Single-Image 3D Face Reconstruction under Perspective Projection

Yueying Kao[1], Bowen Pan[1], Miao Xu[2], Jiangjing Lyu[1], Xiangyu Zhu[2⋆], Yuanzhang Chang[1], Xiaobo Li[1], Zhen Lei[2], and Zixiong Qin[3]

[1] Alibaba Group
[2] CBSR & NLPR, Institute of Automation, Chinese Academy of Sciences
[3] Speechocean

Abstract. In 3D face reconstruction, orthogonal projection has been widely employed to substitute perspective projection to simplify the fitting process. This approximation performs well when the distance between camera and face is far enough. However, in some scenarios that the face is very close to camera or moving along the camera axis, the methods suffer from the inaccurate reconstruction and unstable temporal fitting due to the distortion under the perspective projection. In this paper, we aim to address the problem of single-image 3D face reconstruction under perspective projection. Specifically, a deep neural network, Perspective Network (PerspNet), is proposed to simultaneously reconstruct 3D face shape in canonical space and learn the correspondence between 2D pixels and 3D points, by which the 6DoF (6 Degrees of Freedom) face pose can be estimated to represent perspective projection. Besides, we contribute a large ARKitFace dataset to enable the training and evaluation of 3D face reconstruction solutions under the scenarios of perspective projection, which has 902,724 2D facial images with ground-truth 3D face mesh and annotated 6DoF pose parameters. Experimental results show that our approach outperforms current state-of-the-art methods by a significant margin.

Keywords: 3D face reconstruction, perspective projection, 6DoF pose estimation

1 Introduction

3D face reconstruction [29,13,3,12,42] has draw much attention recently in computer vision and computer graphics communities, due to the increasing demand from many applications, such as virtual glasses try-on and make-up in AR, video editing and animation. Most of 3D face reconstruction methods [41,13,3,12,16] employ the orthogonal projection [9,15] to approximate the real-world perspective projection, which works well when the size of face is small compared to the distance from the camera (roughly 1/20 of the camera distance). However,

⋆ Corresponding author.

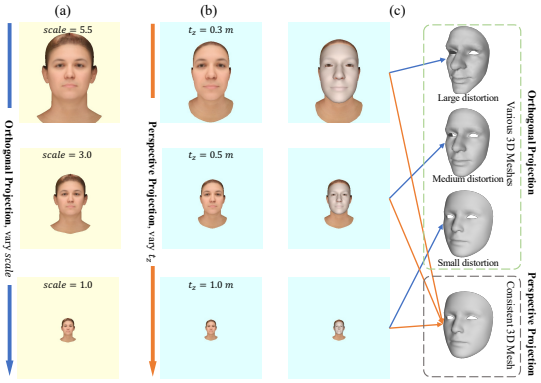

Fig. 1. Orthogonal projection vs. Perspective projection. (a) and (b) show rendered images with a same 3D face by changing the pose parameters that represent the size in the two projections, respectively. The rendered 2D faces are only zoomed in and out with scale variation in orthogonal projection, while in perspective projection there exists obvious distortion by changing t_z, especially in very near distance. (c) shows 3D face reconstruction under the two projections respectively. Orthogonal projection based methods explain the distortion by the shape changes, while perspective projection methods provide the same shape and explain it by different pose parameters.

the scenarios of face capture become more complicated with the popular of selfies, virtual glasses try-on and makeup, etc. When the subject is very close to the camera, the rendered 2D faces in orthogonal projection are only zoomed in, while in perspective projection there exists obvious distortion in the rendered 2D faces, especially in the very close distance, as shown in Fig. 1(a) and (b). Under the approximation of orthogonal projection, the distortion by perspective projection is explained by the shape changes, leading to two significant problems: 1) This distortion is not modeled in the shape models, and the distorted faces are often outside of the shape space, which leads to the unstable temporal fitting. 2) When the subject moves along the camera axis t_z, the orthogonal projection based methods predict different face shapes in different distances due to the distortion, while perspective projection methods provide the consistent shape across frames since it explains the distortions with different 6DoF pose, as shown in Fig. 1(c). Even introducing 6DoF pose apparently improves the accuracy and robustness of 3D face reconstruction, which benefits many VR/AR applications, 6DoF pose estimation for faces is still a challenging problem. Since we should capture the distortion by perspective projection from the large variations of face appearances under complicated environment. Besides, 6DoF pose estimation in other objects [26,18,33] always assumes a pre-defined 3D shape but we only have a face image as input.

To address the problem, we propose a new approach to recover 3D facial geometry under perspective projection by estimating the 6DoF pose, i.e. 3D orientation and translation, simultaneously from a single RGB image (see Sec-

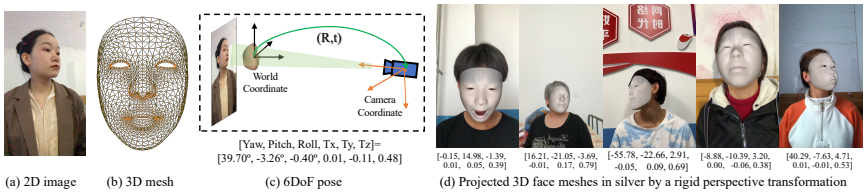

Fig. 2. Some examples of ARKitFace dataset. Each sample contains (a) 2D image, (b) 3D mesh and (c) 6DoF pose annotation. (d) shows more examples.

tion 3). Specifically, 6DoF estimation depends on 3 sub-tasks: 1) Reconstructing 3D face shape in canonical space, 2) Estimating pixel-wise 2D-3D correspondence between the canonical 3D space and image space and 3) Calculating 6DoF pose parameters by the Perspective-n-Point (PnP) algorithm [23]. In this paper, we propose a deep learning based Perspective Network (PerspNet) to achieve the goal in one propagation. For the 3D face shape reconstruction, an encoder-decoder architecture is designed to regress a UV position map [13] from image pixels, which records the specific position information of a 3D face in canonical space. For the 2D-3D correspondence, we construct a sophisticated point description features including aligned-pixel features, 3D features and 2D position features, and map it to a 2D-3D correspondence matrix where each row records the corresponding 3D points in canonical space of one pixel on the input image. With 3D shape and correspondence matrix, 6DoF face pose can be robustly estimated by performing PnP.

