Joint 3D facial shape reconstruction and texture completion from a single image

Xiaoxing Zeng^{1,2}, Zhelun Wu¹, Xiaojiang Peng¹, and Yu Qiao¹ (✉)

© The Author(s) 2021.

Abstract Recent years have witnessed significant progress in image-based 3D face reconstruction using deep convolutional neural networks. However, current reconstruction methods often perform improperly in self-occluded regions and can lead to inaccurate correspondences between a 2D input image and a 3D face template, hindering use in real applications. To address these problems, we propose a deep shape reconstruction and texture completion network, SRTC-Net, which jointly reconstructs 3D facial geometry and completes texture with correspondences from a single input face image. In SRTC-Net, we leverage the geometric cues from completed 3D texture to reconstruct detailed structures of 3D shapes. The SRTC-Net pipeline has three stages. The first introduces a correspondence network to identify pixel-wise correspondence between the input 2D image and a 3D template model, and transfers the input 2D image to a U - V texture map. Then we complete the invisible and occluded areas in the U - V texture map using an inpainting network. To get the 3D facial geometries, we predict coarse shape (U - V position maps) from the segmented face from the correspondence network using a shape network, and then refine the 3D coarse shape by regressing the U - V displacement map from the completed U - V texture map in a pixel-to-pixel way. We examine our methods on 3D reconstruction tasks as well as face frontalization and pose invariant face recognition tasks, using both in-the-lab datasets (MICC, MultiPIE) and in-the-wild datasets (CFP). The qualitative and quantitative results demonstrate the

effectiveness of our methods on inferring 3D facial geometry and complete texture; they outperform or are comparable to the state-of-the-art.

Keywords 3D face reconstruction; U - V completion; pose invariant face recognition; deep learning

1 Introduction

This paper attacks the problem of recovering 3D shape and complete facial texture from a single 2D face image. Image-based 3D face reconstruction is a fundamental yet essential problem in computer vision, with broad applications to facial animation [1, 2], pose-invariant face recognition [3–7], human-machine interaction [8], etc. Dramatic improvements have been made in image-based 3D face reconstruction in recent years [9–11].

Current image-based 3D face reconstruction methods can roughly be divided into two categories: model-based methods and pixel-wise regression methods. The first category utilizes 3D template models of the face for reconstruction. The *3D morphable model* (3DMM) [12–14] is a popular choice within the first category due to its simplicity and efficiency. The main defect of 3DMM, however, is its low-dimensional representation, which limits its representational ability and always results in over-smooth 3D faces. Bas et al. [15] and Feng et al. [10] use a BFM model [16] as a prior shape to densely predict the 3D face as a U - V position map. However, their methods cannot effectively reconstruct local structures of the input face. Pixel-wise regression methods [17–19] directly predict the depth or 3D locations of each pixel in the input, and can be effectively implemented with modern deep networks,

1 Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. E-mail: X. Zeng, xx.zeng@siat.ac.cn; Z. Wu, zl.wu@siat.ac.cn; X. Peng, xj.peng@siat.ac.cn; Y. Qiao, yu.qiao@siat.ac.cn (✉).

2 University of Chinese Academy of Sciences, Beijing, China.

Manuscript received: 2021-03-23; accepted: 2021-05-03

e.g., U-Net [20]. Their pixel-wise nature enables these methods to capture fine facial details, but these methods lack the dense correspondence between 2D input image and 3D facial template, which may be limiting in practical applications like expression transfer and animation. Moreover, these methods cannot handle self-occluded regions well.

Compared to 3D geometry, recovering complete facial texture has received less attention. It is an ill-posed problem since certain face regions can be self-occluded in the input image. Previous texture completion methods roughly fall into two categories: appearance-based methods [14, 21–25] and geometric modeling methods [3, 26–28]. Appearance-based methods often leverage either linear appearance models (e.g., diffuse albedo) to recover low-frequency facial appearance or generative adversarial networks (GANs) [29] to synthesize frontalized face images. However, appearance-based methods may produce unrealistic visual results and are often unable to preserve facial identity. On the other hand, geometric-modeling-based work [3, 26–28] determines dense correspondences between a 2D face image and a predefined 3D face template (model), which allows mapping of the 2D face image into U - V space and acquisition of a U - V texture map. Then, one can inpaint or complete the self-occluded textures using

the U - V map. However, 3D geometric models like 3DMM only use a limited number of coefficients or landmarks for densely correspondence to the input image with tens of thousands of pixels or more. This imbalance inevitably causes certain non-facial areas (e.g., background, hair) in the input to also be mapped into the facial texture map. Downstream inpainting algorithms cannot remove these defects, leading to poor facial texture completion.

To address the above problems, we propose a novel *shape reconstruction and texture completion network*, SRTC-Net, which jointly reconstructs 3D facial geometry and completes texture from a single input image. For texture reconstruction, we first learn the 2D-to-3D correspondence via two cascading tasks: facial segmentation and PNCC regression. Within this correspondence, then, we unwrap input images into U - V space to get U - V textures. Subsequently, we complete the U - V texture with an inpainting network. For shape reconstruction, we obtain coarse shapes with a model-based method. In addition, we extract geometric cues from the completed U - V texture with a pixel-wise method, and integrate them into the outputs of the model-based methods.

As shown in Fig. 1, our method achieves more accurate 2D-to-3D correspondence than previous methods, especially at edges. Our contributions can

Fig. 1 Examples of correspondences between 2D faces and 3D models provided by PRNet [10], Deep3DFace [11], and our method. PRNet and Deep3DFace produce artifacts caused by inaccurate 2D-to-3D correspondences (green box), while ours does not.

be summarized as follows:

1. A deep architecture, SRTC-Net, which reconstructs 3D shape and completes 3D texture jointly. SRTC-Net decomposes this ill-posed task into three more tractable problems.
2. A *corresponding network* (C-Net) to densely align the 2D image and 3D face model via facial segmentation using a segmentation subnetwork and a PNCC prediction subnetwork. This lets us establish more accurate 2D-to-3D correspondences and improves the performance of 3D texture completion results. The strong geometric cues in the completed 3D texture can also be exploited to refine 3D face shapes.
3. A *shape network* (S-Net) for 3D geometry reconstruction, which lets us reconstruct detailed structure in pixel-wise fashion while preserving topology. Specifically, the S-Net reconstructs a coarse shape with a model-based coarse-shape subnetwork, and then predicts detailed structures with a pixel-wise regression architecture in a fine-shape subnetwork. We build the correspondence between coarse and fine shapes by sharing the same U - V coordinates.
4. Extensive qualitative and quantitative experiments on both controlled and in-the-wild databases which demonstrate the effectiveness of our methods: they outperform or are comparable to state-of-the-art approaches to 3D shape reconstruction and texture completion.

