

# Towards High-Fidelity Facial Texture Reconstruction from Multi-View Images Using Self-supervised Learning

Dongjin Huang, Yongjie Xi, Yongsheng Shi and Yufei Liu

Shanghai Film Academy, Shanghai University, 149 Yanchang Road, Jing'an District, Shanghai, China

## Abstract

Deep learning-based monocular 3D face reconstruction methods have been a hot topic in the computer vision community recently. While it is still a great challenge to reconstruct high-fidelity facial texture due to the limitations of low-dimensional subspace of statistical texture model and the occlusion in face region. In this paper, we present an approach to get plausible UV texture with our elaborate Facial Texture Inpainting Module (FTIM) that can inpaint the missing area in the incomplete UV texture map. To ensure the rationality of generated UV texture, we train FTIM with triplets of facial images from different viewpoints. A novel loss is applied to train our framework without ground-truth UV texture maps in a self-supervision way. We provide comprehensive experiments to demonstrate the effectiveness of FTIM and novel loss. Compared with state-of-the-art methods, our method shows better performance in qualitative and quantitative results.

## Keywords

3D face reconstruction, facial texture inpainting, 3DMM, deep learning

## 1. Introduction

In the last decades, 3D face reconstruction from unconstrained 2D images has been a hot research topic in computer vision and computer graphics due to its numerous applications such as facial animation and face recognition. The existing methods of facial texture reconstruction can roughly be into two categories: model-based methods and image-based methods.

Model-based methods depend on the statistical texture model (e.g. 3D Morphable Model [1]). BFM [2] and FLAME [3] are often chosen in 3D face reconstruction due to simplicity and efficiency. The most important problem of model-based methods is that it cannot break through the limitations of low-dimensional subspace of statistical texture model. In addition, the reconstructed UV texture maps from model-based methods are too smooth.

Image-based methods reconstruct 3D facial texture directly from the input image. Specifically, incomplete UV texture can be obtained from the input image. UV-GAN [4] inferred complete UV texture from incomplete UV texture based on deep learning. The quality of the input images, like occlusion and illumination, will greatly affect the generated results and the output of the trained model rely on the training set when image-based methods are adopted.

In this paper, we present an approach that utilizes image-based methods with triplets of facial images, each triplet consists of a nearly frontal, a left-side, and a right-side facial images. We establish the triplets from FaceScape [5], so we can get triplet images that have the same expression and we also propose an improved loss to train our network in the way of self-supervision. The elaborate Facial Texture Inpainting Module (FTIM), which is based on deep learning, can fill the incomplete area and output complete UV map. In addition, multi-view images are only required in the training phase to

---

ICBASE2022@3rd International Conference on Big Data & Artificial Intelligence & Software Engineering, October 21-23, 2022, Guangzhou, China

EMAIL: djhuang@shu.edu.cn (Dongjin Huang); 20723191@shu.edu.cn (Yongjie Xi); yongsheng@shu.edu.cn (Yongsheng Shi); monsantplusone@shu.edu.cn (Yufei Liu)

ORCID: 0000-0002-2437-6657 (Dongjin Huang); 0000-0002-5034-2389 (Yongjie Xi); 0000-0002-9355-2722 (Yongsheng Shi); 0000-0003-0937-3340 (Yufei Liu)



© 2022 Copyright for this paper by its authors.  
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).  
CEUR Workshop Proceedings (CEUR-WS.org)

provide texture aggregation information, and in the test phase only single image is needed. We provide qualitative and quantitative comparisons on CelebA [6] dataset and the results demonstrate our method has better performance than state-of-the-art methods.

