Self-supervised Learning of Detailed 3D Face Reconstruction

Yajing Chen, Fanzi Wu, Zeyu Wang, Yibing Song, Yonggen Ling, and Linchao Bao*

Abstract—In this paper, we present an end-to-end learning framework for detailed 3D face reconstruction from a single image. Our approach uses a 3DMM-based coarse model and a displacement map in UV-space to represent a 3D face. Unlike previous work addressing the problem, our learning framework does not require supervision of surrogate ground-truth 3D models computed with traditional approaches. Instead, we utilize the input image itself as supervision during learning. In the first stage, we combine a photometric loss and a facial perceptual loss between the input face and the rendered face, to regress a 3DMM-based coarse model. In the second stage, both the input image and the regressed texture of the coarse model are unwrapped into UV-space, and then sent through an image-to-image translation network to predict a displacement map in UV-space. The displacement map and the coarse model are used to render a final detailed face, which again can be compared with the original input image to serve as a photometric loss for the second stage. The advantage of learning displacement map in UV-space is that face alignment can be explicitly done during the unwrapping, thus facial details are easier to learn from large amount of data. Extensive experiments demonstrate the superiority of the proposed method over previous work.

Index Terms—3D face reconstruction, self-supervised learning, depth displacement, coarse-to-fine model.

I. INTRODUCTION

RECOVERY of the 3D human facial geometry from a single color image is an ill-posed problem. Existing methods typically employ a parametric face modeling framework named as 3D morphable model (3DMM) [1]. In a 3DMM there are a set of facial shapes and texture bases, which are built from real-world 3D face scans. A linear combination of these bases synthesizes a 3D face model. During the training process, a loss function is constructed to measure the difference between the input face image and the 3D face models. The linear coefficients (i.e., 3DMM parameters) can be generated by minimizing the computed loss. While conventional methods learn these coefficients via analysis-by-synthesis optimization [2], [3], recent studies have shown the effectiveness of regressing 3DMM parameters using CNN based approaches [4], [5], [6], [7], [8].

Learning to regress 3DMM parameters via CNN requires a large amount of data. For methods based on supervised learning, the ground-truth 3DMM parameters are generated by optimization-based fitting [4], [5] or synthetic data generation [10], [11]. The limitations appear that the generated ground-truth labels are not accurate and the synthetic data lacks realism. In comparison, the methods [6], [8] based on self-supervised learning[1] do not employ this process while directly learning from unlabeled face images. For example, MoFA [6] learns to regress 3DMM parameters by forcing the rendered images to have similar pixel colors as input images in facial regions. However, enforcing the pixel level similarity does not imply similar facial identities. Genova et al. [8] rendered several face images from multiple views. They use a face recognition network to measure the perceptual similarity between the input faces and the rendered faces. Although the method is capable of producing 3D models resembling the faces in the input images, it ignores local facial characteristics such as skin colors and facial expressions and leads to unfaithful reconstructions.

In order to model facial details beyond 3DMM, a few deep learning methods recently are proposed. While some methods [12], [13] represent 3D faces completely without 3DMM, severely degraded results are usually obtained. More robust approaches typically represent 3D faces with detail layers in addition to 3DMM [14], [9], [15]. For example, learned parametric correctives are employed in [9], and 3D detail

1We in this paper do not distinguish between the term “self-supervised” and “unsupervised”, as both refer to learning without ground-truth annotations in our case. We prefer the term “self-supervised”.

Fig. 1. Reconstruction of 3D faces of Tewari18[9] and the proposed model.
maps are employed in \[14, 15\]. Since the learned parametric correctives \[9\] have very limited expressive capabilities (see Fig. 1), we advocate 3D detail maps for detail modeling. However, existing approaches employing detail maps \[14, 15\] rely on surrogate ground-truth detail maps computed from traditional approaches, which are error prone and limit the fidelity of the reconstruction.

In this paper, we propose a two-stage framework to regress 3DMM parameters and reconstruct facial details via self-supervised learning. In the first stage, we use a combination of multi-level loss terms to train the 3DMM regression network. These loss terms consist of low-level photometric loss, mid-level facial landmark loss, and high-level facial perceptual loss, which enable the network to preserve both facial appearances and identities. In the second stage, we employ an image-to-image translation network to capture the missing details. We unwrap both the input image and the regressed 3DMM texture maps into UV-space. The corresponding UV maps are together sent into the translation network to obtain the detailed displacement map in UV-space. The displacement map and the 3DMM coarse model together are rendered to a final face image, which is enforced to be photometric consistent with the input face image during training. Finally, the whole network can be trained end-to-end without any annotated labels. The advantage of the detail modeling in UV-space is that all the training face images with different poses are aligned in UV-space, which facilitates the network to capture invariant details in spatial regions around facial components with large amount of data. The main contribution of our work is that we use a self-supervised approach to solve a challenging task of detailed 3D face reconstruction from a single RGB image and we achieve very high quality results. We conduct extensive experiments and analysis to show the effectiveness of the proposed method. Compared with state-of-the-art approaches, the 3D face models produced by our method are generally more faithful to the input face images.