2 Related work

2.1 Correspondence between 2D face images and 3D models

Determining correspondences between a 2D face image and the 3D template model is a crucial step in the U - V texture mapping pipeline. Akshay et al. [30] map a 2D face image to a 3D model by finding a rigid transformation to match the vertices of the template 3D face model to 2D feature points detected by VAAM [31]. This method, however, requires near-frontal views of constrained faces so is not suitable for our purposes. Deng et al. [3] and Xue et al. [28] apply a landmark-driven 3DMM fitting method to establish the correspondences. Chang et al. [27] employ an off-the-shelf 3D face reconstruction pipeline, PRNet [10], to align a 2D face to a 3D face

model. All these 3D face shape estimation methods can introduce facial misalignments, causing non-facial areas to invade the U - V texture map. Tapping into more empirical knowledge, Hassner et al. [26] propose use of a supplemental 3D reference model to produce front-facing views for all input images, in which landmarks are used to align the input image with the template reference image. Their method can handle 2D faces with neutral expressions. However, it distorts the final texture if the input face has an exaggerated expression. This deficiency is one of the reasons we apply the C-Net in our pipeline, to establish accurate 2D–3D correspondences to enable our framework to reconstruct high-quality 3D face textures.

2.2 Image completion

Inspired by empirical observations that there is a relationship between local and global texture, Wei et al. [32] apply a Markov random field model for image completion. Efros and Leung [33] apply non-parametric exemplar-based techniques to synthesize textures by assembling paths in the example image. Such techniques only work for stochastic textures, not an image with fine details. With the advent of deep learning, convolutional neural networks (CNNs) were also deployed in this task. Unlike the general image completion task, face completion is susceptible to significant appearance variations. To ensure genuineness and consistency between local and global content, Li et al. [34] train the face completion network utilizing local and global discriminators as well as a fixed parsing network. The U - V -GAN [3] uses the same loss function, but replaces the parsing network with a fixed identity classification network. Chang et al. [27] propose a U - V texture completion GAN (TC-GAN) to complete the U - V texture map. In our work, we go further, predicting both a U - V texture map and a U - V position map.

2.3 Pose-invariant face recognition

With deeper convolutional neural networks and advanced loss functions [35], many approaches have achieved significant improvements on unconstrained face recognition datasets. However, pose variation is still very challenging even for deep learning-based face recognition methods, due to missing facial texture caused by self-occlusion; recognition accuracy plummets when it comes to profile-frontal

recognition [36]. GAN-based face frontalization [3, 14, 24, 37, 38] is a popular method that decreases the pose discrepancy and improves frontal-profile face verification. Our work can infer both facial texture and geometry, offering an out-of-the-box face frontalization component conducive to pose invariant face recognition.

2.4 3D face reconstruction

As a popular method, the *3D Morphable Model* (3DMM) method [12] reconstructs a 3D model from a single image in a low-dimensional linear space. Tran et al. [13] regress the shape, texture, and parameter coefficients with a deep neural network. Guo et al. [39] use meta-joint optimization and a short-video-synthesis strategy to learn a lightweight network to reconstruct 3D faces quickly, accurately, and robustly. Sanyal et al. [40] use shape consistency across multiple images of the same person and inconsistency across multiple images of different person to train a 3D face reconstruction network without supervision. At the same time, non-linear 3D morphable models [41, 42] were introduced to capture more details. Bas et al. [15] encode 3D coordinates of vertices into a position map and process a 2D position map with convolutional neural network learning. Wu et al. [43] employ a symmetry constraint in a photogeometric autoencoder for unsupervised learning that disentangles shape and appearance.

Lin et al. [44] propose a self-supervised graph network to reconstruct realistic facial texture from

a single image. Lee and Lee [45] embed uncertainty learning in a deep network to reconstruct high-fidelity 3D faces. Lattas et al. [46] jointly capture facial albedo, 3D shape, and a normal map from an input image, using a cascaded pipeline. Our work can build visually vivid 3D faces free from non-facial areas, using a 2D image and a U - V texture map in the last stage.

3 Method

In this section, we first overview our shape reconstruction and texture completion network, SRTC-Net. Then we introduce our network architecture components, training details, and how we created our training data.

3.1 Overview

As Fig. 2 shows, SRTC-Net consists of three modules, the corresponding network (C-Net), the inpainting network (I-Net), and the shape network (S-Net). A face image first goes into the C-Net for face segmentation prediction in the segmentation subnetwork, and projected normalized coordinate code (PNCC) [47] prediction in the PNCC subnetwork. Using the PNCC prediction, we can map the segmented faces into U - V space to generate U - V texture maps. Then the U - V texture maps are fed into the I-Net for texture completion. Finally, we feed the segmented faces into the coarse-shape subnetwork to get coarse 3D faces, and acquire the detailed 3D face structure by predicting U - V displacement maps from complete

Fig. 2 Pipeline of our SRTC-Net. It jointly infers 3D facial geometry and complete texture from a single image via three modules: C-Net, I-Net, and S-Net.

U - V texture maps using the fine-shape subnetwork. Each subnetwork in SRTC-Net is trained separately. We now detail each component of SRTC-Net, and training data collection, in the following sections.

3.2 Corresponding Network (C-Net)

3.2.1 Overview

The C-Net consists of two pixel-wise prediction subnetworks, the segmentation subnetwork and the PNCC subnetwork. Pixel-wise prediction requires an appropriate network architecture. The proposed segmentation subnetwork and PNCC subnetwork are inspired by the *image-to-image translation framework* [20], where skip connections are used between corresponding layers in the encoder and the decoder. We also make additional modifications following Ref. [17]. The segmentation subnetwork takes in face images and outputs face masks, while the PNCC subnetwork takes segmented faces as input and outputs PNCC maps.

The output PNCC maps are used to map face images into our predefined U - V map. Each foreground pixel $I(x, y)$ of the input image has one predicted template NCC coordinate $(\hat{N}_x, \hat{N}_y, \hat{N}_z)$ stored in three channels of point $P(x, y)$ in the predicted PNCC map. Given $(\hat{N}_x, \hat{N}_y, \hat{N}_z)$, we can find the corresponding vertex V_i in the template NCC by nearest point search; V_i has a unique corresponding pixel $D(u, v)$ in the predefined U - V map. Finally, the pixel value of $I(x, y)$ is assigned to $D(u, v)$. We repeat the above process to get the U - V texture map.

3.2.2 Segmentation subnetwork training

Face images are detected and aligned with the face detector [48] at a resolution of 256×256 . The ground-truth masks are binary maps with the same resolution. We employ the binary cross-entropy loss function to train this network:

$$L_{\text{seg}} = -\frac{1}{n} \sum_{i=1}^n (\text{Seg}_{g,i} \log \text{Seg}_{p,i} + (1 - \text{Seg}_{g,i}) \log(1 - \text{Seg}_{p,i})) \quad (1)$$

where $\text{Seg}_{p,i}$ is the output of the i th pixel in the face image, Seg_g is the ground-truth, and n is the number of pixels in the face image, i.e., 256×256 .