To realize 6DoF pose estimation of 3D faces, we need the face images, 3D shapes and annotated camera parameters for each training sample. However, current datasets cannot satisfy our requirement. Most of existing data provide the low-quality 3D face by the optimization-based fitting algorithm in a weak supervised manner, such as AFLW2000-3D [41]. Some datasets, such as BIWI [11], NoW [31], FaceScape [37], lack either exact 3D shape for each 2D image or pose variations. Therefore, we construct a real large 3D face dataset called ARKitFace dataset, with 902,724 2D facial images of 500 subjects taken in diverse of expressions, ages and poses. For each 2D facial image, ground-truth 3D face mesh and 6DoF pose annotations are provided, as shown in Fig. 2. Compared with other 3D face datasets [41,31,11,37], the ARKitFace is a very large amount dataset for single-image 3D face reconstruction under perspective projection.

To summarize, the main contributions of this work are:

- We explore a new problem of 3D face reconstruction under perspective projection, which will provide the exact 3D face shape and location in 3D camera space.
- We propose PerspNet to reconstruct 3D face shape and 2D-3D correspondence simultaneously, by which the 6DoF face pose can be estimated by PnP.

- To enable the training and evaluation of PerspNet, we collect ARKitFace dataset, a large-scale 3D dataset with ground-truth 3D face mesh and 6DoF pose annotations for each image.
- Experimental results on ARKitFace dataset and a public BIWI dataset show that our approach outperforms the state-of-the-art methods. The code and all data will be released to public under the authorization of all the subjects[†].

2 Related Work

2.1 3D Face Reconstruction

3D face reconstruction from a single RGB image is essentially an ill-posed problem. Most methods tackle this problem by estimating the parameters of a statistical face model [16,3,41,13,12,29,42]. Although these methods achieve remarkable results, they suffer from the same fundamental limitation: an orthogonal or weak perspective camera model is utilized when reconstructing the 3D face shape. The deformation caused by perspective projection, especially in the near distance, has to be compensated by face shape. Thus, we insist on following the rule of perspective projection and believe there is still substantial room for improvement on the task of 3D face reconstruction.

2.2 Head/Face Pose Estimation

Due to the prevalence of deep learning, great progresses have been achieved in head pose estimation. The main idea is to regress the euler angles of head pose directly based on deep CNNs. QuatNet [20] addresses the non-stationary property of head pose and proposed to train a quaternions regressor to avoid the ambiguity problem in euler angles. FSA-Net [38] adopts a fine-grained structure mapping for spatially feature grouping to improve the performance of head pose estimation. EVA-GCN [36] views the head pose estimation as a graph regression problem, and leverages the Graph Convolutional Networks to model the complex nonlinear mappings between the graph typologies and the head pose angles. All of the above methods are proposed under the assumption of orthogonal projection, thus only 3DoF (3D rotation) is predicted and the error caused by perspective deformation can not be avoided.

Laterly, Chang *et al.* [8] regress directly the 6DoF parameters of human faces under perspective projection from a face photo. Albiero *et al.* [1] propose a Faster-RCNN [28] like framework named img2pose to predict the full 6DoF of human faces in the setting of perspective projection. Firstly, the original full image is fed into the RPN module, and all faces in the image are detected. Then the local 6DoF of each detected faces is predicted by the ROI head. Finally the local 6DoF is converted to global 6DoF using the information of facial bounding box and camera intrinsic matrix. Since img2pose ignores the influence of perspective deformation, the error of 6DoF will be amplified during local-to-global

[†] code and data will be at *www.to-be-released.com*.

conversion. In addition, the 6DoF pose annotations in their training data are calculated with predicted five landmarks and a mean 3D mesh, not real 6DoF pose.

2.3 3D Face Datasets

Although a large number of facial images are available, the corresponding 3D annotations are expensive and difficult to obtain. For 3D face shape reconstruction, several 3D datasets are built, such as Bosphorus [32], BFM [25], FaceWarehouse [6], MICC [2], 3DFAW [27], BP4D [39], NoW dataset [31]. In these datasets, there are either limited data or no face pose annotation.

To obtain head/face 6DoF pose, some datasets synthesize 3D ground-truth, i.e. the parameters of a statistical face model, by the optimization-based fitting algorithm in a weak supervised manner, such as 300W-LP and AFLW2000-3D [41]. The synthesized 3D ground-truth is coarse because only the reconstruction error of 2D sparse facial landmarks is considered. Despite the expensiveness of 3D annotations, researchers have collected several 3D face datasets using professional imaging devices for the sake of high precision on the tasks of 3D face reconstruction and head pose estimation. For example, BIWI dataset [11] is captured by a Kinect sensor with global rotation and translation to the RGB camera. However, the number of individuals is limited, and facial images with only neutral expression are recorded. Most importantly, the size of BIWI dataset is too small for learning deep networks. FaceScape [37] is a large-scale dataset recorded by multi-view camera system in an extremely constrained environment. It contains sufficient individuals and multiple facial expressions. However, the head pose is fixed by limited number of camera locations and the lighting condition is constant. Models that are trained on such dataset do not have good generalization ability to the real scenario in the wild. Different from these datasets, we aim to collect a 3D dataset with 3D mesh and 6DoF pose estimation in different conditions, such as expression, age and 6D pose variations.