II. RELATED WORK

In this section, we briefly perform a literature survey on single-view 3D face reconstruction methods. These methods can be categorized as the optimization based, the supervised learning and the self-supervised learning based methods. A more complete review can be found in \[16\].

3DMM by Optimization. The 3D morphable model (3DMM) is proposed in \[1\] to reconstruct a 3D face by a linear combination of shape and texture blendshapes. These blendshapes (i.e., bases) are extracted by PCA on aligned 3D face scans. Later, Cao et al. \[17\] bring facial expressions into 3DMM and introduce a bilinear face model named FaceWarehouse. Since then, reconstructing a 3D face from an input image can be formulated as generating the optimal 3DMM parameters including shape, expression, and texture coefficients, such that the model-induced image is similar to the input image in predefined feature spaces. Under this formulation, the analysis-by-synthesis optimization framework \[2, 3, 18, 19\] is commonly adopted. However, these optimization-based approaches are parameter sensitive. Unrealistic appearances exist on the generated 3D model.

3DMM by Supervised Learning. Methods based on supervised learning requires ground-truth 3DMM labels by either optimization-based fitting or synthetic data rendering. Zhu et al. \[4\] and Tran et al. \[5\] use 3DMM parameters generated by optimization-based approaches as ground-truth to learn their CNN models. The performance of these methods are limited by unreliable labels. On the other hand, other approaches \[11, 10, 7\] try to utilize synthetic data rendered with random 3DMM parameters for supervised learning. Dou et al. \[11\] propose to use synthetic face images and corresponding 3D scans together for network learning. Richardson et al. \[10\] train a 3DMM regression network with only synthetic rendered face images. Kim et al. \[7\] show that training with synthetic data can be adapted to real data with the bootstrapping algorithm. However, the performances of these methods are limited by the unrealistic input and the 3D face models do not resemble the input images.

3DMM by Self-supervised Learning. Self-supervised methods derive supervisions by using input images without labels. MoFA \[6\] uses a pixel-wise photometric loss to ensure the rendered image induced by the estimated 3DMM parameters to be similar to the input image. However, the photometric loss makes the network attend to pixel-wise similarity between rendered image and input image, while the personal identity of the input face are ignored. Recently, Genova et al. \[8\] propose to enforce the feature similarity between the rendered image and the input image via a fixed-weight face recognition network. The facial identity can be preserved while the low-level feature similarity is missing (e.g., illumination, skin color, and facial expressions). This is because Genova et al. \[8\] are designed to predict 3DMM parameters from illumination and expression invariant facial feature embeddings instead of original face images.

Detail Modeling beyond 3DMM. Due to the limited expressive power of 3DMM, some approaches try to model facial details with additional layers built upon 3DMM. Examples following this line include the depth maps \[13\] or bump maps \[15\], and trainable corrective models \[9\]. Besides, some other work employs non-parametric 3D representations \[12, 13\] to gain more degree of freedoms, but usually are less robust than the methods built upon 3DMM. We in this paper focus on 3D representations with a coarse 3DMM model with additional detail layers. For the detail layer representation, we prefer detail maps rather than trainable corrective models \[9\] due to more expressive power. Note that existing detail map based approaches \[14, 15\] are all based on supervised learning with surrogate ground-truth detail maps computed with traditional methods, while our approach are completely unsupervised.

In this paper, we proposed a self-supervised model to learn 3D shape and texture in a coarse-to-fine procedure. In the coarse model, different from MoFA \[6\] and Genova et al. \[8\], we combine the use of low-level photometric loss and high-level perceptual loss to provide multi-level supervision for the coarse model. In the fine model, we use the unwrapped input
image and 3D model in the UV space as inputs to the neural network, and make the network learn the detailed information from the difference between real and render images in an aligned space. Different from Guo et al. [20] that used RGBD data as model inputs and learns the per-vertex normal displacement with UV maps, we use RGB images as inputs and build a UV render layer that builds dense correspondence between UV space and the rendered image space. And These differences are vital to better face reconstruction.

III. PROPOSED ALGORITHM

In this section, we illustrate the proposed method. We will give an overview of the proposed framework, and provide a detailed explanation of the loss functions to train the model.