3.2.3 PNCC subnetwork training

To reduce the computational cost, firstly, we crop the 690k PNCC-image pairs (see Section 3.6.4 using bounding squares of their images' foreground contour, and then resize them to 128×128 to give 690k cropped

PNCC-image pairs P_{pc} . We use the pixel-wise $L1$ loss between ground-truth and predicted PNCC maps to train the PNCC subnetwork:

$$L_{\text{pncc}} = \|P_g - P_p\|_1 \quad (2)$$

where P_p is the predicted PNCC map and P_g is the ground-truth.

3.3 Inpainting Network (I-Net)

3.3.1 Overview

We warp the 690k face images described in Section 3.6.4 to U - V space with their counterpart PNCC maps, and obtain 690k U - V texture maps. As these U - V texture maps are incomplete due to self-occlusion, we use the I-Net to fill in missing areas. Specifically, we introduce the adversarial training in this task.

3.3.2 Generator module

We employ the generator from Ref. [49] as our backbone network but make several modifications. Firstly, we employ skip connections between coarse layers of the encoder and decoder to preserve high-frequency details. As 10% of the above training data are half-occluded (i.e., have yaw of 90°), their U - V texture maps require about half of their area to be inpainted. Thus, we adopt dilated convolutions in coarse resolution layers following Ref. [50] to enlarge the receptive fields and provide better completion. To make our inpainted U - V texture maps more visually realistic, we superimpose a Gaussian noise map [51] on the output of each convolutional layer. We further concatenate the U - V texture map with its flipped version to form a six-channel input. We find that such a flipped U - V texture map provides better initialization for missing texture areas compared to random noise. The architecture of the generator is shown in Fig. 3. An $L1$ loss function is applied to train the I-Net:

$$L_{\text{rec}} = \|T_{\text{gt}} - T_G\|_1 \quad (3)$$

where $T_G = G(T_{\text{in}}, T_{\text{in}}^f)$, G is the generator, and T_{gt} , T_{in} , T_{in}^f are U - V texture map ground truth, input, and flipped input map, respectively.

To preserve the content, we also adopt the widely used perceptual loss, defined as

$$L_{\text{per}} = \|V(T_G) - V(T_{\text{gt}})\|_1 \quad (4)$$

where V is the activation map of a pre-trained deep model. We apply the VGGFace2 pre-trained model [52] as the feature extractor, and set pool3 as

Fig. 3 Architecture of the I-Net generator.

the activation map. We also tried an ImageNet pre-trained model [53], but did not see any improvements.

3.3.3 Discriminator module

Although we can get a complete U - V texture map using the generator, the obtained texture lacks detail. To improve the quality of the inpainted texture, we introduce adversarial learning. The loss function of the discriminator is as follows:

$$L_d = -\mathbb{E}_{T_{gt} \in \mathcal{P}} [\log(D(T_{gt}))] - \mathbb{E}_{T_G \in \mathcal{Q}} [\log(1 - D(T_G))] \quad (5)$$

where $\mathcal{P}$ and $\mathcal{Q}$ represent the distribution of ground-truth U - V texture maps and outputs of the generator, respectively. To train the discriminator, we use the following objective function:

$$L_g = -\mathbb{E}_{T_G \in \mathcal{Q}} [\log(D(T_G))] \quad (6)$$

3.3.4 Stable training

To train the I-Net stably, we enforce a gradient penalty regularization term [54] in the discriminator:

$$L_{st} = \mathbb{E}[\|\nabla D(T_{gt})\|^2] \quad (7)$$

where $\nabla D(T_{gt})$ is the gradient of the discriminator with respect to its input T_{in} .

3.4 Total loss

We train the generator and discriminator in an end-to-end way, with the final loss function is defined as

$$L_{total} = L_{rec} + \lambda_1 L_d + \lambda_2 L_g + \lambda_3 L_{st} + \lambda_4 L_{per} \quad (8)$$

We set $\lambda_1 = \lambda_4 = 0.1$, $\lambda_2 = 0.2$, $\lambda_3 = 1$, chosen experimentally.

3.5 Shape Network (S-Net)

3.5.1 Overview

To reconstruct 3D face models, firstly, we train a coarse-shape subnetwork to get coarse (i.e., smooth)

shapes using the same 690k inputs as for the PNCC subnetwork and 32k ground-truth U - V position maps described in Section 3.6.5. Then, we predict local surface structures by training a fine-shape subnetwork with 32k ground-truth U - V texture maps and position maps.

3.5.2 Coarse-shape subnetwork

We use an encoder-decoder structure to transfer the RGB face images into a U - V position map. The encoder begins with one convolution layer followed by six convolution blocks, each of which includes a leaky ReLu activation layer, a convolution layer, and a batch normalization layer, which reduce the $128 \times 128 \times 3$ input image into a $1 \times 1 \times 512$ feature map. The decoder contains ten transposed convolution blocks, each of which includes a ReLu activation layer, a transposed convolution layer, and a batch normalization layer, and predicts the $84 \times 84 \times 3$ U - V position maps. We also tried the FCN network from PRNet [10], but it did not converge.

To learn the parameters of our network, we use the reconstruction loss function defined as

$$L_{sur} = \|S_g - S_p\|_1 \quad (9)$$

where S_g is the ground-truth U - V position map, and S_p is the output of the coarse-shape subnetwork.

3.5.3 Fine-shape subnetwork

In our work, we reconstruct 3D face detail structures by predicting a single-channel U - V displacement map, which represents the displacement of the U - V position map along the z -axis. The U - V displacement map is directly added to the z -axis channel in the U - V position map to generate the final face geometry. As in a previous method [55], we assume faces to be Lambertian surfaces, and apply a shape from shading technique to capture fine details. Such an approach

has been demonstrated to be an effective way of recovering high-frequency structural details from RGB face images [17, 56, 57]. The fundamental of SFS-based refinement method is an image formation term linking the final face geometry and the input U - V texture map. The image formation term is usually defined as follows

$$\hat{I}(l, N, R) = R \sum_{i=1}^9 l_i H_i(N) \quad (10)$$

where $\hat{I}$ is the reflected irradiance, R and l are the albedo map and lighting coefficient estimated by SfSNet [58], and $H_i(N)$ are the spherical harmonic (SH) basis functions computed with unit normal N of the final face geometry. First, it is straightforward to measure the dense photometric discrepancy between the input U - V texture map I and the reconstruction $\hat{I}$ with the MSE loss function:

$$L_{\text{sh}} = \|\hat{I} - I\|_2 \quad (11)$$

where $\|\cdot\|$ denotes the l_2 norm. To prevent face shape degeneration, we add a commonly-used two-norm regularization term L_{regu} to the output U - V displacement map. The loss weights of L_{regu} and L_{sh} are 1 and 100, respectively. For the fine-shape subnetwork, we employ the same architecture as the PNCC subnetwork, since its skip links can transfer detailed information from the RGB domain to the U - V position domain.

3.6 Construction of training datasets

3.6.1 Overview

Training data plays an essential role in deep-learning-based computer vision algorithms. For our task, it is difficult and tedious to directly collect real face images together with their ground-truth 3D shapes. Existing public 3D face datasets, e.g., BU4DFE [59] and Bosphorus [60], are limited to constrained environments and a few hundred subjects, which are insufficient for training our SRTC-Net.