3 Proposed Method

In this paper, we propose a novel framework for 3D face reconstruction under perspective projection from a single 2D face image. In previous single-image 3D face reconstruction method [41,13,3,12,16], scaled orthographic projection camera model is adopted to project 3D face shape into image space. We denote the 3D face shape as $X \in \mathbb{R}^{3\times n}$ representing n 3D vertices (points) on the surface of the 3D shape in world coordinate system (canonical space). This projection process is usually formulated as

$$V = s\Pi(X) + t_{2d}, \tag{1}$$

where $V \in \mathbb{R}^{2\times n}$ denotes projected 2D coordinates in 2D image of X, $\Pi \in \mathbb{R}^{2\times 3}$ is the orthographic 3D-2D projection matrix, s is isotropic scale and $t_{2d} \in$

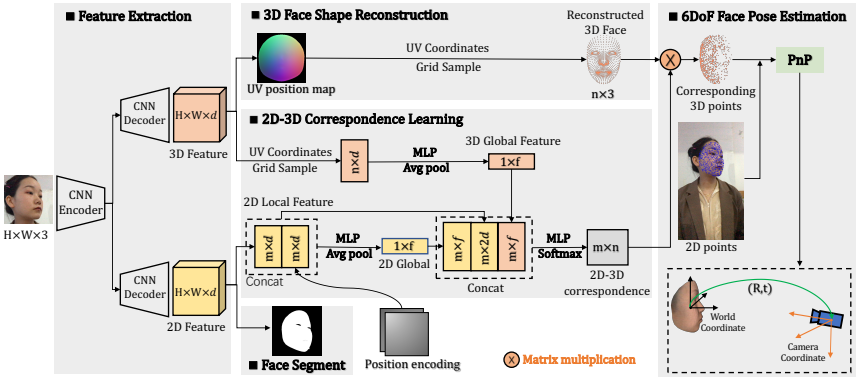

Fig. 3. The framework of our proposed method based on PerspNet. 3D features and 2D image features are extracted from encoder-decoder architectures respectively from a 2D facial image. The 3D features are fed into 3D face shape reconstruction module to predict 3D face shape information in world coordinate system. The 2D facial image features, 2D position encoding features and 3D features are fused learn the correspondence between 2D pixels and 3D points in reconstructed 3D face shape. With the corresponding 2D pixels and 3D points in a face, the 6DoF pose of the face can be computed by a PnP algorithm. In addition, 2D image features are also fed into 2D face segmentation, which is used to extract 2D pixels in face regions when testing.

$\mathbb{R}^2$ denotes 2D translation. Different from orthogonal projection, in perspective projection, 3D face shape X is firstly transformed from the world coordinate system to the camera coordinate system by using 6DoF face pose (R, t),

$$X_c = K(RX + t), \tag{2}$$

with known intrinsic camera parameters K, where R, t and X_c represent the 3D rotation $R \in SO(3)$, the 3D translation $t \in \mathbb{R}^3$, and 3D face vertices in camera coordinate system. Then the X_c is projected to image space by $V = X_c[0:2,:]/Z$, where $Z = X_c[2,:]$ represents the distance from each vertex to camera. This illustrates that the distance Z mainly leads to the difference of the two projections.

In this work, we focus on 3D face reconstruction under perspective projection, especially in very near distance. Given an RGB image $I \in \mathbb{R}^{H\times W\times 3}$, the goal is to find a method F to recover the 3D face shape X and estimate its 6DoF pose (R, t): $[X, R, t] = F(I)$. It is achieved by a framework based on a new deep learning network, PerspNet, as shown in Fig. 3. The proposed method contains two sub-tasks: 3D face shape reconstruction and 6DoF face pose estimation. For the 3D face shape reconstruction, we design a UV position map [13] regression method. For 6DoF face pose estimation, we use a two-stage pipeline [26,18,33], that first chooses m 2D pixels $V \in \mathbb{R}^{2\times m}$ in 2D image and learns their corresponding 3D points in reconstructed 3D face shape and 6D face pose parameters can be calculated by a PnP [23] algorithm.

3.1 Perspective Network (PerspNet)

Specifically, PerspNet consists of four modules: feature extraction, 3D face shape X reconstruction, 2D-3D correspondence M learning and 2D face segmentation c. Given a 2D facial image I, PerspNet G predicts $[X, M, c] = G(I)$. With the input, an encoder and two decoders are trained to extract 3D features f_{3D} and 2D image features f_{2D} respectively. The 3D features f_{3D} are fed into 3D face shape reconstruction module to regress the UV position map, which represents the 3D face shape information in canonical space. Then the 2D features f_{2D} includes 2D facial image features f_{im} and encoded 2D position features f_{pos}, and then fuse with 3D features f_{3D} to learn the correspondence between 2D pixels in 2D image and 3D points in 3D face shape. With the correspondence, the 6DoF pose of the face can be computed. In addition, 2D image features are also fed into 2D face segmentation module, which is used to extract 2D observed pixels in face regions during testing.

3D Face Shape Reconstruction. Different from the UV formulation in [13] which is defined in image coordinate system, our UV position map $S \in \mathbb{R}^{H\times W\times 3}$ records the 3D coordinates of 3D facial structure with the canonical pose, which only represents the facial shape. Specifically,

$$S = Render(S_{coord}, X, T), \tag{3}$$

where S is the rendered UV position map, $S_{coord} \in \mathbb{R}^{2\times n}$ represents UV coordinates recording the 2D locations of n 3D vertices X in UV map, and $T \in \mathbb{R}^{k\times 3}$ is denoted as k triangles in a 3D face mesh. A fully convolutional encoder-decoder architecture is utilized to regress the UV position map, as shown in Fig. 3. To supervise the 3D face shape prediction, a weighted L1 loss function L_S is used to measure the difference between ground-truth position map S and the network output $\hat{S}$,

$$L_S = \sum ||S - \hat{S}|| \cdot W(S_{coord}), \tag{4}$$

where $W \in \mathbb{R}^{H\times W}$ is a weight matrix for S, and we set $W(S_{coord}) = 1$, others in W to 0. Then we extract the 3D vertices X of the face from UV position map S using UV coordinates.

2D-3D Correspondence Learning. To estimate the 6DoF pose of the face in 2D images, we use a two-stage pipeline that first learns the correspondence of 2D pixels in 2D image and 3D points in reconstructed 3D face shape, and then compute 6D face pose parameters using a PnP algorithm. We design a 2D-3D correspondence learning module in PerspNet for the first stage, as shown in Fig. 3. We build a correspondence probability matrix $M \in \mathbb{R}^{m\times n}$ where each row records the corresponding 3D points in canonical space of one pixel on the input image. Here m is the number of face region pixels selected from a 2D image and n is the number of vertices in 3D face shape. The estimating of M is as:

$$M = \text{softmax}(MLP(f_{2D}, f_{3D})). \tag{5}$$

To learn the correspondence M between 2D and 3D points, we extract 2D features f_{2D} and 3D features f_{3D} respectively.