A. Framework Overview

Figure 2 shows the pipeline of the proposed framework. It consists of two modules. The first one is the 3DMM encoder and the second one is the detail modeling encoder-decoder network. The 3DMM encoder regresses the 3DMM parameters of an input image. In the detail modeling module, we unwrap both the input image and the corresponding 3D model into UV space. We send these two UV maps into the detail modeling encoder-decoder to measure their difference where the missing details on the 3D face model resides. The output of the decoder is a displacement depth map that contains the missing details. The displacement depth map is then added back to the coarse model to enhance fine details on the surfaces. We are able to achieve the goal of self-supervision two technical novelties. First, we propose to combine low-level photometric loss and high-level perceptual loss for the coarse model regression. Second, we unwrap both the input image and the regressed 3DMM texture maps into UV-space, and the corresponding UV maps are together sent into the translation network to obtain the detailed displacement map in UV-space.

We exploit the potential of the 3DMM encoder during the training process. We propose multiple loss terms to minimize the difference between the input image and the output image produced by a differentiable render layer from the coarse 3D face model. The difference is measured from hierarchical levels of facial perception ranging from low-level photometric difference to high-level perceptual difference. The learned 3DMM encoder is capable of representing the input image in this parametric model. In the fine detail modeling stage, we mainly use photometric loss to compare the difference between the input image and the output image rendered by a differentiable UV render layer. But due to the limitation in image resolution, the input image often contain noise that might confuse the fine network with facial details. Thus, additional smoothing losses and regularization terms are included to encourage the proposed model to reconstruct smooth faces with fine details.

B. 3DMM Encoder

We first revisit the 3DMM and explain the 3DMM encoder. Then, we introduce hierarchical loss terms to exploit the network potentials. The 3DMM parameters for shape include the identity and expression parameters, the reconstructed 3D face model can be written as:

\[
s = \bar{s} + B_{id}x_{id} + B_{exp}x_{exp}
\]

(1)

where \(\bar{s}\) is the vector format of the mean 3D face model, \(B_{id}\) and \(B_{exp}\) are the identity bases and the expression bases from [1] and [17], respectively. The 3D reconstruction is formulated as regressing 3DMM parameters \(x_{id}\) and \(x_{exp}\) as shown in Eq. [1]

Our 3DMM encoder shares similarity with existing methods [6], [8]. It takes a color face image as input and transforms it progressively to the latent code vector using multiple convolutional layers and nonlinear activations. Specifically, we adopt the VGG-Face [21] structure in the 3DMM encoder. As we notice the feature representation discrepancies between 2D face images and 3D face models, we randomly initialize the network parameters and train them from scratch. During the training process, we project the output 3D face model into a 2D face image. The loss functions are mainly designed to measure the difference between the projected face and the input face. The total loss function for training the 3DMM-based coarse model is denoted as:

\[
L_{\text{coarse}} = w_1 \cdot L_{\text{pixel}} + w_2 \cdot L_{\text{lm}}
\]

\[
+ w_3 \cdot L_{\text{id}} + w_4 \cdot R_{\text{param}},
\]

(2)

where \(L_{\text{pixel}}\) is the photometric loss, \(L_{\text{lm}}\) is the landmark consistency loss, \(L_{\text{id}}\) is the perceptual identity loss, and \(R_{\text{param}}\) is the 3DMM parameter regularization term. The weights \([w_1, w_2, w_3, w_4]\) control the influence of each term and are set as constant values. We will illustrate the detailed loss formulation in the following.

1) Photometric loss: The photometric loss is set to measure the pixel-wise difference between the input face image and the rendered face image. We denote the input face image as \(I\) and the rendered face image as \(I^R\). The loss function is defined as follows:

\[
L_{\text{pixel}} = \frac{1}{|M|} \sum_{(i,j) \in M} ||I_{i,j} - I_{i,j}^R||_2
\]

(3)

where \(M\) the visible pixels on the image and \((i, j)\) is the location of each visible pixel. We compute the photometric loss by averaging the \(L_{2,1}\)-distances for all visible pixels.