## 2. Method

### 2.1. Overview

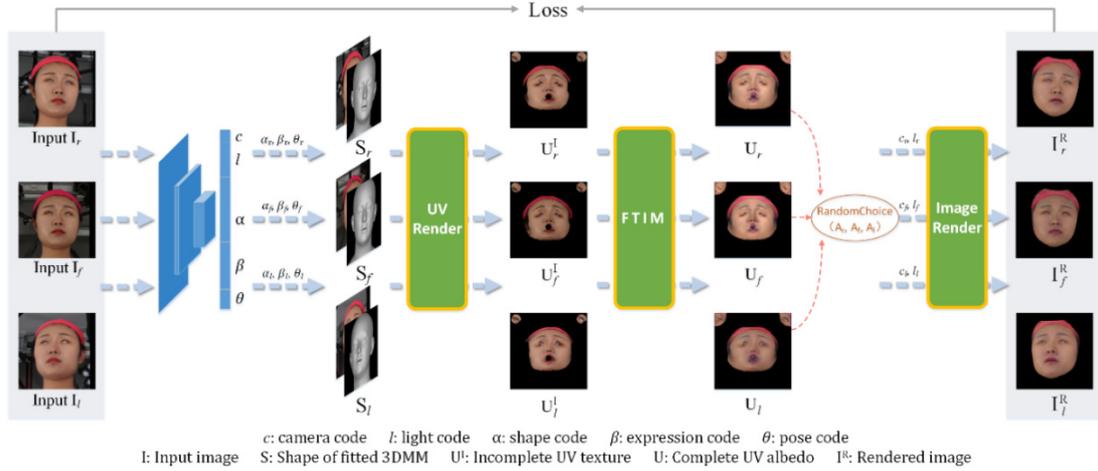


Figure 1: The framework of our proposed network.

As illustrated in Figure 1, our method consists of four modules, namely 3DMM Regressor, UV Render, Facial Texture Inpainting Network, and Image Render. We train our framework with triplets that provide texture information from different viewpoints. During training phase, we first use fixed 3DMM Regressor to get predicted parameters and face meshes. By responding to each face mesh and the input image, we can render their UV texture with the UV Render. Due to self-occlusion, the outputs of UV render are often incomplete. To handle this problem, we propose FTIM that can inpaint the missing area on the UV texture and output complete UV texture maps and UV albedo maps. After then, we randomly select a single UV albedo map from the outputs of the FTIM and render images from different viewpoints with Image Render using the predicted parameters. The reason for random selecting is to avoid dependence on a fixed viewpoint. To train our framework in a self-supervision way, we compute the difference between rendered images and input images as the loss function.

### 2.2. Facial Texture Inpainting Module

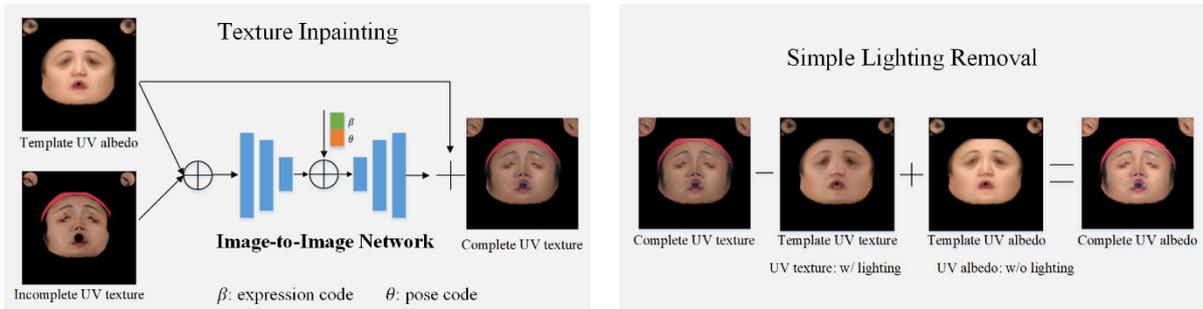


Figure 2: The architecture of FTIM.

we propose the FTIM as illustrated in Figure 2. We feed a template UV albedo map together with the incomplete texture map as input to the FTIM, the template UV texture map can help output canonical UV texture with face semantics. We make a key observation that different expression and pose have an impact on the UV texture, so further add expression and pose parameters to latent space. Finally, we

first subtract the template UV texture map and then add the template albedo map to get the final complete UV albedo map as a simple lighting removal process. We can obtain the template UV texture map from the template albedo map and predicted illumination parameters.