For 2D features f_{2D}, we firstly extract the image features f_{im} from another fully convolutional decoder architecture after the encoder network from the same facial image. For each pixel in V, we also extract 2D position features f_{pos} in 2D images and fuse f_{im} as its local feature $f_{2Dlocal}$. The position features are encoded by a 2D position encoding method, which is an extension of 1D position encoding from [34]. 2D global features $f_{2Dglobal}$ are also learned by feeding 2D local features $f_{2Dlocal}$ into Multi-Layer Perceptron (MLP) layers, followed with a global average pooling layer. Then the 2D local $f_{2Dlocal}$ and global features $f_{2Dglobal}$ are fused as 2D features f_{2D}.

As for the 3D features f_{3D}, since the UV position map contains 3D geometry information, we also extract 3D global features f_{3D} from the UV position regression network followed by MLP layers and a global average pooling layer. Then the 2D features f_{2D} and 3D features f_{3D} are fused and fed into MLP layers and a softmax layer. In this way, the correspondence matrix M is predicted from the network.

To achieve the ground-truth M, for each pixel in the 2D face region, we compute its barycentric coordinates [35] based on three vertices of its located triangle. The barycentric coordinates can be taken as corresponding probability between this 2D pixel and the three vertices in the 3D face mesh. Other values in M are set to 0. Each row M_i of the matrix M represents the distribution over the correspondences between i-th pixels in V and the vertices X. We utilize a Kullback-Leibler (KL) divergence loss for the correspondence matrix M. In addition, since the matrix M is very sparse, we also minimize its entropy to regularize the matrix. The final loss for M is

$$L_M = \frac{1}{m}(\sum_{i=0}^{m} D_{KL}(\hat{M}_i||M_i) - \lambda \sum_{i=0}^{m}\sum_{j=0}^{n} \hat{M}_{ij} \log \hat{M}_{ij}). \tag{6}$$

Here M and $\hat{M}$ are ground-truth and predicted correspondence matrix, and λ is a constant weight. With the predicted matrix $\hat{M}$ and reconstructed 3D face $\hat{X}$, the corresponding 3D points for each 2D pixels X_{corr} are obtained by matrix multiplication, $\hat{X}_{corr} = \hat{M} \times \hat{X}^{\mathrm{T}}, X_{corr} \in \mathbb{R}^{m\times 3}$. L1 loss L_{corr} is also applied to supervise the $\hat{X}_{corr}$:

$$L_{corr} = ||X_{corr} - \hat{X}_{corr}||_{L1} \tag{7}$$

where ground-truth X_{corr} is achieved by perspective projection.

2D Face Segmentation. 2D face segmentation module in the proposed network is trained for selecting 2D pixels from face regions in the inference phase. When training the whole network, the m image pixels are randomly chosen from ground-truth face segmentation mask. While in the testing phase, the m image pixels are randomly chosen from predicted face mask. 2D face segmentation task follows the 2D image feature extraction network and a 2-class softmax loss L_{seg} is used.

Training Objective. We train out whole network with a multi-task loss. The final loss function is

$$L = \lambda_1 L_S + \lambda_2 L_M + \lambda_3 L_{corr} + \lambda_4 L_{seg}. \tag{8}$$

Dataset	Sub. Num	Image Num	3D Mesh Num	Exp. Num	camera	Vert. Num	6DoF Pose
Bosphorus[32]	105	4,666	4,666	35	Mega	35K	No
MICC[2]	53	53	203	≤ 5	3DMD	40K	No
3DFAW[27]	26	26	26	Neutral	DI3D	20K	No
BP4D[39]	41	328	328	8	3DMD	70k	No
AFLW2000-3D[41]	2,000	2,000	3DMM	-	-	53,149	No
BIWI[11]	20	15,678	24	Neutral	Kinect	6,918	Yes
NoW[31]	100	2,054	100	Neutral	iPhone X	58,668	No
FaceScape[37]	938	1,275,680	18,760	20	DSLR	2M	fixed pose
ARKitFace	500	902,724	902,724	33	iPhone 11	1,220	Yes

Table 1. Comparing ARKitFace with other 3D Face datasets. Exp. and Vert. are abbreviations of the annotation number of categories of expressions and number of vertices for 3D mesh, respectively.

where $\lambda_1, \lambda_2, \lambda_3$ and λ_4 are the weights of the four losses respectively. Experimental results reveal that jointly training these tasks boosts the performance of each other.

3.2 6DoF Face Pose Estimation

Based on the output of the network, the final 6DoF pose estimation can be computed. Given the chosen 2D pixel coordinates V in the original full 2D image, their corresponding 3D points coordinates X_{corr} from reconstructed faces in world coordinate and the camera intrinsic parameters K, we apply a PnP algorithm [23] with Random Sample Consensus (RANSAC) [14] to compute the 6D face pose parameters, $(R, t) = PnP(K, V, X_{corr})$. Perspective-n-Point is the problem of estimating the pose of a calibrated camera given a set of m 3D points in the world and their corresponding 2D projections in the image. The camera pose consists of 6DoF which are made up of the rotation (roll, pitch, and yaw) and 3D translation of the camera with respect to the world. With the estimated 6DoF face pose, the reconstructed 3D face shapes can be projected to 2D images. It is worth noting that, directly regressing 6DoF pose parameters from a single image by CNN is also feasible, but it achieves much worse performance than our method due to the nonlinearity of the rotation space [26]. It will be validated in our experiments.