2) Landmark Consistency Loss: The landmark consistency loss measures the \(L_2\)-distance between the 68 detected landmarks in the input face image and the rendered locations of the 68 key points in the 3D mesh. The 2D landmarks on the images are detected by [22]. The loss function is defined as:

\[
L_{\text{lm}} = \frac{1}{N} \sum_{i=1}^{N} ||p_i^R - p_i||_2^{\frac{3}{2}}
\]

(4)

where \(p_i\) is the \(i\)-th landmark position in the input face image, \(p_i^R\) is the corresponding \(i\)-th landmark position in the rendered face image, and \(N = 68\) is the number of landmarks. The landmark consistency loss effectively controls the pose and expression of the 3D face model based on the guidance of the input face image.
3) Perceptual Identity Loss: The perceptual identity loss reflects the perception similarity between two images. We send both the input face image and the rendered face image into the VGG-face recognition network \[21\] for feature extraction. We denote the extracted CNN features of the input face image as \(\phi(I)\), the features of the rendered face image as \(\phi(I^R)\). The perceptual consistency loss is defined as:

\[
L_{id} = ||\phi(I) - \phi(I^R)||_2^2
\]  

(5)

where \(\phi\) is the parameters of the VGG-face network \[21\] and is kept fixed during the training process.

4) Regularization Term: We propose a regularization term for the 3DMM parameters. Since the values of 3DMM parameters subject to normal distribution, we have to prevent their values from deviating from zeros too much. Otherwise, the 3D faces reconstructed from the parameters are distorted. The regularization term is:

\[
R_{param} = \omega_s||x_{id}||^2 + \omega_c||x_{exp}||^2
\]  

(6)

where \(\omega_s\) and \(\omega_c\) are constant values.

C. Detail Modeling

The detail modeling framework is an image-to-image translation network in UV space. The framework consists of an encoder-decoder network with skip connections. The input image and the reconstructed coarse 3D face model are unwrapped into two UV texture maps of the same resolution. These two UV maps are concatenated and the invisible regions are masked out based on the estimated 3D poses. Then, the concatenated UV maps are fed into the encoder-decoder network. The network produces a displacement depth map. This map is added to the UV position map of the coarse 3DMM to generate a refined UV map, which is wrapped back to a 2D face image by the UV render layer. We compare the output 2D image of the detail modeling network and the input image by a photometric loss. The refined UV position map is used to reconstruct a 3D face with fine details. During training, we add smoothness loss and regularization terms together with the photometric loss to facilitate the network learning process. The smoothness loss and regularization terms are set on the displacement depth maps to reduce both artifacts and distortions in the reconstruction process.

The total loss function of the detail modeling network is:

\[
L_{\text{fine}} = \omega_p \cdot L_{\text{pixel}} + \omega_s \cdot L_{\text{smooth}} + \omega_d \cdot L_{\text{disp}}
\]  

(7)

where \(L_{\text{pixel}}\) is the photometric loss to measure the pixel-wise difference between the input image and warped 2D face image from detail enhanced UV map. The \(L_{\text{smooth}}\) term is the smoothness loss, and \(L_{\text{disp}}\) is the regularization term on displacement map. The weights \(\{\omega_p, \omega_s, \omega_d\}\) are constant values to balance the influence of each loss term. We will introduce the smoothness loss and the regularization terms in the following:

1) Smoothness Loss: We propose the smoothness loss on both the UV displacement normal map and the displacement depth map to ensure the similar representation of the neighboring pixels on these maps. Another advantage of the smoothness loss is that it ensure the robustness to mild occlusions. The smoothness loss can be written as:

\[
L_{\text{smooth}} = \sum_{i \in V_{UV}} \sum_{j \in N(i)} w_{sn} ||\Delta n(i) - \Delta n(j)||^2 + w_{sz} ||\Delta z(i) - \Delta z(j)||^2
\]  

(8)

where \(\Delta n(i)\) is the difference measurement on pixel \(i\) in the UV map. It computes the pixel distance between the original UV normal map and the UV normal map integrated with the displacement depth map. Similarly, \(\Delta z(i)\) computes the pixel distance between the original displacement depth map and the updated displacement depth map. The \(V_{UV}\) are vertices in the UV space and \(N(i)\) is the neighborhood of vertex \(i\) with a radius of 1. The \(\Delta n(i)\) measures the difference between the UV normal map before and after adding displacement map. The weights \(w_{sn}\) and \(w_{sz}\) are used to combine these two
smoothing losses and they are empirically set as 20 and 10, respectively.

2) Regularization Term: We propose the regularization terms on both the displacement depth map and the displacement normal map to reduce severe depth changes, which may bring face distortion on the 3D mesh. The regularization term can be written as:

$$L_{\text{disp}} = \sum_i w_{dn} ||\Delta n(i)||^2 + w_{dz} ||\Delta z(i)||^2$$ (9)

where $w_{dn}$ and $w_{dz}$ are set to 0.5 and 0.01, respectively.

