3D Human Face Reconstruction From Single Image

Zhipeng Fan

1 Introduction

High fidelity 3D human face reconstruction from single image has long been an interested topic studied by both the computer graphics community and the computer vision researchers. Traditionally, the 3DMM [2, 1] face models has been adopted by most of the works to perform the face reconstruction. As a Principle Component Analysis based method, 3DMM face models assumes that human face could be linearly represented by low dimensional data combined with the corresponding learned face basis, which could be mathematically formulated as

$$p = \bar{p} + A_{\text{id}}\alpha_{\text{id}} + A_{\text{exp}}\alpha_{\text{exp}} \quad (1)$$

$$b = \bar{b} + A_{\text{alb}}\alpha_{\text{alb}} \quad (2)$$

where p denotes the shape of the face mesh, which is further separated to the average shape $\bar{p}$ under neutral expression, the offset basis corresponds to the identity of the subject A_{id} and the offset basis corresponds to different expression of the subject A_{exp}. Similarly, b represents the texture of the face, which usually denotes as the rgb color on each vertices. A_{alb} corresponds to the offset basis of face texture differences caused by different subject identity. α_{id}, α_{exp} and α_{alb} are the linear coefficient vectors that characterize the 3D face models for different subjects. Within this line of work, multiple different face models were learned from multiple collected face scan datasets. To name a few, the Basel Face Model(BFM) [8] and the FaceWarehouse [3].

However, those methods face a common issue: representing the high dimensional human faces in lower dimension inevitably lose the high frequency information of human faces, which results in an over-smoothed reconstructed human faces. Several works that published recently tries to address this issues by directly learning an non-linear face models[10, 11, 12, 13, 14]. Specifically, [10] directly learns a face model from scratch based on the large amount of videos in an unsupervised manner. In [12], instead of learning the color on the each vertex, the author innovatively proposed to learn the corresponding 2D texture map while in [14], the focus is on learning a depth map that could faithfully reflects the fine details on human face (including the winkles, moustache, eyebrows, etc).

In this project, we'd like to borrow the best from [12] and [14], where in addition to the base face model reconstruction using the 3DMM face, we use a 2D texture map to represent the textures of human face as well as a 2D shape offset map to add the shape details of human faces.

2 Methodology

In general, our methods contains an encoder network to map the human faces from the 2D images to the lower dimensional 3DMM identity, expression and albedo coefficients space. In addition to the 3DMM parameters, this model also regress the scale/rotation/translation parameters to perform global transformations and the lighting parameters. We assume the face of human beings as a Lambertian surface, which could be parameterized with the following equation:

$$L(b_i, n_i|\gamma) = b_i \sum_{b=1}^{B^2} \gamma_k \phi_k(n_i) \quad (3)$$

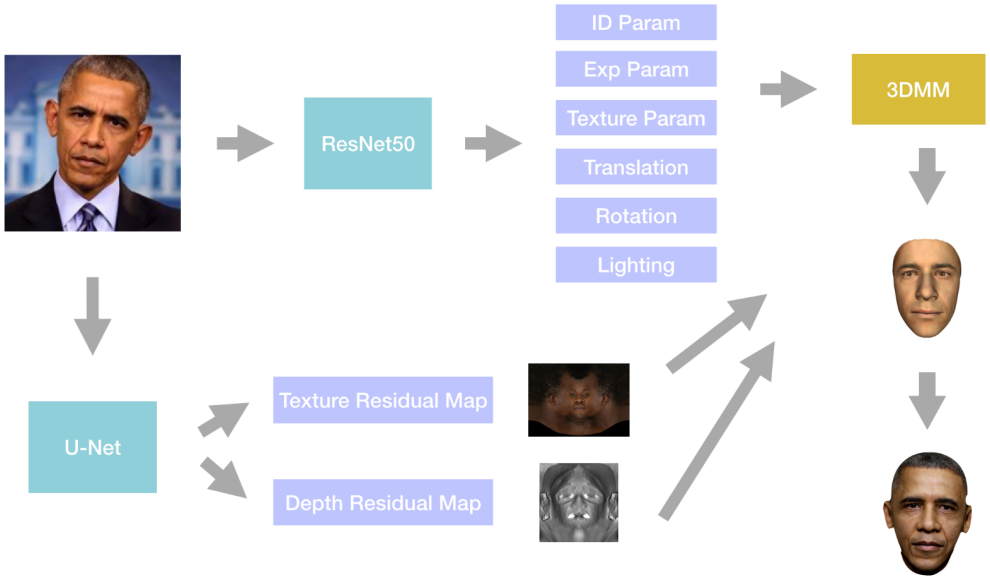

Figure 1: Pipeline for 3D human face reconstruction with offset texture map and displacement map learning. In the upper branch, we show the base face reconstruction module, which employs the 3DMM face model. For the lower branch, we additionally proposed to learn the offset texture map and displacement to add additional details to the face model

Where L is the final color of the face model, n_i and b_i are the normal and albedo at vertex i. B corresponds to the order of spherical harmonics and γ is the corresponding coefficient. ϕ is the spherical harmonic function, which takes a surface normal as the input. In our experiments, we use third order spherical harmonics lighting function, which corresponds to 9 lighting parameters per band and 27 lighting parameters in total.

The predicted α_{id}, α_{exp} and α_{albedo} is then used to reconstruct the face mesh following the equation 1 and equation 2. After transforming the reconstructed mesh with the predicted transform parameter(scale, rotation and translation), we employed a differentiable renderer [7] to rasterize the image. The reconstruct image, along with the original input image, are concatenated and then fed into a double-head UNet [9] was designed to predict the surface offset map and the texture map, which is applied on the rough estimation from the 3DMM models to reconstruct the face in the input images faithfully and vividly. The entire pipeline is illustrated in Figure 1. We refer to the upper branch, where we perform reconstruction based on the 3DMM face model as the base branch, and the lower branch, where we introduced the residual texture map and displacement map, as the augmented branch.

2.1 Base branch: Model based 3D human face reconstruction

In the base branch, we employed the ResNet50 [6] as the model to map the input 2D images to the 3DMM coefficients. We reconstruct the shape of face follow equation 1 and the albedo follow equation 2. After performing model transformation, We compute the face normal first and then compute the vertex normal based on the trianglation between vertices. The vertex normal is then used in equation 3 to compute the color after adding lighting.

The final mesh is then rendered with a differentiable renderer (SoftRasterizer[7]). We use the differentiable renderer instead of an ordinary rasterization renderer since traditional rasterization process is a non-differentiable process: for example, consider a moving object that is gradually occluded by another

Figure 2: 3D face reconstruction result from the base branch.

static object that lies in the front, the change from "visible" to "non visible" is abrupt, which blocks the gradient to back propagated to the base neural net. For rendering, we assume perspective projection with a FoV of 30°. The rendered image is compared with the original input image on the rendered part. We introduced 2 supervision here: the pixel-wise distance, the landmark distance and face recognition model feature loss.

$$L_{\text{pixel}} = \frac{1}{n}\sum_{\text{k}=1}^{n}\left\|\widehat{I}\odot I - \widehat{I}\right\|^2 \tag{4}$$

$$L_{\text{landmark}} = \frac{1}{n}\sum_{\text{k}=1}^{n}\frac{1}{68}\sum_{i=1}^{68}\|y-\widehat{y}\|_1 \tag{5}$$

$$L_{\text{recog}} = \frac{1}{n}\sum_{\text{k}=1}^{n}\left\|F(\widehat{I}\odot I) - F(\widehat{y})\right\|_1 \tag{6}$$

where $\widehat{I}$ indicates the reconstructed image using predicted coefficients and I is the ground truth image. $\odot$ indicates the element-wise multiplication. We masked out the face region to avoid the loss incurred by the background difference. In 5, $\widehat{y}$ indicates landmark extracted from the face mesh and y is the pseudo-ground truth for the 68 landmarks on human face, which we obtained by running [15]. For the recognition loss6, $F()$ corresponds to the face recognition model loss. The overall loss we used here the weighted combination of pixel loss, landmark loss and recognition loss:

$$L_{\text{base}} = L_{\text{pixel}} + L_{\text{landmark}} + 20L_{\text{recog}} \tag{7}$$

We set the learning rate to $3\text{e}-5$ and train the base ResNet50 model for 20 epochs on CACD2000 dataset[4], which contains more than 160,000 images of 2,000 celebrities with age ranging from 16 to 62. We random select 70% of them for training and 15% for validation and test respectively. We initialize the model with the pre-trained model on ImageNet[5] for object classification and each epoch take around 7 hours. The results are visualized in Figure 2

2.2 Augmented branch: 3D human face reconstruction with

After finishing the base branch training, we fix the ResNet50 model to train the second model to learn the offset texture map and the displacement map. We design the texture map and displacemap as 2D images with the spatial resolution of 224×224.