4 ARKitFace Dataset

The ARKitFace dataset is established by this work in order to train and evaluate both 3D face shape and 6DoF in the setting of perspective projection. A total of 500 volunteers, aged 9 to 60, are invited to record the dataset. They sit in a random environment, and the 3D acquisition equipment is fixed in front of them, with a distance ranging from about 0.3m to 0.9m. Each subject is asked to perform 33 specific expressions with two head movements (from looking left to looking right / from looking up to looking down). 3D acquisition equipment we used is an iPhone 11. The shape and location of human face are tracked by structured light sensor. The triangle mesh and 6DoF information of the RGB

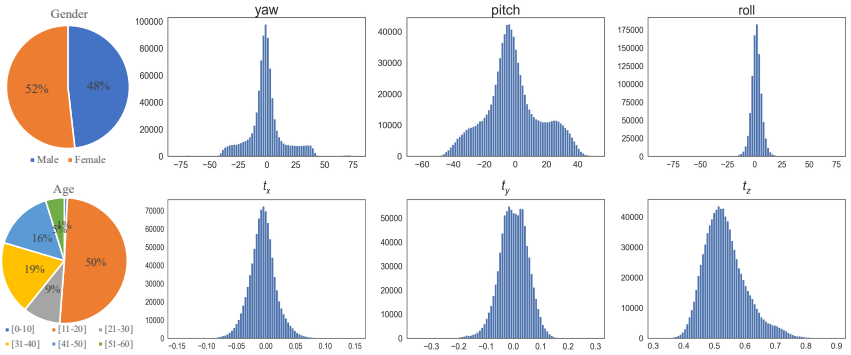

Fig. 4. Distributions of age, gender, and each pose parameter of 6DoF on ARKitFace dataset.

images are obtained by built-in ARKit toolbox. The triangle mesh is made up of 1,220 vertices and 2,304 triangles. In total, 902,724 2D facial images (resolution 1280×720 or 1440×1280) with ground-truth 3D mesh and 6DoF pose annotation are collected. An example is shown in Fig. 2. Distributions of age, gender, and each pose parameter of 6DoF on ARKitFace dataset are shown in Fig. 4. We can observe that our dataset has balanced gender, diverse age and the 6DoF pose varation. Comparisons between different datasets shown in Table 1 reveal that ARKitFace surpasses the existing datasets in terms of scale, 3D exact shape annotations and diversity of poses.

Authorization: All the 500 subjects consent to use their data. We will release all the subjects with 2D facial images, 3D mesh and 6DoF pose annotation under the authorization of all the subjects. We will not release their personal privacy information, including age, gender etc.

5 Experiments

5.1 Implementation Details

Our PerspNet is implemented by PyTorch [24]. During training, the PerspNet takes the input image cropped from a full 2D image and resized to $192\times192\times3$, based on the ground-truth face segmentation mask. To augment the data with large poses, we utilize face profiling method [40] to generate the profile view of faces from medium-pose samples for three euler angles. We enlarge all the three angles to max 90° and min -90°. We also apply online data augmentation including random cropping, resizing and color jittering during training. We use a pre-trained ResNet-18 [17] architecture, where the final encoded feature map is $6\times6\times512$. To regress the $192\times192\times3$ UV position map , the first decoder is implemented by 5 up-sampling layers and its output is a $192\times192\times32$ feature map, which is regarded as the 3D features. The number of point clouds n in 3D face shape is 1220. To extract 2D image features and segment 2D face region

from background, the second decoder consists of five up-sampling layers and each up-sampled feature map is concatenated with a feature map which have the same size in the encoder backbone network. The size of 2D image features is also $192 \times 192 \times 32$. The number of randomly sampled 2D pixels m for face region is 1024. If point count in face region is insufficient, we sample these pixels by repetition. The 2D and 3D global feature sizes are all 1×1024. At the training phase, the 2D pixels are randomly chosen from ground-truth face mask. We set the weights of losses respectively, $\lambda_1 = 0.5, \lambda_2 = 0.01, \lambda_3 = 1.0, \lambda_4 = 0.01, \lambda = 0.1$. The initial learning rate is set as 0.0001 and is updated linearly after 10 epochs. We train all models for 20 epochs. All the networks are trained and evaluated on the ARKitFace dataset with ground-truth bounding box. At the testing phase, the 2D pixels are randomly chosen from segmented face region. The PnP algorithm is implemented in OpenCV [4].

5.2 Dataset

To validate our proposed method, we conducted experiments on our collected ARKitFace dataset and a public BIWI dataset.

ARKitFace. We randomly use 400 people in our dataset as training data with a total of 717,840 2D facial images and annotations, leaving 100 people with totally 184,884 samples for testing.

BIWI. BIWI [11] contains 24 videos of 20 subjects in an indoor environment. Each video is coupled with a neutral 3D face mesh of a specific person. There are totally 15, 678 frames with a wide range of face poses in this dataset. Paired RGB and Depth images are provided for each frame. We only use the RGB images as inputs in this work. This benchmark provides ground-truth labels for rotation (rotation matrix) and translation for full 6DoF. Since there is not each 3D face mesh for each frame, we can not evaluate our method on 3D face reconstruction task. We only evaluate our method for 6DoF pose estimation task. In addition, we train our method on training set of ARKitFace dataset, and test our method on the entire BIWI dataset following the previous methods [20,38,1].

5.3 Evaluation Metric

For the 3D face shape reconstruction, we follow previous works [12,31], median distance and average distance between predicted 3D mesh vertices and ground-truth 3D mesh vertices are utilized. For the 6DoF face pose estimation, we follow previous head/face pose estimation methods [1,38,20], and convert the rotation matrixes to 3 euler angles,Yaw, Pitch, Roll and compute the mean absolute error (MAE) for 6DoF, $Yaw, Pitch, Roll, t_x, t_y, t_z$. MAE_r and MAE_t, the rotational and translational MAE are also computed. Furthermore, to validate the 6DoF face pose in a metric, we adopt the average 3D distance (ADD) metric [19] used for object pose evaluation. Given the ground-truth rotation R and translation t and the estimated rotation $\hat{R}$ and translation $\hat{t}$, the ADD computes the mean of the pairwise distances between the ground-truth 3D face model points $x \in X$

Method	Yaw	Pitch	Roll	MAE_r	t_x	t_y	t_z	MAE_t	ADD
img2pose(retrain) [1]	5.07	7.32	4.25	5.55	1.39	3.72	15.95	7.02	20.54
Direct 6DoF Regress	1.86	2.72	1.03	1.87	2.80	5.23	19.16	9.06	21.39
PerspNet w/o PE	1.01	1.53	0.61	1.05	1.17	2.39	11.77	5.11	12.34
PerspNet w/o L_M	1.04	1.45	0.60	1.03	1.09	2.13	10.28	4.50	10.89
PerspNet (ours)	**0.99**	**1.43**	**0.55**	**0.99**	**0.97**	**2.12**	**9.45**	**4.18**	**10.01**

Table 2. Comparisons with different methods for 6DoF Face Pose Estimation on ARKitFace test dataset.

