c). For video tracking, we detect the face in the first frame and use the bounding box from previous frame for subsequent frames similar to [8]. However, the reconstruction is performed on a per-frame basis to avoid inconsistency due to the choice of random frames.

Facial motion retargeting: In order to evaluate whether our tracked facial expression gets correctly retargeted on the target model, we use the expression metric defined as the mean absolute error between the predicted and ground

Table 1: Ablation study. Evaluation of different components of our proposed method in terms of standard evaluation metrics. Note that B and C are obtained with all the loss functions other than the parsing loss and the gradient loss.

Method	**3D error (mm)**				**NME**	**AUC**	**Photo error**
	BU-3DFE		**FWH**		**AFLW2000-3D**	**300VW**	**CelebA***
	Mean	SD	Mean	SD	Mean	Mean	Mean
3DMM prior (A)	2.21	0.52	2.13	0.49	3.94	0.845	22.76
A + blendshape corrections (B)	2.04	0.41	1.98	0.44	3.73	0.863	22.25
B + albedo corrections (C)	1.88	0.39	1.85	0.41	3.68	0.871	20.13
C + parsing loss (D)	1.67	0.35	1.73	0.37	3.53	0.883	19.49
D + gradient loss (final)	1.61	0.32	1.68	0.35	3.49	0.890	18.91

Table 2: Quantitative evaluation with state-of-the-art methods. (a) 3D reconstruction error (mm) on BU-3DFE and FWH datasets, (b) NME (%) on AFLW2000-3D (divided into 3 groups based on yaw angles), (c) AUC for cumulative error distribution of the 2D landmark error for 300VW test set (divided into 3 scenarios by the authors). Note that higher AUC is better, whereas lower value is better for the other two metrics.

(a)

Method	**BU-3DFE**		**FWH**	
	Mean	SD	Mean	SD
[45]	1.83	0.39	1.84	0.38
[46]	3.22	0.77	2.19	0.54
[44]	1.74	0.43	1.90	0.40
Ours	**1.61**	**0.32**	**1.68**	**0.35**

(b)

Method	[0-30°]	[30-60°]	[60-90°]	Mean
[58]	3.43	4.24	7.17	4.94
[1]	3.15	4.33	5.98	4.49
[12]	2.75	3.51	4.61	3.62
[8]	2.91	3.83	4.94	3.89
Ours	**2.56**	**3.39**	**4.51**	**3.49**

(c)

Method	Sc. 1	Sc. 2	Sc. 3
[54]	0.791	0.788	0.710
[56]	0.748	0.760	0.726
[10]	0.847	0.838	0.769
[8]	0.901	0.884	0.842
Ours	**0.913**	**0.897**	**0.861**

Table 3: Quantitative evaluation of retargeting accuracy (measured by the expression metric) on [8] expression test set. Lower error means the model performs better for extreme expressions.

Model	Eye Close	Eye Wide	Brow Raise	Brow Anger	Mouth Open	Jaw L/R	Lip Roll	Smile	Kiss	Avg
(1) Retargeting [8]	0.117	0.407	0.284	0.405	0.284	0.173	0.325	0.248	0.349	0.288
(2) Ours	0.140	0.389	0.259	0.284	0.208	0.394	0.318	0.121	0.303	**0.268**

truth blendshape coefficients as in [8]. Our evaluation results in Table 3 emphasize the importance of personalized face model in improved retargeting, since [8] uses a generic 3DMM as its face model.

6 Conclusion

We propose a novel deep learning based approach that learns a user-specific face model (expression blendshapes and dynamic albedo maps) and user-independent facial motion disentangled from each other by leveraging in-the-wild videos. Extensive evaluation have demonstrated that our personalized face modeling combined with our novel constraints effectively performs high-fidelity 3D face reconstruction, facial motion tracking, and retargeting of the tracked facial motion from one identity to another.

Acknowledgements: We thank the anonymous reviewers for their constructive feedback, Muscle Wu, Wenbin Zhu and Zeyu Chen for helping, and Alex Colburn for valuable discussions.