To address this problem, we compile our training data from three parts; some come from previous work and others are hand constructed. The first part contains image-mask pairs for the face segmentation network of the C-Net. The second one contains PNCC-image pairs to train the PNCC prediction network of the C-Net and S-Net, while the last one comprises complete U - V texture maps and U - V position maps to train the I-Net and S-Net.

3.6.2 Registered 3D face models

Most of our data generation relies on two kinds of 3D face models, so we start by introducing how to create them. We generate these face models in a way such that they have the same number of vertices, with uniform topology. Specifically, there are two kinds of registered 3D face models: the mean model and training models. For the former, we use the template model from the MeshMonk toolbox [61] as the source model, and the template model of Ref. [11] as the target model. Then we employ MeshMonk to automate a non-rigid transformation to get the mean model. To acquire training models, we non-rigidly register the template model of Ref. [61] to the texture-symmetry point clouds in Section 3.6.4. In total, we produce 32k training models.

The mean model is only used in face segmentation data generation (see Section 3.6.3). The training models are the ground-truth 3D shape in S-Net, and are also applied to generate PNCC-image pairs in Section 3.6.4.

3.6.3 C-Net face segmentation module data

We first collect two kinds of image from the Internet according to face pose; one set has 60k near-frontal face images, and the other has 7k face images with large yaw.

In order to automate the segmentation procedure on these near-frontal face images, we manually specify 60 3D landmarks on the boundary of the mean model. Then, we specify 60 counterpart 3D landmarks on the template model of Ref. [11] by finding the nearest vertex to each of the former 60 landmarks. For each of the 60k near-frontal face images, we get the predicted 3D model using the approach in Ref. [11], and project its 60 3D landmarks explained above into the image plane. Then, we find the convex hull of these 60 projected landmarks. Pixels inside the convex hull are foreground, and the others are background.

For the 7k collected face images with large yaw (from 30° to 90°), we extract facial profile contours manually, and pixels inside the contours constitute the foreground. We then apply data augmentation methods such as in-plane rotation, flipping, and random cropping to increase the number of large pose-variation image-mask pairs to 70k. The final face segmentation training data has 130k image-mask pairs.

3.6.4 C-Net PNCC prediction module data

We select 32k image-mask pairs from the 130k training data explained in Section 3.6.3, and obtain their camera parameters and 3D models following Ref. [11]. To copy the pixel values of 2D faces to 3D models, we project vertices into image planes with predicted camera parameters, and assign image pixel values to the corresponding vertex. In this process, projected vertices which lie in the background or belong self-occluded areas have their texture values set to zero. For each of the above textured models, we then displace each vertex with zero value using its symmetric counterpart; we refer to this zero-valued displaced model as a texture-symmetry point cloud. We warp the template model of Ref. [61] to all texture-symmetry point clouds, obtaining 32k textured 3D face models altogether.

Given 32k complete textured 3D face models, we generate the *normalized coordinate code* (NCC) [47] for each one, and set the texture value of each vertex to its NCC value. Next, we project the complete textured 3D face models and accompanying corresponding 3D face model with NCC texture into the image plane with poses at uniform intervals (with yaw from -90° to 90° in 10° steps), giving 690k PNCC-image pairs for PNCC subnetwork training. All images of the 690k PNCC-image pairs are also used to train the coarse-shape subnetwork of S-Net.

3.6.5 I-Net U - V texture inpainting and S-Net U - V position prediction data

To unwrap a 3D face model to the U - V plane, we need parametric model of our 3D face. We thus use a conformal parametric method [62] to unwrap the template 3D face model of MeshMonk toolbox [61] to a square map. Our template 3D model contains 7052 vertices. Although the resolution of the U - V map can be arbitrarily defined, to reduce the computational cost, we set it to 84×84 . Given the 32k complete textured 3D face models described in Section 3.6.4, we can generate 32k ground-truth U - V texture maps

and U - V position maps to train the I-Net and S-Net separately.

The training data described in Section 3.6 is summarised in Table 1.

4 Experiments

In this section, we first evaluate landmark localization on the Multi-PIE dataset to compare our C-Net and the current state-of-the-art methods. Then we assess the I-Net on the two most widely used pose invariant face recognition datasets [36, 63] by frontalizing faces with large pose variations. Finally, we compare our S-Net against currently popular 3D face reconstruction methods both qualitatively and quantitatively. In this section, to detect five keypoints of extreme-pose faces for alignment, we employ the OpenFace 2.0 toolbox [64].

4.1 Evaluation of C-Net

Since there is no publicly available dataset for comparing C-Net with previous methods on the dense correspondence between 2D images and 3D models, we measure the sparse landmark localization error on face images of varying poses instead. Visible landmarks on the face contour are significant indicators, as they are the boundary for U - V texture mapping. The Multi-PIE [63] dataset contains face images with large pose variation with 39 visible landmarks, among which we select images with extreme poses $[\pm 60^\circ, \pm 90^\circ]$ from Session 1 as our test images. We discard 12 landmarks in areas not covered by our mesh and measure the remaining 27 landmarks inside the face profile on 256 images. We compare C-Net to 3DDFA methods [47], PRNet [10], and Deep3DFace [11]. For a fair comparison, we adopt the normalized mean error (NME) as our evaluation metric; the average landmark error is normalized by the nose length (distance from Point 1 to Point 4, using Multi-PIE landmark labels). Table 2 shows that our method outperforms previous methods. We give various profile landmark localization results in Fig. 4, demonstrating that C-Net can capture an

Table 1 Training data for different modules in SRTC-Net

	Segmentation subnet	PNCC subnet	I-Net	Coarse-shape subnet	Fine-shape subnet
Image-mask	✓				
PNCC-image		✓	✓	✓	
U - V texture map			✓		✓
U - V position map				✓	✓

Fig. 4 Qualitative evaluation of C-Net. Top to bottom: original images, ground-truth profile landmarks, and the results from C-Net.

Table 2 Profile landmark localization result (%) for faces from Session 1 of Multi-PIE with large pose variations [$\pm 60^\circ$, $\pm 90^\circ$]. Results are mean error normalized by nose length

Method	3DDFA	PRNet	Deep3DFace	C-Net
NME	19.02	16.87	17.53	13.47

accurate correspondence between 2D face image and 3D model even for extreme poses.

4.2 Correspondence and texture inference

Previous methods build correspondences between a 2D facial image and a 3D model via low-dimensional 3D face reconstruction. We decompose the problem of 2D-to-3D correspondence building into two more manageable sub-tasks: face segmentation and PNCC prediction. We give a qualitative comparison of texture mapping from 2D images to 3D models between our method, PRNet [10], and Deep3DFace [11]. As shown in Fig. 5, the texture mapping results of PRNet and Deep3DFace contain non-facial areas that can neither be removed by their invisibility mask computed using standard Z-buffering [69, 70] nor image inpainting algorithms. Our C-Net, however, builds more accurate correspondences between input faces and 3D models, resulting in better texture inference.