Method	Yaw	Pitch	Roll	MAE_r	t_x	t_y	t_z	MAE_t	ADD
Dlib (68 points)[22]	16.76	13.80	6.19	12.25	-	-	-	-	-
3DDFA[40]	36.18	12.25	8.78	19.07	-	-	-	-	-
FAN (12 points)[5]	8.53	7.48	7.63	7.88	-	-	-	-	-
Hopenet ($\alpha = 1$)[30]	4.81	6.61	3.27	4.90	-	-	-	-	-
QuatNet[20]	4.01	5.49	2.94	4.15	-	-	-	-	-
FSA-NET[38]	4.56	5.21	3.07	4.28	-	-	-	-	-
HPE[21]	4.57	5.18	3.12	4.29	-	-	-	-	-
TriNet[7]	**3.05**	4.76	4.11	3.97	-	-	-	-	-
RetinaFace R-50(5 points)[10]	4.07	6.42	2.97	4.49	-	-	-	-	-
img2pose[1]	4.57	3.55	3.24	3.79	-	-	-	-	-
Direct 6DoF Regress	16.49	14.03	5.81	12.11	62.36	85.01	366.52	171.30	562.38
PerspNet w/o PE	3.63	3.81	3.48	3.64	6.03	9.11	77.87	31.00	142.20
PerspNet w/o L_M	3.67	3.52	3.26	3.48	5.57	8.53	75.23	29.78	136.16
PerspNet (ours)	3.10	**3.37**	**2.38**	**2.95**	**4.15**	**6.43**	**46.69**	**19.09**	**100.09**

Table 3. Comparisons with different methods for 6DoF Face Pose Estimation on BIWI dataset.

transformed based on the ground-truth pose and the estimated pose: $ADD = \text{avg}_{x \in X}||(Rx + t) - (\hat{R}x + \hat{t})||$.

5.4 Evaluation for 6DoF Face Pose Estimation

Comparison with the state-of-the-art methods. To compare with the state-of-the-art methods on head or face pose estimation, we firstly retrain the most recent state-of-the-art method, img2pose [1], on ARKitFace training set, and compute the performance on testing data. As shown in Table 2, our method outperforms it significantly. In addition, the public BIWI dataset is used as a cross-data evaluation to test our final network. Since some faces with large angles can not be detected, we follow the img2pose [1] method and test on 13,219 images with detected bbox. We use the code of [5] to detect the face 68 landmarks and crop facial region. All the results are shown in Table 3. We can see that our method performs much better than previous methods, especially the recent img2pose method [1]. The experiment also demonstrates that our method and dataset can be well generalized to the data in different domain. The inference time of our proposed model is 11.7ms with a P100 GPU.

Ablation Studies. We build several baselines to evaluate the components that contribute to our performance. Since img2pose [1] directly regresses the 6 pose parameters, we build a baseline, Direct 6DoF Regress, to regress the 6 pose parameters after the backbone network. Other baselines include our PerspNet without the position encoding features (PerspNet w/o PE), which is used to validate the effectiveness of the 2D position encoding features, and our Persp-

Fig. 5. Qualitative results for 6DoF pose estimation and 3D face shape reconstruction on ARKitFace dataset.

Method	Median(mm)	Mean(mm)
PRNet [13]	1.97	2.05
3DDFA_v2 [16]	2.35	2.31
PerspNet (ours)	**1.72**	**1.76**

Table 4. Results for 3D face shape reconstruction on ARKitFace.

Net without L_M loss (PerspNet w/o L_M), which is built to evaluate the corresponding components. The results are shown in Table 2 and Table 3. We can observe that our two-stage method outperforms direct regression method significantly, and 2D position encoding features are helpful and L_M is effective for face pose estimation task. Moreover, to explain the influence of segmentation mask, the GT face mask is used during the inference time. Its results on $Yaw, Pitch, Roll, MAE_r, t_x, t_y, t_z, MAE_t, ADD$ are 0.72, 1.10, 0.54, 0.79, 0.92, 1.47, 9.59, 3.99, 9.99 respectively. It shows that the method with the ground-truth face mask performs better than that with predicted segmentation mask, which indicates that more accurate segmentation results are needed.

5.5 Evaluation on 3D Face Shape Reconstruction

To validate the proposed method on 3D face shape reconstruction task, we display the results on ARKitFace test data in Table 4. For comparison, we also train a single-task PRNet [13] with the same encoder-decoder UV regression network in our multi-task network on ARKitFace training data. As shown in Table 4, our multi-task network outperforms the single-task PRNet, which reveals that the pose estimation task contributes to the improvement of 3D face shape reconstruction task. In addition, we compare our method with other SOTA like 3DDFA_v2 [16] in Table 4. We can see that our method still achieves the best performance of 3D face reconstruction.

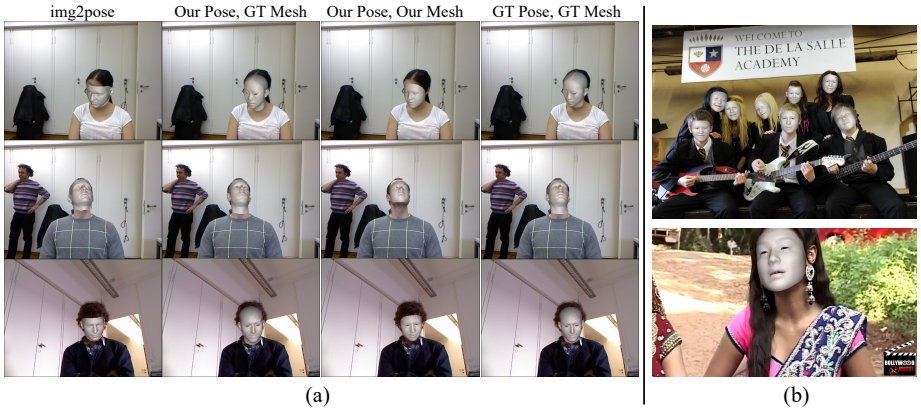

Fig. 6. Qualitative results for 6DoF pose estimation and 3D face shape reconstruction on (a) BIWI dataset and (b) WIDER FACE validation images.