D. Camera View

The pose parameter in the proposed model is 7D, including scale $f$, rotation angles (in rads) $r_x, r_y, r_z$, and translation $t_x, t_y, t_z$. We apply orthogonal projection to project the 3D vertices into 2D. We denote the vertex in 3D as $v$, the projection operation as $\Pi$, the projected 2D points as $p$, respectively. Then we have:

$$p = \Pi(fRv + t),$$ (10)

where $R$ is the rotation matrix computed by $r_x, r_y, r_z$, and $t = (t_x, t_y, t_z)^T$ is the translation vector.

E. Rendering Layer

The rendering layer is a modification to [8]. We use spherical harmonics as our lighting model instead of the Phong reflection model. And we use orthogonal projection instead of full perspective projection.

F. UV Render Layer

The UV render layer takes two inputs. One is a coarse UV position map that is built by unwrapping the coarse face mesh. The other is a predicted displacement depth map in UV space from the detailed model. Using these two inputs, the UV render layer first computes the detail UV map by adding the displacement to the coarse UV map. Then, for each pixel coordinates, it finds the best correspondence in UV map and fills in the depth values. Finally renders an output image based on other parameters like pose, lighting, and texture from 3DMM regression network. The triangles for the output mesh are redefined as neighboring pixels in the output UV maps. Thus, the UV render layer performs dense rendering compared to the vertex render layer used in the coarse model.

IV. EXPERIMENTS

In this section, we illustrate the implementation details of the proposed method. Then, we evaluate the proposed method and make comparisons with state-of-the-art methods including MoFA [6], Genova18 [8] for coarse model, and Tran18 [15], Sela17 [13] for the fine model. More results are provided in the supplementary files. Our implementation will be made available to the public.

A. Implementation Details

Our network training consists of two stages. In the first stage, we train the 3DMM encoder to obtain 3DMM parameters. The reconstructed 3D face is similar to the input images in the general facial perception. In the second stage, we fix the parameters in the 3DMM encoder and apply an image-to-image translation network in the UV space to estimate a displacement depth map.

Our training dataset is from the CelebA dataset [23]. Before training, we use the landmark detector [22] to exclude the failure samples. Then, we separate the remaining images into two parts. The first part is the training dataset which contains 162,129 images. The second part is the testing set which contains 19,899 images. The network structure of the 3DMM encoder is the same as that of the VGG-face model [21]. We randomly initialize all the weights in our network and train them from scratch.

When projecting the 3D face models into 2D images, we follow [8] to construct a differentiable layer. The weight values $\{\omega_1, \omega_2, \omega_3, \omega_4\}$ controlling the total loss for the coarse model shown in equation 2 are set as $\{1.3, 1.0, 1.5, 20.0\}$, respectively. We set the weight values $\{\omega_s, \omega_e, \omega_l\}$ in the regularization terms shown in equation 6 as $\{1.3, 1.0, 1.3\}$, respectively. The weight values $\{\omega_p, \omega_n, \omega_d\}$ of the total loss for the fine model shown in equation 7 are set as $\{1.0, 10.0, 10.0\}$. The training process for the coarse model used an initial learning rate as 0.0001 and it decayed every 5000 steps at rate 0.9. The learning rate to train the fine model was set to 0.002 and it decayed every 5000 steps at rate 0.98. The batch size was set to 10. We adopted Adam optimizer to train the network on NVIDIA Tesla M40 for over 200,000 steps for the coarse model and 20,000 steps for the fine model.

B. Evaluation on 3DMM Regression

1) Shape Analysis: We evaluate the shape reconstruction precision for different configurations on the MICC Florence dataset [25]. In this dataset, videos are taken on the 53 subjects under three different conditions. These three different conditions are defined as Indoor Cooperative, PTZ Indoor and PTZ Outdoor. The ground truth 3D scans are provided for 52 out of the 53 people. We used each video frame as the network input. Before evaluation, we remove the frames where the faces are not detected by the landmark detector [22]. The 3D shape for each video sequence is reconstructed using the average of the shape parameters in the remaining frames. We follow the procedures mentioned in [8] to compute the point-to-plane error between the predicted 3D face models and ground truth scans. Table II lists all the models trained using configurations for comparison. Existing methods [6, 8] use either the photometric loss or the identity loss to train the encoder. Their performance is similar under all the three conditions. When we involve both photometric loss and identity loss during training, the performance is improved compared with [6, 8], which adopt only one of the two losses for training.

We perform the qualitative evaluation in three parts as shape and texture, expression, lighting and albedo. Figure 3 shows
Fig. 3. Shape and texture reconstructions of 3D faces by three approaches on the CelebA [23] and LFW [24] datasets.
the test images and their reconstructed shape and texture results on the CelebA [25] and the LFW [24] datasets. The proposed method is compared with MoFA [6] and Genova18 [8] for performance evaluation. The predicted expression parameters of MoFA [6] and the proposed method are excluded during this comparison.