References

1. Bhagavatula, C., Zhu, C., Luu, K., Savvides, M.: Faster than real-time facial alignment: A 3D spatial transformer network approach in unconstrained poses. In: IEEE International Conference on Computer Vision (ICCV) (2017)
2. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D faces. In: Proceedings SIGGRAPH. pp. 187–194 (1999)
3. Bouaziz, S., Wang, Y., Pauly, M.: Online modeling for realtime facial animation. ACM Transactions on Graphics **32**(4) (Jul 2013)
4. Cao, C., Chai, M., Woodford, O., Luo, L.: Stabilized real-time face tracking via a learned dynamic rigidity prior. ACM Transactions on Graphics **37**(6) (Dec 2018)
5. Cao, C., Hou, Q., Zhou, K.: Displaced dynamic expression regression for real-time facial tracking and animation. ACM Transactions on Graphics **33**(4) (Jul 2014)
6. Cao, C., Weng, Y., Lin, S., Zhou, K.: 3D shape regression for real-time facial animation. ACM Transactions on Graphics **32**(4) (Jul 2013)
7. Cao, C., Wu, H., Weng, Y., Shao, T., Zhou, K.: Real-time facial animation with image-based dynamic avatars. ACM Transactions on Graphics **35**(4), 126:1–126:12 (Jul 2016)
8. Chaudhuri, B., Vesdapunt, N., Wang, B.: Joint face detection and facial motion retargeting for multiple faces. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
9. Chung, J.S., Nagrani, A., Zisserman, A.: Voxceleb2: Deep speaker recognition. In: INTERSPEECH (2018)
10. Deng, J., Trigeorgis, G., Zhou, Y., Zafeiriou, S.: Joint multi-view face alignment in the wild. arXiv preprint arXiv:1708.06023 (2017)
11. Deng, Y., Yang, J., Xu, S., Chen, D., Jia, Y., Tong, X.: Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In: IEEE Conference on Computer Vision and Pattern Recognition Workshop on Analysis and Modeling of Faces and Gestures (CVPRW) (2019)
12. Feng, Y., Wu, F., Shao, X., Wang, Y., Zhou, X.: Joint 3D face reconstruction and dense alignment with position map regression network. In: European Conference on Computer Vision (ECCV) (2018)
13. Garrido, P., Valgaerts, L., Wu, C., Theobalt, C.: Reconstructing detailed dynamic face geometry from monocular video. ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2013) **32**(6) (Nov 2013)
14. Garrido, P., Zollhöfer, M., Casas, D., Valgaerts, L., Varanasi, K., Pérez, P., Theobalt, C.: Reconstruction of personalized 3D face rigs from monocular video. ACM Transactions on Graphics **35**(3) (May 2016)
15. Gecer, B., Ploumpis, S., Kotsia, I., Zafeiriou, S.: GANFIT: generative adversarial network fitting for high fidelity 3d face reconstruction. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
16. Genova, K., Cole, F., Maschinot, A., Sarna, A., Vlasic, D., Freeman, W.T.: Unsupervised training for 3d morphable model regression. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
17. Gotardo, P., Riviere, J., Bradley, D., Ghosh, A., Beeler, T.: Practical dynamic facial appearance modeling and acquisition. ACM Transactions on Graphics **37**(6) (Dec 2018)
18. Guo, Y., Zhang, J., Cai, J., Jiang, B., Zheng, J.: Cnn-based real-time dense face reconstruction with inverse-rendered photo-realistic face images. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2018)

19. Huynh, L., Chen, W., Saito, S., Xing, J., Nagano, K., Jones, A., Debevec, P., Li, H.: Mesoscopic Facial Geometry Inference Using Deep Neural Networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
20. Ichim, A.E., Bouaziz, S., Pauly, M.: Dynamic 3D avatar creation from hand-held video input. ACM Transactions on Graphics **34**(4) (Jul 2015)
21. Jackson, A.S., Bulat, A., Argyriou, V., Tzimiropoulos, G.: Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. In: IEEE International Conference on Computer Vision (ICCV) (2017)
22. Jiang, L., Zhang, J., Deng, B., Li, H., Liu, L.: 3d face reconstruction with geometry details from a single image. IEEE Transactions on Image Processing **27**(10), 4756–4770 (Oct 2018)
23. Kim, H., Garrido, P., Tewari, A., Xu, W., Thies, J., Niessner, M., Pérez, P., Richardt, C., Zollhöfer, M., Theobalt, C.: Deep video portraits. ACM Transactions on Graphics **37**(4), 163:1–163:14 (Jul 2018)
24. Kim, H., Zollöfer, M., Tewari, A., Thies, J., Richardt, C., Theobalt, C.: Inversefacenet: Deep single-shot inverse face rendering from a single image. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
25. Laine, S., Karras, T., Aila, T., Herva, A., Saito, S., Yu, R., Li, H., Lehtinen, J.: Production-level facial performance capture using deep convolutional neural networks. In: Eurographics Symposium on Computer Animation (2017)
26. Li, H., Weise, T., Pauly, M.: Example-based facial rigging. ACM Transactions on Graphics (Proceedings SIGGRAPH) **29**(3) (Jul 2010)
27. Li, H., Yu, J., Ye, Y., Bregler, C.: Realtime facial animation with on-the-fly correctives. ACM Transactions on Graphics **32**(4) (Jul 2013)
28. Li, T., Bolkart, T., Black, M.J., Li, H., Romero, J.: Learning a model of facial shape and expression from 4D scans. ACM Transactions on Graphics **36**(6) (Nov 2017)
29. Lin, J., Yang, H., Chen, D., Zeng, M., Wen, F., Yuan, L.: Face parsing with roi tanh-warping. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
30. Liu, F., Zhu, R., Zeng, D., Zhao, Q., Liu, X.: Disentangling features in 3d face shapes for joint face reconstruction and recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
31. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: International Conference on Computer Vision (ICCV) (2015)
32. Lombardi, S., Saragih, J., Simon, T., Sheikh, Y.: Deep appearance models for face rendering. ACM Transactions on Graphics **37**(4) (Jul 2018)
33. Nagano, K., Seo, J., Xing, J., Wei, L., Li, Z., Saito, S., Agarwal, A., Fursund, J., Li, H.: paGAN: Real-time avatars using dynamic textures. ACM Transactions on Graphics **37**(6) (Dec 2018)
34. Olszewski, K., Li, Z., Yang, C., Zhou, Y., Yu, R., Huang, Z., Xiang, S., Saito, S., Kohli, P., Li, H.: Realistic dynamic facial textures from a single image using gans. In: The IEEE International Conference on Computer Vision (ICCV) (2017)
35. Paysan, P., Knothe, R., Amberg, B., Romdhani, S., Vetter, T.: A 3d face model for pose and illumination invariant face recognition. In: IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) (2009)
36. Ribera, R.B.i., Zell, E., Lewis, J.P., Noh, J., Botsch, M.: Facial retargeting with automatic range of motion alignment. ACM Transactions on Graphics **36**(4) (2017)
37. Richardson, E., Sela, M., Or-El, R., Kimmel, R.: Learning detailed face reconstruction from a single image. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