We employed the U-Net[9] structure for the texture map and the displacement generation as U-Net has been largely adopted for image to image generation task and show superior results compared to the other architectures. U-Net contains an encoder part and decoder part, which perform the information compression and generation respectively. Here, we designed a double head U-Net structure, meaning that in the encoder

Figure 3: Degenerated results when training the augmented branch using pixel loss and landmark loss only.

part, the texture map and displacement map share the model. We split the texture map branch and the displacement branch after the 2nd up-sampling layer. At the output side, we generate the texture offset map and the displacement map, which each has 3 channels.

We use bi-linear sampling to obtain the offset for each vertex on the 3DMM model instead of hyperbolic sampling as the denominator term in hyperbolic sampling makes the training much more unstable in our preliminary study (fitting the U-Net on a subset of the training set). We conjecture the reason is that 3DMM model resides in the clip coordinates, which has relatively small depth value and the denominator term therefore magnifies the gradient.

Directly training the U-Net using the loss in 4 and 5 is sub-optimal as it often generates degenerated results, which we visualize a couple of them in Figure 3

Here, we proposed to employ a few more loss to better regularize the predicted texture offset map and the displacement map. We assume that the offset map should be small and symmetry for most of human faces, therefore $L2$ loss and symmetric loss is imposed:

$$L2 = \frac{1}{n}\sum_{\mathrm{k}=1}^{n}\|M\|^2 \tag{8}$$

$$L_{\mathrm{sym}} = \frac{1}{n}\sum_{\mathrm{k}=1}^{n}\|M - \mathrm{flip}(M)\|^2 \tag{9}$$

where M is the predicted texture offset map or the displacement map. In order to further ensure a smooth reconstruction of the shape and the albedo after adding the sample offset from the prediction results, we apply laplacian filter on the mesh model and also minimize the filtered results:

$$L_{\text{smooth shape}} = \frac{1}{nL}\sum_{\mathrm{k}=1}^{n}\sum_{i=1}^{L}\frac{1}{e(i)}\sum_{j=1}^{e(i)}\|p_i - p_j\|^2 \tag{10}$$

$$L_{\text{smooth albedo}} = \frac{1}{nL}\sum_{\mathrm{k}=1}^{n}\sum_{i=1}^{L}\frac{1}{e(i)}\sum_{j=1}^{e(i)}\|b_i - b_j\|^2 \tag{11}$$

where n corresponds to number of samples in a batch and L is the number of vertices in the face model. $e(i)$ is the set of the neighbour vertices of vertex i. The overall loss is the weighted summation of them:

$$L_{\text{base}} = L_{\text{pixel}} + L_{\text{landmark}} + 20L_{\text{recog}} + (1\text{e}-4)(L2 + L_{\text{sym}}) + 0.08L_{\text{smooth shape}} + 0.1L_{\text{smooth albedo}} \tag{12}$$

We were only able to train the model for 7 epochs since each epoch takes around 9 hours and we only have limited GPU resources. The final results are visualized in Figure 4.

Figure 4: From top to bottom: the visualization of the reconstructed human face, the predicted texture offset map and the predicted displacement map