Compared with MoFA [6] and Genova18 [8], the proposed method is able to generate faces with varying global shapes according to the identity difference in the input images. Though changes in overall shapes can also be observed for Genova18 [8], they are inconsistent with the input images sometimes. For example, the generated face by Genova18 [8] at Row 7 is too short along the vertical axis compared with the corresponding input image.

To show the results more clearly, we manually adjust the generated faces to almost the same size. We notice that some reconstructions by Genova18 [8] are extremely large or small, while MoFA [6] and the proposed approach produce faces of more appropriate size. This phenomenon may indicate that the landmark and photometric loss in MoFA [6] and the proposed approach can provide constraints on face size.

When focusing on the texture, we notice that MoFA [6] tends to generate texture with smooth and shallow colors. The results from Genova18 [8] are more realistic but do not show sufficient skin color distinction between people from different races. In contrast, the proposed method shows more color diversity for individuals.

2) Expression Analysis: We evaluate the expression reconstruction results on the FaceWarehouse dataset [17]. The dataset contains 150 subjects in 20 different expressions. We compare the proposed method with MoFA [6]. We find the correspondence between BFM 2009 Model [11] and the FaceWarehouse model [17]. Then, we apply rigid transforms to align the predicted meshes and the ground truth scans provided in FaceWarehouse [17]. We compute the point-to-point errors as the root-mean-square-error of the distances for the corresponding vertices of the two meshes.

Table III shows the mean and standard deviation of the point-to-point rmse error on the FaceWarehouse [17] dataset. The proposed method outperforms MoFA [6] significantly, indicating that the proposed method can reconstruct facial expressions with higher precision.

Since Genova18 [8] does not estimate expression parameters, we only make comparisons between MoFA [6] and the proposed model for expression on FaceWarehouse dataset [17]. Figure 4 shows the expression reconstruction results of the two methods. When the commonly-seen expression (i.e., smiling with visible teeth) appears on the subjects of the input image as shown in the first row, both MoFA and the proposed method are effective to reconstruct the 3D model. Meanwhile, the 3D model generated by the proposed method contains more identity-specific details. When some uncommon expression appears (i.e., pouted mouths) on the second row, MoFA does not reconstruct the 3D model effectively while the proposed method does. When the expression is extreme (i.e., largely-open mouths and closed eyes) as shown in the last row, neither of these two methods performs well. However, the proposed method still performs favorably against MoFA.

3) Lighting and Albedo: Figure 5 shows the visualization of the albedo and lighting reconstruction results from MoFA and the proposed method. We set the meshes as white for a clear display. Though the overall color looks similar between these two results, the lighting and albedo are different. The colors in the overlay of MoFA [6] are mostly from lighting, which leads to smooth and fair albedo. In comparison, the albedo for the proposed method is more consistent to the input faces, and the lighting color is close to white, which is more common in the real-world scenarios. The satisfying performance of the proposed approach is owing to the use of lighting-insensitive identity loss, which assists the network in decoupling lighting and albedo.

C. Evaluation on Final Reconstruction

We evaluate the visual performance of the 3D models reconstructed by different methods. The input testing images are from the CelebA [25] dataset. Figure 6 shows the evaluation results. We observe that the 3D face models generated by Tran18 are often noisy, as high frequency information are spread all over the meshes regardless of the input images. For example, on the third row of fig. 6, the mouth region on the input face is smooth with salient texture, but the corresponding 3D face model for Tran18 is noisy on the mouth. Meanwhile, the wrinkles on the cheeks are not reconstructed well according to the input image. Similar phenomena appears on other 3D face models which are not faithfully represent the input faces. On the other hand, the 3D models generated by Sela17 are erroneous and do not accurately convey the facial components of the input face images. Also, it turns out that sometimes Sela17 fails to generate face-like meshes. In comparison, the proposed method effectively generates 3D

| Method     | Loss | Condition |
|------------|------|-----------|
|            | id   | Indoor Cooperative | PTZ Indoor | PTZ Outdoor |
| MoFA [6]   | ✓    | 1.38 ± 0.35 | 1.27 ± 0.29 | 1.28 ± 0.27 |
| Genova18 [8] | ✓    | 1.41 ± 0.37 | 1.34 ± 0.37 | 1.26 ± 0.31 |
| Ours-3DMM  | ✓    | 1.35 ± 0.31 | 1.27 ± 0.24 | 1.25 ± 0.21 |

Table II point-to-plane error on the MICC Florence dataset [25] for coarse 3D face reconstructing methods.