4.3 Pose invariant face recognition with I-Net

To evaluate the effectiveness of I-Net, we apply it to pose invariant face recognition (PIFR) by projecting the reconstructed 3D texture faces into the image

plane with frontal view. We use the Celebrities Frontal-Profile (CFP) [36] and CMU Multi-PIE [63] datasets. Specifically, we fuse the feature vectors of probes and their frontalized versions extracted from a face network, and then compute the cosine distance between fused probe features and gallery features as the similarity measurement.

4.3.1 CFP dataset

CFP is a challenging dataset for PIFR algorithms, and here we focus on its most challenging task: frontal-profile matching. It contains 500 identities; each has ten frontal and four profile faces. The whole dataset has 7000 pairs for the task: 3500 same and 3500 differing pairs.

For the evaluation, we employed two advanced face networks, ResNet50 and ResNet50-Dream (ResNet50 with embedded Dream Block) proposed in Ref. [68] as our feature extractor. The naive ResNet50 model in Ref. [68] already achieved an accuracy of 97% on frontal-profile matching, which is higher than UV-GAN (94.05%) [3] and ArcFace (95.56%) [71].

In this section, we evaluate the face recognition performance on our two incorporated pipelines and other methods. As shown in Table 3, although the two face networks, ResNet50 and ResNet50-Dream both obtain impressive results on CFP in a frontal-profile setting, our SRTC-Net still improves the results considerably. Aided by the SRTC-Net, the

Fig. 5 Qualitative texture mapping comparison between our method, PRNet, and Deep3DFace, showing side and frontal views of textured 3D faces. PRNet and Deep3DFace both leave artifacts in the texture of visible vertices, which cannot be removed by post-processing algorithms.

Table 3 Evaluation of our algorithm using the CFP dataset with frontal profile setting. Lower equal error rate (EER) is better

Method	Training data	EER (%)
DR-GAN [14]	MS-Celeb-1M	14.26
TPE [65]	CASIA-WebFace	8.85
PIM [66]	MS-Celeb-1M	7.69
JB [67]	MS-Celeb-1M	8.49
TPE [65]	CASIA-WebFace	8.85
DREAM [68] End2end	MS-Celeb-1M	6.43
ResNet50	MS-Celeb-1M subset	7.85
ResNet50 + SRTC-Net	MS-Celeb-1M subset	7.02
ResNet50-Dream	MS-Celeb-1M subset	6.94
ResNet50-Dream + SRTC-Net	MS-Celeb-1M subset	6.39

ResNet50 model achieves similar performance to the Dream-Block-embedded ResNet50 model (ResNet50-Dream). Further, equipped with SRTC-Net, the ResNet50-Dream model trained on a small MS-Celeb-1M [73] subset fares better than when it is trained on the full MS-Celeb-1M. Some examples of frontalized faces from the CFP dataset are showcased in Fig. 6. These results demonstrate that our method can infer complete facial textures robustly despite significant variations in pose, expression, and illumination in an unconstrained environment.

4.3.2 Multi-PIE database

The CMU Multi-PIE [63] dataset is the largest controlled multiview face recognition database. It was collected in four sessions; for evaluation, we use Session 1, which includes faces of 250 subjects. Following the two most popular test protocols in Refs. [30, 74], we select images of the last 100 subjects for our evaluation. The gallery set comprises a frontal view face image for each subject under normal illumination. For probe sets, we only consider challenging faces whose poses are larger than 45° .

Fig. 6 Frontalization results on the CFP dataset. In each triple, left-to-right: face profile, frontalized face, actual frontal face from CFP.

Altogether, there are 100 galleries and 16,000 probe images.

We employ the pre-trained Light CNNv2 model [75] as the face feature extractor. As shown in Table 4, although this powerful model achieves nearly perfect accuracy for non-extreme poses, our method still improves upon it for all poses; the accuracy increases by about 10% for full side views. We also exhibit face frontalization results for images from the Multi-PIE dataset in Fig. 7.

Table 4 Rank-1 recognition rate (%) for varying views and illumination for Multi-PIE [63] Setting-1

Method	$\pm 90^\circ$	$\pm 75^\circ$	$\pm 60^\circ$	$\pm 45^\circ$
HPN [76]	29.82	47.57	61.24	72.77
c-CNN [77]	47.26	60.7	74.4	89.0
PIM [66]	75.0	91.2	97.7	98.3
TP-GAN [24]	64.0	84.1	92.9	98.6
CAPG-GAN [78]	77.1	87.4	93.7	98.3
FNM [38]	55.8	81.3	93.7	98.2
Light CNNv2 [75]	54.7	89.5	98.65	99.77
[75]+SRTC-Net	63.2	91.1	99.2	99.92

4.4 Evaluation of S-Net

To validate the efficacy of our S-Net, we assess it on the MICC Florence 3D Face dataset [79]. MICC contains 53 subjects, each of which has a 3D scanned model and three videos captured in cooperative, indoor, and outdoor scenarios. We compare our method with several state-of-the-art image-based 3D face reconstruction methods [10, 11, 47, 72] in indoor scenarios. We crop the ground-truth face scans to 100 mm around the nose tip and remove the forehead region of the scans following Ref. [11]. In this experiment, we extracted an image every ten frames from each video, and used a total of 7895 frames from videos of 53 subjects.

For 3DDFA^①, VRN^②, PRNet^③, Deep3DFace^④, and our S-Net, we reconstructed an average shape from all frames of one subject to evaluate the error. To compute error distances, we used a publicly available evaluation tool^⑤. Specifically, we employ 7 points:

^① <https://github.com/cleardusk/3DDFA>

^② <https://github.com/AaronJackson/vrn>

^③ <https://github.com/YadiraF/PRNet>

^④ <https://github.com/microsoft/Deep3DFaceReconstruction>

^⑤ <https://github.com/patrikhuber/fg2018-competition>

Fig. 7 Example frontalization results on the Multi-PIE dataset. Each triple, left-to-right: original face image, frontalized face, ground-truth frontal face.

the inner and outer eye corners of both eyes, the nose-tip, and the mouth corners to rigidly align the predicted mesh with the ground truth. The distance error for each ground-truth vertex is the distance between the vertex and the closest point on the surface of the predicted mesh. For comparison, we use root mean square error (RMSE) to measure the difference between the predicted mesh and the ground-truth mesh. Table 5 shows that our reconstruction performance is close to Ref. [11] and better than Refs. [10, 47, 72]. Various qualitative results on the MICC dataset are shown in Fig. 8. Additionally, we display a qualitative comparison for some in-the-wild images in Fig. 9, which shows that our method provides better geometry fitting and facial identity detail preservation. To make our framework computationally efficient, we restrict the resolution

of the texture map to 84×84 , which may limit reconstruction detail. Thus, we only compare our method to low-dimensional methods to show that our completed U - V texture can provide detail structures beyond 3DMM methods.