5.6 Qualitative Results

We display qualitative results for 6DoF pose estimation and 3D reconstruction on ARKitFace and BIWI dataset in Fig. 5 and Fig. 6 (a), where img2pose, our predicted results and GT results participate in the comparison. The predicted face pose with GT 3D mesh, the predicted face pose with predicted 3D mesh and their error map are demonstrated, respectively. We can see that our results are effective and outperforms img2pose, especially in large pose. We also show some qualitative results on in-the-wild images from WIDER FACE dataset in Fig. 6 (b). It shows that our method also performs well in in-the-wild images.

6 Conclusion

We explore 3D face reconstruction under perspective projection from a single RGB image for 3D face AR applications. We introduce a novel framework, in which a deep learning network, PerspNet, is proposed, for 3D face shape reconstruction, corresponding learning between 2D pixels and 3D points in 3D face models, and 2D face region segmentation. With 2D pixels in facial images and corresponding 3D points in reconstructed 3D face mesh, 6DoF face pose is estimated by a PnP method. This 6DoF face pose is used for perspective projection transformation. To enable our PerspNet, we build a large-scale 3D face dataset, ARKitFace dataset, annotating 2D facial images, 3D face mesh and 6DoF pose. Experiments demonstrate the effectiveness of our approach for 3D face shape reconstruction and 6DoF pose estimation. As most of the face analysis methods, our method and data may raise privacy concerns when misused. Therefore, the release of the data is fully authorized by the subjects. We wish this work would spur the future researches including 3D face reconstruction and face pose estimation.

References

1. Albiero, V., Chen, X., Yin, X., Pang, G., Hassner, T.: img2pose: Face alignment and detection via 6dof, face pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7617–7627 (2021)
2. Bagdanov, A.D., Del Bimbo, A., Masi, I.: The florence 2d/3d hybrid face dataset. In: Proceedings of the 2011 joint ACM workshop on Human Gesture and Behavior Understanding. pp. 79–80 (2011)
3. Bai, Z., Cui, Z., Liu, X., Tan, P.: Riggable 3d face reconstruction via in-network optimization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6216–6225 (2021)
4. Bradski, G., Kaehler, A.: Opencv. Dr. Dobb's journal of software tools **3** (2000)
5. Bulat, A., Tzimiropoulos, G.: How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In: Proceedings of the International Conference on Computer Vision (2017)
6. Cao, C., Weng, Y., Zhou, S., Tong, Y., Zhou, K.: Facewarehouse: A 3d facial expression database for visual computing. IEEE Transactions on Visualization and Computer Graphics **20**(3), 413–425 (2013)
7. Cao, Z., Chu, Z., Liu, D., Chen, Y.: A vector-based representation to enhance head pose estimation. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision. pp. 1188–1197 (2021)
8. Chang, F.J., Tran, A.T., Hassner, T., Masi, I., Nevatia, R., Medioni, G.: Deep, landmark-free fame: Face alignment, modeling, and expression estimation. International Journal of Computer Vision **127**(6), 930–956 (2019)
9. CS, C.V.: The geometry of perspective projection. `https://www.cse.unr.edu/~bebis/CS791E/Notes/PerspectiveProjection.pdf` (2021)
10. Deng, J., Guo, J., Ververas, E., Kotsia, I., Zafeiriou, S.: Retinaface: Single-shot multi-level face localisation in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5203–5212 (2020)
11. Fanelli, G., Dantone, M., Gall, J., Fossati, A., Van Gool, L.: Random forests for real time 3d face analysis. International Journal of Computer Vision **101**(3), 437–458 (2013)
12. Feng, Y., Feng, H., Black, M.J., Bolkart, T.: Learning an animatable detailed 3d face model from in-the-wild images. ACM Transactions on Graphics **40**(4), 1–13 (2021)
13. Feng, Y., Wu, F., Shao, X., Wang, Y., Zhou, X.: Joint 3d face reconstruction and dense alignment with position map regression network. In: Proceedings of the European Conference on Computer Vision. pp. 534–551 (2018)
14. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6), 381–395 (1981)
15. Forsyth, D., Ponce, J.: Computer vision: A modern approach. Prentice hall (2011)
16. Guo, J., Zhu, X., Yang, Y., Yang, F., Lei, Z., Li, S.Z.: Towards fast, accurate and stable 3d dense face alignment. In: Proceedings of the European Conference Computer Vision. pp. 152–168 (2020)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
18. He, Y., Sun, W., Huang, H., Liu, J., Fan, H., Sun, J.: Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 11632–11641 (2020)

19. Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., Navab, N.: Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In: Proceedings of the Asian Conference on Computer Vision. pp. 548–562 (2012)
20. Hsu, H.W., Wu, T.Y., Wan, S., Wong, W.H., Lee, C.Y.: Quatnet: Quaternion-based head pose estimation with multiregression loss. IEEE Transactions on Multimedia **21**(4), 1035–1046 (2018)
21. Huang, B., Chen, R., Xu, W., Zhou, Q.: Improving head pose estimation using two-stage ensembles with top-k regression. Image and Vision Computing **93**, 103827 (2020)
22. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1867–1874 (2014)
23. Lepetit, V., Moreno-Noguer, F., Fua, P.: Epnp: An accurate o (n) solution to the pnp problem. International Journal of Computer Vision **81**(2), 155 (2009)
24. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in Neural Information Processing Systems **32**, 8026–8037 (2019)
25. Paysan, P., Knothe, R., Amberg, B., Romdhani, S., Vetter, T.: A 3d face model for pose and illumination invariant face recognition. In: IEEE International Conference on Advanced Video and Signal based Surveillance. pp. 296–301 (2009)
26. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4561–4570 (2019)
27. Pillai, R.K., Jeni, L.A., Yang, H., Zhang, Z., Yin, L., Cohn, J.F.: The 2nd 3d face alignment in the wild challenge (3dfaw-video): Dense reconstruction from video. In: ICCV Workshops. pp. 3082–3089 (2019)
28. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in Neural Information Processing Systems **28**, 91–99 (2015)
29. Romdhani, S., Vetter, T.: Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. vol. 2, pp. 986–993 (2005)
30. Ruiz, N., Chong, E., Rehg, J.M.: Fine-grained head pose estimation without keypoints. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 2074–2083 (2018)
31. Sanyal, S., Bolkart, T., Feng, H., Black, M.J.: Learning to regress 3d face shape and expression from an image without 3d supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7763–7772 (2019)
32. Savran, A., Alyüz, N., Dibeklioğlu, H., Çeliktutan, O., Gökberk, B., Sankur, B., Akarun, L.: Bosphorus database for 3d face analysis. In: European Workshop on Biometrics and Identity Management. pp. 47–56 (2008)
33. Tian, M., Ang, M.H., Lee, G.H.: Shape prior deformation for categorical 6d object pose and size estimation. In: Proceedings of the European Conference on Computer Vision. pp. 530–546 (2020)
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. pp. 5998–6008 (2017)