MEAN AND STANDARD DEVIATION OF POINT-TO-PLANE ERROR ON FACEWAREHOUSE [17] FOR MOFA AND THE COARSE MODEL OF THE PROPOSED METHOD.

| Method | MoFA [6] | Ours |
|--------|----------|------|
| Error  | 2.26 ± 0.68 | 1.81 ± 0.43 |
models preserving the global shape and structure. Besides, the facial details are enriched on the generated 3D models.

Figure 7 compares the results of 3DMM regression and the fine model. We can see that the fine model can retain more details and express more vivid facial characteristics than using merely the 3DMM model. In addition, we compared the close-ups of the reconstruction results by the fine model with Tran18[15] and Sela17[13]. The result is shown in Figure 8. Compared with Tran18 and Sela17, our fine model strengthens the details with less noise or distortion, showing more robustness over the other two approaches. Also, our model closely resembles the input images as well as preserves the detailed wrinkles. As in Row 4, distortion in the results of Tran18 and Sela17 can be easily observed, making the reconstructed mouth unnatural. What's more, the positions of these wrinkles shift in Tran18 and Sela17, while our model successfully reconstruct the details.

We perform quantitative evaluation on the MICC Florence dataset [25] and the Face Recognition Grand Challenge V2 (FRGC2) dataset [26]. In MICC, we evaluate the shape reconstruction precision and in FRGC2 we evaluate the depth estimation errors. We first evaluate the shape reconstruction precision on MICC dataset. As we aim to model facial details, we select frontal video frames of each subject for evaluation. The frontal frames only exist in the Indoor Cooperative condition. We compute the point-to-point error between the reconstructed 3D faces and the ground truth scans. We first crop the ground truth scan to 95mm around the tip of the nose. Then, we run ICP (i.e., iterative closed points) with isotropic scale to find an alignment between the ground truth and the reconstruction. The Point-to-point distances are then computed for each subject.

Table V shows the comparison results. It indicates that the
proposed method achieves lower error under both average and standard deviations in the MICC dataset. The lower error demonstrates that the proposed method can reconstruct fine details with higher accuracy and stability than the other methods. During the evaluation process, Sela17 [13] fails to reconstruct 5 subjects, while Tran18 [15] and the proposed method successfully reconstruct all subjects. The failure reconstruction results appear in Sela17 [13] where the faces are distorted or even not reconstructed. Thus, ICP are unstable for Sela17 and the resulting error may be huge. Figure 9 shows some error maps and Figure 10 shows the individual errors. Sela17 is not robust compared with Tran18 and the proposed method with higher error and standard deviation metrics. On the other side, the proposed method stably produces state-of-the-art results.

Besides MICC, we also evaluate on the FRGC2 datasets where we estimate the depth of the input face images. There are around 5000 images and the corresponding ground-truth depth maps in this dataset. We evaluate the depth estimation results generated from all the methods. To calculate the depth error, we first scale the depth estimation of each method to fit the ground truth depths in ranges. Then the mean distance...
| Method    | Tran18 [15] | Sela17 [13] | Ours  |
|-----------|-------------|-------------|-------|
| Error(mm) | 2.61 ± 0.66 | 3.19 ± 0.58 | 2.41 ± 0.62 |

TABLE V
POINT-TO-POINT ERROR ON MICC FLORENCE [25] FOR DIFFERENT FINE-DETAIL RECONSTRUCTING APPROACHES.

between the two depth maps at valid pixel positions provided by a fixed binary mask are computed as depth error. Table IV shows the depth estimation results from these methods. The proposed method achieve lowest mean and standard deviation in depth errors compared with the other approaches. The low depth errors indicate that the proposed model generate 3D faces with higher accuracy.

D. Application

An additional advantage of employing UV position maps for reconstruction is that it enables easier integration of 3D reconstructions from different views of a same face. The UV maps from different views can be easily combined by a simple blending in UV-space. Thus the combined full UV map can represent a complete 3D face model that are visible in different views. The induced detailed 3D reconstruction are more complete compared to depth map based representations. Figure 11 shows several examples of blending two partial UV maps from two views of a same face.

E. Limitation Analysis

Figure 12 shows some failure cases for the proposed model under difficult scenarios including extreme expression, occlusion and large pose. Since the proposed model are not trained with specific policy to deal with these difficult conditions, whenever the input images show extreme expressions or missing information due to large pose or occlusion, the reconstruction results are often unsatisfactory.