5 Conclusions and limitations

This paper has proposed the deep SRTC-Net to attack the challenging problem of inferring 3D facial geometry and texture from a single image. Our method consists of three subnetworks, the C-Net, I-Net, and S-Net, which decompose the hard problem into several more tractable problems. The C-Net builds accurate correspondences between 2D face images and 3D template models with two pixel-wise prediction subnetworks. Our carefully designed I-Net completes missing areas of U - V texture maps. Our reconstructed U - V texture maps are applied in the S-Net to extract detailed surface structures. Specifically, the S-Net reconstructs a coarse shape (U - V position map) first, and then regresses 3D surface

Table 5 Mean RMSE (in mm) for the MICC dataset, for various methods

VRN	3DDFA	PRNet	Deep3DFace	Ours
2.53	2.20	2.32	1.91	2.09

Fig. 8 Comparison of 3D reconstruction examples from the MICC dataset. Left to right: example input frame image and ground truth 3D face, and 3D estimation results for VRN, 3DDFA, PRNet, Deep3DFace, and our S-Net.

Fig. 9 Qualitative comparison on unconstrained images between our method and several state-of-the-art methods: VRN, 3DDFA, PRNet, and Deep3DFace.

details (U - V displacement map) from the complete U - V texture maps pixel-by-pixel. A comparison between our C-Net and state-of-the-art face reconstruction methods on the Multi-PIE dataset demonstrates the effectiveness of C-Net. For 3D facial geometry reconstruction, we get better or close quantitative performance to state-of-the-art deep learning based methods on the MICC dataset. Furthermore, our methods achieve better qualitative results than previous low-dimensional methods on unconstrained images. Our face recognition results on the CFP and Multi-PIE datasets demonstrate that our method can also significantly promote the performance of face recognition models on the PIFR task. Our framework is trained separately, which makes our method a little complicated. Although symmetry is reasonable for a facial texture map, it may not be appropriate for some facial details: e.g., a crooked smile can break the symmetry of facial texture. The cascading structure of our whole framework means that error in each sub-module is transmitted to next stage. Our work cannot provide accurate PNCC of face areas occluded by non-facial objects, such as a hat. In future, we hope to make our framework more easily trainable and more robust to extreme expressions.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Nos. U1613211 and U1813218), and Shenzhen Research Program (Nos. JCYJ20170818164704758 and JCYJ20150925163005055). We would like to thank Yu Deng et al. for their Deep 3D Face work in 3D face analysis, whose contribution to this field permitted our further study.

References

- [1] Ichim, A. E.; Bouaziz, S.; Pauly, M. Dynamic 3D avatar creation from hand-held video input. *ACM Transactions on Graphics* Vol. 34, No. 4, Article No. 45, 2015.
- [2] Thies, J.; Zollhöfer, M.; Stamminger, M.; Theobalt, C.; Nießner, M. Face2Face: Real-time face capture and reenactment of RGB videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2387–2395, 2016.
- [3] Deng, J.; Cheng, S.; Xue, N.; Zhou, Y.; Zafeiriou, S. UV-GAN: Adversarial facial UV map completion for pose-invariant face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 7093–7102, 2018.
- [4] Zhao, J.; Xiong, L.; Cheng, Y.; Cheng, Y.; Li, J.; Zhou, L.; Xu, Y.; Karlekar, J.; Pranata, S.; Shen, S. et al. 3D-aided deep pose-invariant face recognition. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, 1184–1190, 2018.
- [5] Bessaoudi, M.; Belahcene, M.; Ouamane, A.; Chouchane, A.; Bourennane, S. Multilinear Enhanced Fisher Discriminant Analysis for robust multimodal 2D and 3D face verification. *Applied Intelligence* Vol. 49, No. 4, 1339–1354, 2019.
- [6] Devi, P. R. S.; Baskaran, R. SL2E-AFRE: Personalized 3D face reconstruction using autoencoder with simultaneous subspace learning and landmark estimation. *Applied Intelligence* Vol. 51, No. 4, 2253–2268, 2021.
- [7] Yuan, Z. H.; Lu, T. Incremental 3D reconstruction using Bayesian learning. *Applied Intelligence* Vol. 39, No. 4, 761–771, 2013.
- [8] Alexander, O.; Rogers, M.; Lambeth, W.; Chiang, J. Y.; Ma, W. C.; Wang, C. C.; Debevec, P. The Digital Emily project: Achieving a photorealistic digital actor. *IEEE Computer Graphics and Applications* Vol. 30, No. 4, 20–31, 2010.
- [9] Jackson, A. S.; Bulat, A.; Argyriou, V.; Tzimiropoulos, G. Large pose 3D face reconstruction from a single image via direct volumetric CNN regression. In: Proceedings of the IEEE International Conference on Computer Vision, 1031–1039, 2017.
- [10] Feng, Y.; Wu, F.; Shao, X. H.; Wang, Y. F.; Zhou, X. Joint 3D face reconstruction and dense alignment with position map regression network. In: *Computer Vision—ECCV 2018. Lecture Notes in Computer Science, Vol. 11218*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 557–574, 2018.
- [11] Deng, Y.; Yang, J. L.; Xu, S. C.; Chen, D.; Jia, Y. D.; Tong, X. Accurate 3D face reconstruction with weakly-supervised learning: From single image to image set. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 285–295, 2019.
- [12] Blanz, V.; Vetter, T. A morphable model for the synthesis of 3D faces. In: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, 187–194, 1999.
- [13] Tran, A. T.; Hassner, T.; Masi, I.; Medioni, G. Regressing robust and discriminative 3D morphable models with a very deep neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1493–1502, 2017.