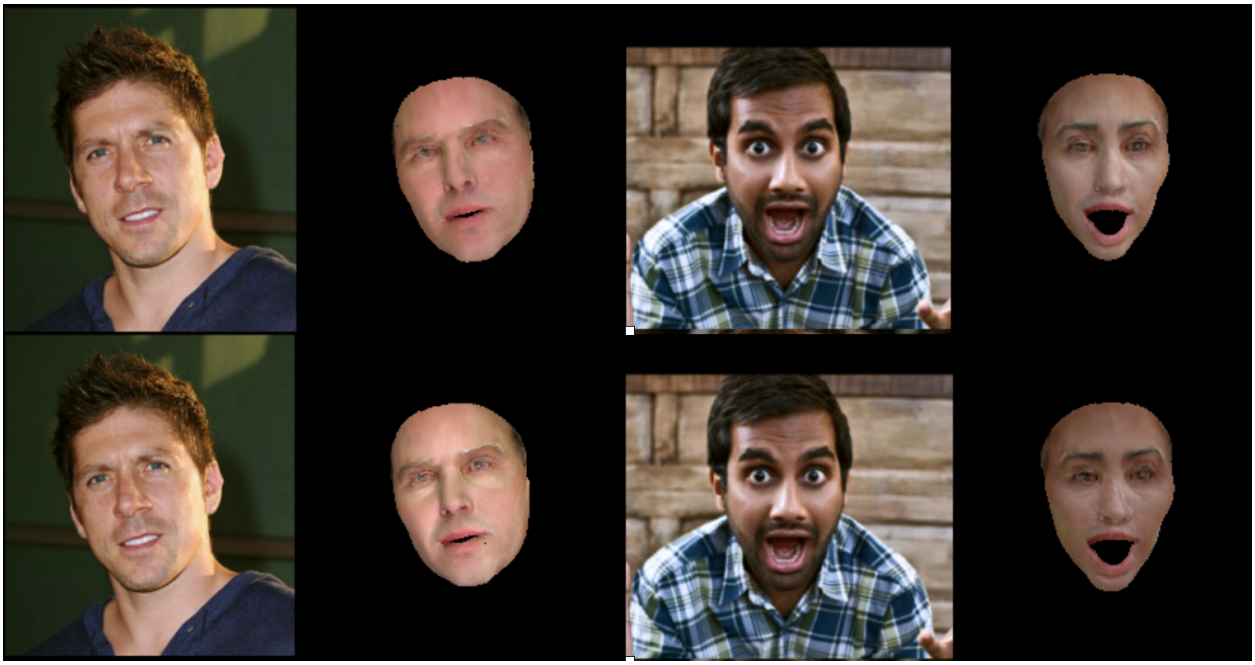

Figure 5: Compared the reconstructed mesh with additional texture offset map and displacement map. Mainly the skin color changes.

2.3 Comparison

Because the limited time we have, we could try any more hyper-parameter settings or training the augmented branch for any more epochs. Based on the current results, the predicted displacement map is almost zero everywhere and mainly modify the lower jaw part. For the texture offset map, it mainly modify the skin color since the 3DMM basis could not perform a well construction for people with dark skin. We visualize the result in Figure 5

References

[1] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Transactions on pattern analysis and machine intelligence*, 25(9):1063–1074, 2003.

[2] V. Blanz, T. Vetter, et al. A morphable model for the synthesis of 3d faces. In *Siggraph*, volume 99, pages 187–194, 1999.

[3] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, 2013.

[4] B.-C. Chen, C.-S. Chen, and W. H. Hsu. Cross-age reference coding for age-invariant face recognition and retrieval. In *European conference on computer vision*, pages 768–783. Springer, 2014.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[7] S. Liu, W. Chen, T. Li, and H. Li. Soft rasterizer: Differentiable rendering for unsupervised single-view mesh reconstruction. *arXiv preprint arXiv:1901.05567*, 2019.

[8] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3d face model for pose and illumination invariant face recognition. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 296–301. Ieee, 2009.

[9] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[10] A. Tewari, F. Bernard, P. Garrido, G. Bharaj, M. Elgharib, H.-P. Seidel, P. Pérez, M. Zollhofer, and C. Theobalt. Fml: face model learning from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10812–10822, 2019.

[11] L. Tran and X. Liu. Nonlinear 3d face morphable model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7346–7355, 2018.

[12] L. Tran and X. Liu. On learning 3d face morphable model from in-the-wild images. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[13] L. Tran, F. Liu, and X. Liu. Towards high-fidelity nonlinear 3d face morphable model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1126–1135, 2019.

[14] X. Zeng, X. Peng, and Y. Qiao. Df2net: A dense-fine-finer network for detailed 3d face reconstruction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2315–2324, 2019.

[15] X. Zhu, X. Liu, Z. Lei, and S. Z. Li. Face alignment in full pose range: A 3d total solution. *IEEE transactions on pattern analysis and machine intelligence*, 41(1):78–92, 2017.