### 2.3. Training Losses

In order to train the FTIM in a self-supervised manner, we propose an improved loss function, which can minimize the difference between the rendered image and the input image. It combines six terms:

$$L = L_{pho} + \lambda_{UV}L_{UV} + \lambda_{sym}L_{sym} + \lambda_{id}L_{id} + \lambda_{mrf}L_{mrf} + \lambda_{reg}L_{reg}, \quad (1)$$

where photometric loss  $L_{pho}$ , UV albedo loss  $L_{UV}$ , identity loss  $L_{id}$ , ID-MRF loss  $L_{mrf}$ , symmetry loss  $L_{sym}$  and regularization  $L_{reg}$ . We use  $\lambda_{pho} = 1.0$ ,  $\lambda_{UV} = 0.2$ ,  $\lambda_{id} = 0.2$ ,  $\lambda_{mrf} = 0.02$ ,  $\lambda_{sym} = 0.25$  and  $\lambda_{reg} = 0.01$  for our results. UV albedo loss, Photometric loss and identity loss are often used in face reconstruction and we further add symmetry loss to provide information when the missing area is huge and ID-MRF loss to output clearer rendered images.

**ID-MRF loss:** Besides identity loss, we also use Implicit Diversified Markov Random Fields (ID-MRF) loss as our perception-level loss to reconstruct details. The ID-MRF can compute the difference between the features patches from different layers of a pretrained network. We compute the loss on layers  $conv3\_2$  and  $conv4\_2$  of VGG19 as:

$$L_{mrf} = \sum_{i \in \{l, f, r\}} (2L_{conv4\_2}(I_i, I_i^R) + L_{conv3\_2}(I_i, I_i^R)), \quad (2)$$

where  $L_{layer}(I_i, I_i^R)$  denotes the ID-MRF loss which is computed on the feature patches extracted from the input  $I$  and the rendered image  $I^R$  with the layer of VGG19. Note that we only compute the  $L_{mrf}$  for face region like  $L_{pho}$ .  $I_l, I_f, I_r$  denote the input images from three viewpoints.

**Symmetry loss:** To add robustness to occlusion, we add the symmetry loss to regularize the invisible face region:

$$L_{sym} = \sum_{i \in \{l, f, r\}} \sum_{j \in U_{proj}} \|A_{i,j} - flip(A_{i,j})\|_1, \quad (3)$$

where  $U_{proj}$  denotes the visible face region in UV space,  $A$  denotes the UV map and  $flip(\cdot)$  is the horizontal flip operation.  $A_l, A_f, A_r$  denote the input images from three viewpoints.

## 3. Results and Analysis

### 3.1. Implementation Details

**Dataset:** We generate our training set from FaceScape dataset, which has multi-view images of an identity that have the same expression and lighting condition. The number of available images reaches to over 400k and we first use FaceNet to eliminate images that don't contain faces. Then a state-of-the-art face pose estimation network is used to divide the images into 3 subsets according to their yaw angles:  $[-90^\circ, -10^\circ]$ ,  $[-10^\circ, 10^\circ]$ ,  $[10^\circ, 90^\circ]$ . For those whose angle of the face is greater than  $90^\circ$  or less than  $-90^\circ$ , we just ignore it. As a result, we can get a total about 120k triplets of faces from the original dataset and select 1.5k triplets as our training set manually. Finally, We align and resize images to  $224 \times 224$  following to fit the 3DMM regressor from DECA. For test set, we randomly select 10K images from the rest of CelebA.

**Implementation details:** Our method is implemented in PyTorch and PyTorch3D for rendering. Adam as the optimizer with a learning rate of  $1e-5$ . The size of the inputs and UV images are  $224 \times 224$  and  $256 \times 256$  respectively.