- [14] Tran, L.; Yin, X.; Liu, X. M. Disentangled representation learning GAN for pose-invariant face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1283–1292, 2017.
- [15] Bas, A.; Huber, P.; Smith, W. A. P.; Awais, M.; Kittler, J. 3D morphable models as spatial transformer networks. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, 895–903, 2017.
- [16] Paysan, P.; Knothe, R.; Amberg, B.; Romdhani, S.; Vetter, T. A 3D face model for pose and illumination invariant face recognition. In: Proceedings of the 6th IEEE International Conference on Advanced Video and Signal Based Surveillance, 296–301, 2009.
- [17] Sela, M. T.; Richardson, E.; Kimmel, R. Unrestricted facial geometry reconstruction using image-to-image translation. In: Proceedings of the IEEE International Conference on Computer, 1585–1594, 2017.
- [18] Li, Y.; Ma, L. Q.; Fan, H. Q.; Mitchell, K. Feature-preserving detailed 3D face reconstruction from a single image. In: Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production, Article No. 1, 2018.
- [19] Richardson, E.; Sela, M. T.; Or-El, R.; Kimmel, R. Learning detailed face reconstruction from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5553–5562, 2017.
- [20] Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In: *Medical Image Computing and Computer-Assisted Intervention. Lecture Notes in Computer Science, Vol. 9351*. Navab, N.; Hornegger, J.; Wells, W.; Frangi, A. Eds. Springer Cham, 234–241, 2015.
- [21] Shen, Y. J.; Luo, P.; Luo, P.; Yan, J. J.; Wang, X. G.; Tang, X. O. FaceID-GAN: Learning a symmetry three-player GAN for identity-preserving face synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 821–830, 2018.
- [22] Yin, X.; Yu, X.; Sohn, K.; Liu, X. M.; Chandraker, M. Towards large-pose face frontalization in the wild. In: Proceedings of the IEEE International Conference on Computer Vision, 4010–4019, 2017.
- [23] Zhou, Y. X.; Deng, J. K.; Kotsia, I.; Zafeiriou, S. Dense 3D face decoding over 2500FPS: Joint texture & shape convolutional mesh decoders. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1097–1106, 2019.
- [24] Huang, R.; Zhang, S.; Li, T.; He, R. Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis. In: Proceedings of the IEEE International Conference on Computer Vision, 2458–2467, 2017.
- [25] Gecer, B.; Ploumpis, S.; Kotsia, I.; Zafeiriou, S. GANFIT: Generative adversarial network fitting for high fidelity 3D face reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1155–1164, 2019.
- [26] Hassner, T.; Harel, S.; Paz, E.; Enbar, R. Effective face frontalization in unconstrained images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4295–4304, 2015.
- [27] Chang, F. J.; Yu, X.; Nevatia, R.; Chandraker, M. Pose-variant 3D facial attribute generation. *arXiv preprint arXiv:1907.10202*, 2019.
- [28] Xue, N. N.; Deng, J. K.; Cheng, S. Y.; Panagakis, Y.; Zafeiriou, S. Side information for face completion: A robust PCA approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 41, No. 10, 2349–2364, 2019.
- [29] Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, 2672–2680, 2014.
- [30] Athana, A.; Marks, T. K.; Jones, M. J.; Tieu, K. H.; Rohith, M. Fully automatic pose-invariant face recognition via 3D pose normalization. In: Proceedings of the International Conference on Computer Vision, 937–944, 2011.
- [31] Cootes, T. F.; Walker, K.; Taylor, C. J. View-based active appearance models. In: Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition, 227–232, 2000.
- [32] Wei, L.-Y.; Lefebvre, S.; Kwatra, V.; Turk, G. State of the art in example-based texture synthesis. In: Proceedings of the Eurographics 2009, State of the Art Reports (STARs), 93–117, 2009.
- [33] Efros, A. A.; Leung, T. K. Texture synthesis by non-parametric sampling. In: Proceedings of the 7th IEEE International Conference on Computer Vision, 1033–1038, 1999.
- [34] Li, Y. J.; Liu, S. F.; Yang, J. M.; Yang, M. H. Generative face completion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5892–5900, 2017.
- [35] Wen, Y. D.; Zhang, K. P.; Li, Z. F.; Qiao, Y. A discriminative feature learning approach for deep face recognition. In: *Computer Vision—ECCV 2016. Lecture*

- Notes in Computer Science*, Vol. 9911. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 499–515, 2016.
- [36] Sengupta, S.; Chen, J. C.; Castillo, C.; Patel, V. M.; Chellappa, R.; Jacobs, D. W. Frontal to profile face verification in the wild. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 1–9, 2016.
- [37] Tran, L.; Yin, X.; Liu, X. Representation learning by rotating your faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 41, No. 12, 3007–3021, 2018.
- [38] Qian, Y. C.; Deng, W. H.; Hu, J. N. Unsupervised face normalization with extreme pose and expression in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9843–9850, 2019.
- [39] Guo, J. Z.; Zhu, X. Y.; Yang, Y.; Yang, F.; Lei, Z.; Li, S. Z. Towards fast, accurate and stable 3D dense face alignment. In: *Computer Vision–ECCV 2020. Lecture Notes in Computer Science*, Vol. 12364. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 152–168, 2020.
- [40] Sanyal, S.; Bolkart, T.; Feng, H. W.; Black, M. J. Learning to regress 3D face shape and expression from an image without 3D supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7755–7764, 2019.
- [41] Tran, L.; Liu, X. M. Nonlinear 3D face morphable model. *arXiv preprint* arXiv: 1804.03786, 2018.
- [42] Tewari, A.; Zollhöfer, M.; Garrido, P.; Bernard, F.; Kim, H.; Perez, P.; Theobalt, C. Self-supervised multi-level face model learning for monocular reconstruction at over 250 Hz. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2549–2559, 2018.
- [43] Wu, S. Z.; Rupperecht, C.; Vedaldi, A. Unsupervised learning of probably symmetric deformable 3D objects from images in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1–10, 2020.
- [44] Lin, J. K.; Yuan, Y.; Shao, T. J.; Zhou, K. Towards high-fidelity 3D face reconstruction from in-the-wild images using graph convolutional networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5890–5899, 2020.
- [45] Lee, G. H.; Lee, S. W. Uncertainty-aware mesh decoder for high fidelity 3D face reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6099–6108, 2020.
- [46] Lattas, A.; Moschoglou, S.; Gecer, B.; Ploumpis, S.; Triantafyllou, V.; Ghosh, A.; Zafeiriou, S. AvatarMe: Realistically renderable 3D facial reconstruction “in-the-wild”. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 757–766, 2020.
- [47] Zhu, X. Y.; Liu, X. M.; Lei, Z.; Li, S. Z. Face alignment in full pose range: A 3D total solution. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 41, No. 1, 78–92, 2019.
- [48] Zhang, K. P.; Zhang, Z. P.; Li, Z. F.; Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters* Vol. 23, No. 10, 1499–1503, 2016.
- [49] Iizuka, S.; Simo-Serra, E.; Ishikawa, H. Globally and locally consistent image completion. *ACM Transactions on Graphics* Vol. 36, No. 4, Article No. 107, 2017.
- [50] Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv preprint* arXiv:1511.07122, 2015.
- [51] Karras, T.; Laine, S.; Aila, T. M. A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4396–4405, 2019.
- [52] Parkhi, O. M.; Vedaldi, A.; Zisserman, A. Deep face recognition. In: Proceedings of the British Machine Vision Conference, Article No. 41, 2015.
- [53] Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint* arXiv:1409.1556, 2014.
- [54] Mescheder, L.; Geiger, A.; Nowozin, S. Which training methods for GANs do actually converge? *arXiv preprint* arXiv:1801.04406, 2018.
- [55] Ma, L. M.; Deng, Z. G. Real-time hierarchical facial performance capture. In: Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, Article No. 11, 2019.
- [56] Tran, L.; Liu, F.; Liu, X. M. Towards high-fidelity nonlinear 3D face morphable model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1126–1135, 2019.
- [57] Tewari, A.; Zollhoefer, M.; Bernard, F.; Garrido, P.; Kim, H.; Perez, P.; Theobalt, C. High-fidelity monocular face reconstruction based on an unsupervised model-based face autoencoder. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 42, No. 2, 357–370, 2018.
- [58] Sengupta, S.; Kanazawa, A.; Castillo, C. D.; Jacobs, D. W. SFSNet: Learning shape, reflectance and illuminance of faces ‘in the wild’. In: Proceedings of