38. Romdhani, S., Vetter, T.: Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2005)
39. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI). pp. 234–241 (2015)
40. Roth, J., Tong, Y., Liu, X.: Adaptive 3D face reconstruction from unconstrained photo collections. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
41. Sanyal, S., Bolkart, T., Feng, H., Black, M.: Learning to regress 3d face shape and expression from an image without 3d supervision. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
42. Shen, J., Zafeiriou, S., Chrysos, G.G., Kossaifi, J., Tzimiropoulos, G., Pantic, M.: The first facial landmark tracking in-the-wild challenge: Benchmark and results. In: IEEE International Conference on Computer Vision Workshops (ICCVW) (2015)
43. Sumner, R.W., Popović, J.: Deformation transfer for triangle meshes. In: ACM SIGGRAPH (2004)
44. Tewari, A., Bernard, F., Garrido, P., Bharaj, G., Elgharib, M., Seidel, H., Pérez, P., Zollhöfer, M., Theobalt, C.: FML: face model learning from videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
45. Tewari, A., Zollhöfer, M., Garrido, P., Bernard, F., Kim, H., Pérez, P., Theobalt, C.: Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
46. Tewari, A., Zollöfer, M., Kim, H., Garrido, P., Bernard, F., Perez, P., Christian, T.: MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction. In: IEEE International Conference on Computer Vision (ICCV) (2017)
47. Thies, J., Zollhöfer, M., Stamminger, M., Theobalt, C., Nießner, M.: Face2Face: Real-time Face Capture and Reenactment of RGB Videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
48. Tran, A.T., Hassner, T., Masi, I., Medioni, G.: Regressing robust and discriminative 3d morphable models with a very deep neural network. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
49. Tran, L., Liu, F., Liu, X.: Towards high-fidelity nonlinear 3D face morphable model. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
50. Tran, L., Liu, X.: Nonlinear 3D face morphable model. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
51. Vlasic, D., Brand, M., Pfister, H., Popović, J.: Face transfer with multilinear models. In: ACM SIGGRAPH (2005)
52. Weise, T., Bouaziz, S., Li, H., Pauly, M.: Realtime performance-based facial animation. In: ACM SIGGRAPH (2011)
53. Wu, C., Shiratori, T., Sheikh, Y.: Deep incremental learning for efficient high-fidelity face tracking. ACM Transactions on Graphics **37**(6) (Dec 2018)
54. Yang, J., Deng, J., Zhang, K., Liu, Q.: Facial shape tracking via spatio-temporal cascade shape regression. In: IEEE International Conference on Computer Vision Workshops (ICCVW) (2015)
55. Yin, L., Wei, X., Sun, Y., Wang, J., Rosato, M.J.: A 3D facial expression database for facial behavior research. In: IEEE International Conference on Automatic Face and Gesture Recognition (FGR). pp. 211–216 (2006)

56. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (Oct 2016)
57. Zhu, W., Wu, H., Chen, Z., Vesdapunt, N., Wang, B.: Reda:reinforced differentiable attribute for 3d face reconstruction. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
58. Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S.Z.: Face alignment across large poses: A 3D solution. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
59. Zollhöfer, M., Thies, J., Garrido, P., Bradley, D., Beeler, T., Pérez, P., Stamminger, M., Nießner, M., Theobalt, C.: State of the art on monocular 3D face reconstruction, tracking, and applications. Computer Graphics Forum **37**, 523–550 (2018)