### 3.2. Qualitative Comparison

We make a qualitative comparison between our method and recent methods and Figure 3 shows our unwarp texture and rendered images from different methods. The UV texture shows our unwarp UV texture from FTIM and the rest of rows shows the input images and the rendered images from our method and others. The methods of DECA [7], Chen [8] and Deng [9] are based on linear texture model and the results of DECA are too smooth and lack of personal identity. While the impressive results of Chen, Deng are better than DECA, but they can't reconstruct details, like wrinkles (column 2, 5). The method of Tran [10] is based on images, but there are artifacts in nose and forehead area in rendered images (column 2, 4). By contrast, our method achieves better results with high-fidelity UV texture.

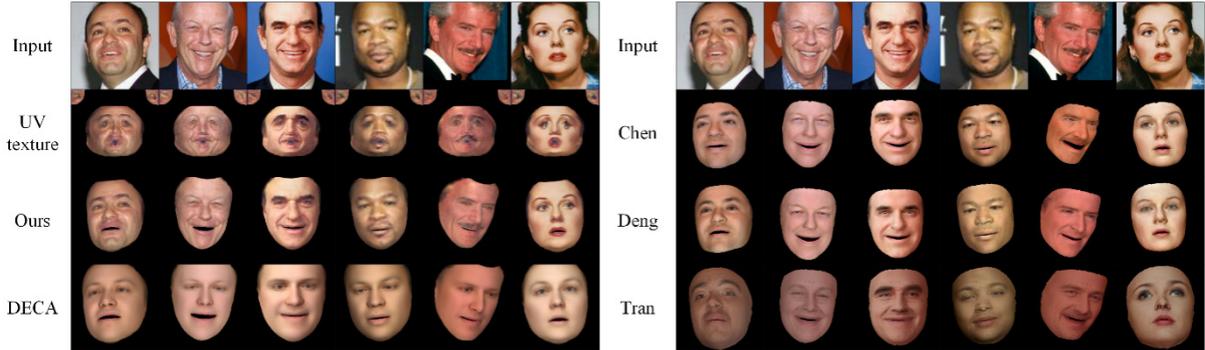


Figure 3: Qualitative comparison on CelebA.

### 3.3. Quantitative Comparison

Table 1

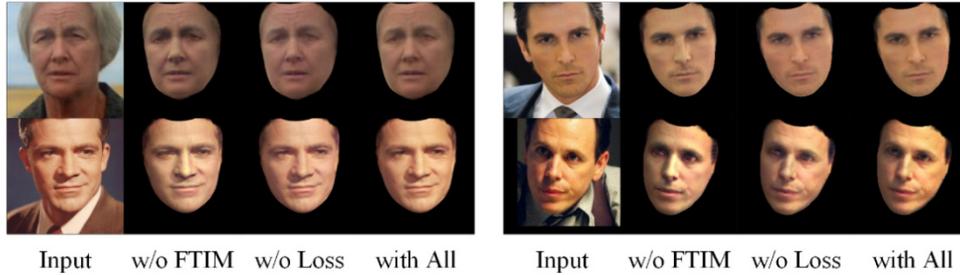
Quantitative comparison on CelebA. The symbol  $\uparrow$  and  $\downarrow$  mean the higher the better and the lower the better respectively.

Method	L1 Distance $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	Cosine Distance $\uparrow$
DECA	0.044	21.741	0.783	0.613
Chen	0.036	22.653	0.854	0.788
Deng	0.030	23.568	0.883	0.781
Tran	0.080	17.575	0.787	0.432
Ours	<b>0.024</b>	<b>27.585</b>	<b>0.928</b>	<b>0.918</b>

We employ L1 distance, the peak signal-to-noise ratio (PSNR), structural similarity index (SSIM) to evaluate our reconstructed face images in 2D image level. Moreover, we also apply cosine distance metric in perception level. In this paper, we utilize FaceNet [11] to extract face features and calculate the cosine distance. All metrics are computed between input images and reconstructed overlays. Table 1 shows the quantitative comparison on CelebA and it shows our method achieves better performance than others on all evaluation metrics.

### 3.4. Ablation Study

To prove the effectiveness of the proposed FTIM and the loss, we conduct two groups of ablation studies as follows.