- the IEEE Conference on Computer Vision and Pattern Recognition, 6296–6305, 2018.
- [59] Yin, L. J.; Chen, X. C.; Sun, Y.; Worm, T.; Reale, M. A high-resolution 3D dynamic facial expression database. In: Proceedings of the 8th IEEE International Conference on Automatic Face & Gesture Recognition, 1–6, 2008.
- [60] Savran, A.; Alyüz, N.; Dibeklioglu, H.; Çeliktutan, O.; Gökberk, B.; Sankur, B.; Akarun, L. Bosphorus database for 3D face analysis. In: *Biometrics and Identity Management. Lecture Notes in Computer Science, Vol. 5372*. Schouten, B.; Juul, N. C.; Drygajlo, A.; Tistarelli, M. Eds. Springer Berlin Heidelberg, 47–56, 2008.
- [61] White, J. D.; Ortega-Castrillón, A.; Matthews, H.; Zaidi, A. A.; Ekrami, O.; Snyders, J.; Fan, Y.; Penington, T.; Van Dongen, S.; Shriver, M. D. MeshMonk: Open-source large-scale intensive 3D phenotyping. *Scientific Reports* Vol. 9, No. 1, 6085, 2019.
- [62] Hormann, K.; Lévy, B.; Sheffer, A. Mesh parameterization: Theory and practice. In: Proceedings of the ACM SIGGRAPH 2007 Courses, 1–es, 2007.
- [63] Gross, R.; Matthews, I.; Cohn, J.; Kanade, T.; Baker, S. Multi-PIE. *Image and Vision Computing* Vol. 28, No. 5, 807–813, 2010.
- [64] Baltrusaitis, T.; Zadeh, A.; Lim, Y. C.; Morency, L. P. OpenFace 2.0: Facial behavior analysis toolkit. In: Proceedings of the 13th IEEE International Conference on Automatic Face & Gesture Recognition, 59–66, 2018.
- [65] Sankaranarayanan, S.; Alavi, A.; Castillo, C. D.; Chellappa, R. Triplet probabilistic embedding for face verification and clustering. *arXiv preprint arXiv:1604.05417*, 2016.
- [66] Zhao, J.; Cheng, Y.; Xu, Y.; Xiong, L.; Li, J. S.; Zhao, F.; Jayashree, K.; Pranata, S.; Shen, S.; Xing, J. L. Towards pose invariant face recognition in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2207–2216, 2018.
- [67] Chen, D.; Cao, X.; Wang, L.; Wen, F.; Sun, J. Bayesian face revisited: A joint formulation. In: Proceedings of the 12th European Conference on Computer Vision, 566–579, 2012.
- [68] Cao, K. D.; Rong, Y.; Li, C.; Tang, X. O.; Loy, C. C. Pose-robust face recognition via deep residual equivariant mapping. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5187–5196, 2018.
- [69] Catmull, E. E. A subdivision algorithm for computer display of curved surfaces. Technical Report, CSTD-74-006. The University of Utah, 1974.
- [70] Hughes, J. F.; van Dam, A.; Foley, J. D.; McGuire, M.; Feiner, S. K.; Akeley, K.; Sklar, D. F. *Computer Graphics: Principles and Practice*, 3rd edn. Addison-Wesley, 2014.
- [71] Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. ArcFace: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4690–4699, 2019.
- [72] Jackson, A. S.; Bulat, A.; Argyriou, V.; Tzimiropoulos, G. Large pose 3D face reconstruction from a single image via direct volumetric CNN regression. In: Proceedings of the IEEE International Conference on Computer Vision, 1031–1039, 2017.
- [73] Guo, Y. D.; Zhang, L.; Hu, Y. X.; He, X. D.; Gao, J. F. MS-celeb-1M: A dataset and benchmark for large-scale face recognition. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9907*. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 87–102, 2016.
- [74] Li, A. N.; Shan, S. G.; Gao, W. Coupled bias-variance tradeoff for cross-pose face recognition. *IEEE Transactions on Image Processing* Vol. 21, No. 1, 305–315, 2012.
- [75] Wu, X.; He, R.; Sun, Z. N.; Tan, T. N. A light CNN for deep face representation with noisy labels. *IEEE Transactions on Information Forensics and Security* Vol. 13, No. 11, 2884–2896, 2018.
- [76] Ding, C. X.; Tao, D. C. Pose-invariant face recognition with homography-based normalization. *Pattern Recognition* Vol. 66, 144–152, 2017.
- [77] Xiong, C.; Zhao, X. W.; Tang, D. H.; Jayashree, K.; Yan, S. C.; Kim, T. K. Conditional convolutional neural network for modality-aware face recognition. In: Proceedings of the IEEE International Conference on Computer Vision, 3667–3675, 2015.
- [78] Hu, Y. B.; Wu, X.; Yu, B.; He, R.; Sun, Z. N. Pose-guided photorealistic face rotation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8398–8406, 2018.
- [79] Bagdanov, A. D.; del Bimbo, A.; Masi, I. The Florence 2D/3D hybrid face dataset. In: Proceedings of the Joint ACM Workshop on Human Gesture and Behavior Understanding, 79–80, 2011.

Xiaoxing Zeng is now a Ph.D. student at Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, majoring in computer applied technology. His research interests include computer vision and deep learning.

Zhelun Wu is currently visiting at Shenzhen Institute of Advanced Technology. He received his master degree from Princeton University in 2018 and B.Eng. degree from Tsinghua University in 2016. His research interests lie in computer vision and deep learning.

Xiaojiang Peng received his Ph.D. degree from School of Information Science and Technology, Southwest Jiaotong University, China, in 2014. He currently is an associate professor at the Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, China. He was a postdoctoral researcher

at Idiap Institute, Switzerland, from 2016 to 2017, and was a postdoctoral researcher in LEAR Team, INRIA, France, working with Prof. Cordelia Schmid from 2015 to 2016. He serves as a reviewer for IJCV, TMM, TIP, CVPR, ICCV, AAAI, IJCAI, FG, *Image and Vision Computing*, *IEEE Signal Processing Letter*, *Neurocomputing*, etc. His research focus is in the areas of action recognition and detection, face recognition, facial emotion analysis, and deep learning.

Yu Qiao received his Ph.D. degree from the University of Electro-Communications, Japan, in 2006. He was a JSPS Fellow and a Project Assistant Professor with the University of Tokyo, from 2007 to 2010. He is currently a professor with the Shenzhen Institute of Advanced Technology, Chinese

Academy of Sciences. He has authored over 140 papers in journals and conferences including PAMI, IJCV, TIP, ICCV, CVPR, ECCV, and AAAI. His research interests include computer vision, deep learning, and intelligent robots. He was a recipient of the Lu Jiaxi Young Researcher Award from the Chinese Academy of Sciences in 2012. He was the first runner-up at the ImageNet Large Scale Visual Recognition Challenge 2015 in scene recognition and the recipient at the ActivityNet Large Scale Activity Recognition Challenge 2016 in video classification.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made.

The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Other papers from this open access journal are available free of charge from <http://www.springer.com/journal/41095>. To submit a manuscript, please go to <https://www.editorialmanager.com/cvmj>.