35. Weisstein, E.W.: Barycentric coordinates. https://mathworld.wolfram.com/ (2003)
36. Xin, M., Mo, S., Lin, Y.: Eva-gcn: Head pose estimation based on graph convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1462–1471 (2021)
37. Yang, H., Zhu, H., Wang, Y., Huang, M., Shen, Q., Yang, R., Cao, X.: Facescape: A large-scale high quality 3d face dataset and detailed riggable 3d face prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (June 2020)
38. Yang, T.Y., Chen, Y.T., Lin, Y.Y., Chuang, Y.Y.: Fsa-net: Learning fine-grained structure aggregation for head pose estimation from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1087–1096 (2019)
39. Zhang, X., Yin, L., Cohn, J.F., Canavan, S., Reale, M., Horowitz, A., Liu, P., Girard, J.M.: Bp4d-spontaneous: a high-resolution spontaneous 3d dynamic facial expression database. Image and Vision Computing **32**(10), 692–706 (2014)
40. Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S.Z.: Face alignment across large poses: A 3d solution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 146–155 (2016)
41. Zhu, X., Liu, X., Lei, Z., Li, S.Z.: Face alignment in full pose range: A 3d total solution. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(1), 78–92 (2017)
42. Zhu, X., Yang, F., Huang, D., Yu, C., Wang, H., Guo, J., Lei, Z., Li, S.Z.: Beyond 3dmm space: Towards fine-grained 3d face reconstruction. In: Proceedings of the European Conference Computer Vision. pp. 343–358 (2020)

A Supplementary Materials

A.1 Additional Qualitative Results

Fig. 7, Fig. 8 and Fig. 9 present more qualitative results. In Fig. 8, there are results constructed by our pose and ground-truth (GT) mesh, our pose and our mesh, GT pose and GT mesh. Our pose and GT mesh means the projected faces in silver are rendered with our predicted pose and GT mesh. Our pose and our mesh means the projected faces in silver are rendered with our predicted pose and predicted mesh. GT pose and our mesh means the projected faces in silver are rendered with our predicted pose and predicted mesh. We also give the distance error map coupled with the rendered face. We can observe that our predicted results are much similar to GT face and the error often exists in large pose.

In Fig. 7, there are results constructed by img2pose [1], our pose and GT mesh, our pose and our mesh, GT pose and GT mesh. For img2pose, we use their code to render these results. We can see that our method predicts better alignment even with our pose and our mesh than img2pose. In img2pose, they simulate the process of camera visual field from focusing on the whole image to focusing on the local bounding box, and converts the global 6DoF to local 6DoF by specific linear transformations. Since the perspective distortion of the local facial appearance is ignored, there exists ambiguity in local 6DoF. While our proposed PerspNet selects 2D pixel points from the full image and predicts their corresponding 3D vertices in canonical space. With the known camera intrinsic matrix, the final predicted 6DoF can be recovered by PnP more accurately. Hence better performance can be achieved by our proposed method. The images displayed also show the robustness of our method across dataset, especially in large pose.

We also show some qualitative results on in-the-wild images from WIDER FACE dataset in Fig. 9 with our predicted pose and our predicted mesh. It shows that our method also performs well in in-the-wild images.

A.2 Details about ARKitFace Dataset

More samples on ARKitFace dataset are provided in Fig. 10 with projected 3D face in silver. All the rendered 3D faces are projected by a perspective transformation with our pose annotations.

A.3 Implenmental details about 2D-3D Correspondence Matrix

To supervise the matrix M, we compute its ground truth for each face based on barycentric coordinates [35]. In this work, our 3D face mesh is a triangle mesh with n 3D vertices and t triangles. Each triangle consists of 3 vertices and 3 edges. When a 3D face mesh is projected to a 2D image, a pixel P in the 2D face region only belongs to a triangle with three vertices,P_i, P_j, P_k, as shown in Fig. 11. Its barycentric coordinate (w_i, w_j, w_k) in this triangle, with one

Fig. 7. Qualitative results for 6DoF pose estimation and 3D face shape reconstruction on BIWI dataset.

Fig. 8. Qualitative results for 6DoF pose estimation and 3D face shape reconstruction on ARKitFace dataset.

Fig. 9. Qualitative results for 6DoF pose estimation and 3D face shape reconstruction on ARKitFace dataset.

Fig. 10. More samples on ARKitFace dataset.

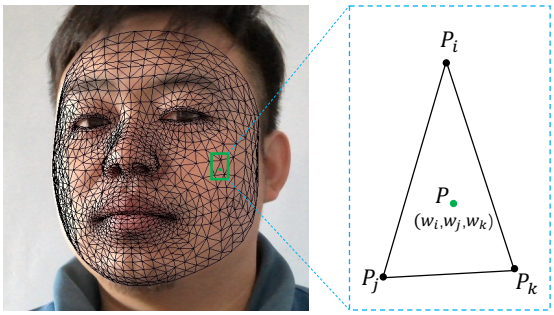

Fig. 11. A cropped 2D facial image with its projected 3D face triangles mesh in black. The barycentric coordinate of a pixel P (green) in 2D facial image is calculated by three projected vertices P_i, P_j, P_k of a triangle.

additional condition $w_i + w_j + w_k = 1$, can be taken as corresponding probability between these 2D pixels and the three vertices in the 3D face mesh.