**Figure 4:** Ablation experiments.

FTIM: Using our FTIM instead of a simple Image-to-Image network can get better results as shown in the second column in Figure 4 and the results demonstrate the contribution of FTIM. The results in the second column are from network trained without FTIM but with all losses and there are artifacts in forehead and nose area. Quantitative comparison results are summarized in Table 2 and we can find it out that all metrics have been improved with FTIM.

**Table 2**

Ablation study on CelebA. The FTIM column indicates using FTIM or not. The  $L_{sym}$ ,  $L_{mrf}$  indicates using the two loss or not.

FTIM	$L_{sym}$ , $L_{mrf}$	L1 Distance↓	PSNR↑	SSIM↑	Cosine Distance↑
		0.036	23.804	0.910	0.885
√		0.027	25.582	0.926	0.904
√	√	<b>0.024</b>	<b>27.585</b>	<b>0.928</b>	<b>0.918</b>

Loss: To evaluate the loss of  $L_{sym}$ ,  $L_{mrf}$  playing an important role in our method, we train our model without them. The results in Figure 4, especially in the nose and eye area, in the fourth column are clearer than those in the third column. Table 2 shows higher accuracy and rendered images are closer to the input images with  $L_{sym}$ ,  $L_{mrf}$ .

## 4. Conclusion

In this paper, we propose a method with FTIM that can generate more realistic UV texture map to improve the quality of 3D facial texture reconstruction. To go beyond the limitation of lacking of ground-truth UV texture maps, we introduce an improved loss to train our model with multi-view triplets of images, which can provide missing face region information from other perspectives. The results of ablation study and comparative experiment show the effectiveness of FTIM and our loss and demonstrate our method achieves better performance compared to other recent methods. In the future, we will learn to reconstruct detailed facial geometry with our reconstructed UV texture.

## 5. References

- [1] V. Blanz and T. Vetter. "A morphable model for the synthesis of 3d faces." Annual Conference on Computer Graphics and Interactive Techniques (Proc.SIGGRAPH), pp. 187–194, 1999.
- [2] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. "A 3d face model for pose and illumination invariant face recognition." IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS), pp. 296–301, IEEE, 2009.
- [3] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. "Learning a model of facial shape and expression from 4d scans." ACM Transactions on Graphics (SIGGRAPH Asia), vol. 36, no. 6, pp. 194–1, 2017.
- [4] J. Deng, S. Cheng, N. Xue, Y. Zhou, and S. Zafeiriou. "UV-GAN: Adversarial facial uv map completion for pose-invariant face recognition." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7093–7102, 2018.

- [5] H. Yang, H. Zhu, Y. Wang, M. Huang, Q. Shen, R. Yang, and X. Cao. "Facescape: a large-scale high quality 3d face dataset and detailed riggable 3d face prediction." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 601–610, 2020.
- [6] Z. Liu, P. Luo, X. Wang, and X. Tang. "Deep learning face attributes in the wild." International Conference on Computer Vision (ICCV), pp. 3730–3738, 2015.
- [7] Y. Deng, J. Yang, S. Xu, D. Chen, Y. Jia, and X. Tong. "Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set." IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW), pp. 285–295, 2019.
- [8] Y. Feng, H. Feng, M. J. Black, and T. Bolkart. "Learning an animatable detailed 3d face model from in-the-wild images." ACM Transactions on Graphics (TOG), vol. 40, no. 4, pp. 1–13, 2021.
- [9] Y. Chen, F. Wu, Z. Wang, Y. Song, Y. Ling, and L. Bao. "Self-supervised learning of detailed 3d face reconstruction." Transactions on Image Processing (TIP), vol. 29, pp. 8696–8705, 2020.
- [10] L. Tran, F. Liu, and X. Liu. "Towards high-fidelity nonlinear 3d face morphable model." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1126–1135, 2019.
- [11] F. Schroff, D. Kalenichenko, and J. Philbin. "Facenet: A unified embedding for face recognition and clustering." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 815–823, 2015.