V. CONCLUSION

We propose a detailed 3D face reconstruction framework with self-supervised learning. We use a coarse 3DMM encoder to reconstruct the general 3D face model and capture facial details in the UV space. When learning the 3DMM encoder, we incorporate multiple loss terms measurement ranging from the pixel-wise similarity to the global facial perception. After learning the 3DMM parameters, we unwrap both input face image and 3D model into UV space where all the faces...
are precisely aligned. The details from the input image are effectively transferred to the 3D model in the detail modeling step as the aligned facial details facilitate the learning process. Experiments on the benchmark datasets indicate the proposed method performs favorably against state-of-the-art 3D face modeling approaches.

REFERENCES

[1] V. Blanz and T. Vetter, “A morphable model for the synthesis of 3d faces,” in ACM Transactions on Graphics (Proceedings of SIGGRAPH), 1999.

[2] ——, “Face recognition based on fitting a 3d morphable model,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003.

[3] S. Ronchetti and T. Vetter, “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, 2005.

[4] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, “Face alignment across large poses: A 3d solution,” in IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[5] A. T. Tran, T. Hassner, I. Masi, and G. Medioni, “Regressing robust and discriminative 3d morphable models with a very deep neural network,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[6] A. Tewari, M. Zollhöfer, H. Kim, P. Garrido, F. Bernard, P. Pérez, and C. Theobalt, “Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction,” in IEEE International Conference on Computer Vision, 2017.

[7] H. Kim, M. Zollhöfer, A. Tewari, J. Thies, C. Richardt, and C. Theobalt, “Inversion on Graphs: Deep monocular inverse face rendering,” in IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[8] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman, “Unsupervised training for 3d morphable model regression,” in IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[9] A. Tewari, M. Zollhfer, P. Garrido, F. Bernard, H. Kim, P. Prez, and C. Theobalt, “Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz,” in IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[10] E. Richardson, M. Sela, and R. Kimmel, “3d face reconstruction by learning from synthetic data,” in International Conference on 3D Vision, 2016.

[11] P. Dou, S. K. Shah, and I. A. Kakadiaris, “End-to-end 3d face reconstruction with deep neural networks,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[12] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos, “Large pose 3d face reconstruction from a single image via direct volumetric cnn regression,” in IEEE International Conference on Computer Vision, 2017.

[13] M. Sela, E. Richardson, and R. Kimmel, “Unrestricted facial geometry reconstruction using image-to-image translation,” in IEEE International Conference on Computer Vision, 2017.

[14] E. Richardson, M. Sela, R. Or-El, and R. Kimmel, “Learning detailed face reconstruction from a single image,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[15] A. T. Tran, T. Hassner, I. Masi, E. Paz, Y. Nirkin, and G. Medioni, “Extreme 3D face reconstruction: Seeing through occlusions,” in IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[16] M. Zollhöfer, J. Thies, P. Garrido, D. Bradley, T. Beeler, P. Pérez, M. Staminger, M. Nießner, and C. Theobalt, “State of the art on monocular 3d face reconstruction, tracking, and applications,” in Computer Graphics Forum, 2018.

[17] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, “Facewarehouse: A 3d facial expression database for visual computing,” IEEE Transactions on Visualization and Computer Graphics, 2014.

[18] P. Garrido, L. Valgaerts, C. Wu, and C. Theobalt, “Reconstructing detailed dynamic face geometry from monocular video,” in ACM Transactions on Graphics (Proceedings of SIGGRAPH), 2013.

[19] J. Thies, M. Zollhofer, M. Staminger, C. Theobalt, and M. Nießner, “Face2face: Real-time face capture and reenactment of rgb videos,” in IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[20] Y. Guo, J. Zhang, J. Cai, B. Jiang, and J. Zheng, “Cnn-based real-time dense face reconstruction with inverse-rendered photo-realistic face images,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 6, pp. 1294–1307, June 2019.

[21] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep face recognition,” in British Machine Vision Conference, 2015.

[22] A. Bulat and G. Tzimiropoulos, “How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks),” in IEEE International Conference on Computer Vision, 2017.

[23] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild,” in IEEE International Conference on Computer Vision, 2015.

[24] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” Tech. Rep., 2007.

[25] A. D. Bagdanov, A. Del Bimbo, and I. Masi, “The florence 2d/3d hybrid face database,” in Joint ACM Workshop on Human Gesture and Behavior Understanding, 2011.

[26] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of the face recognition grand challenge,” in IEEE Conference on Computer Vision and Pattern Recognition, 2005.

[27] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, “Joint 3d face reconstruction and dense alignment with position map regression network,” in European Conference on Computer Vision, 2018.