Towards Realistic Generative 3D Face Models [Supplementary]

Aashish Rai[1] Hiresh Gupta[*1] Ayush Pandey[*1] Francisco Vicente Carrasco[1]
Shingo Jason Takagi[2] Amaury Aubel[2] Daeil Kim[2] Aayush Prakash[2]
Fernando De la Torre[1]

[1]Carnegie Mellon University [2]Meta Reality Labs

https://aashishrai3799.github.io/Towards-Realistic-Generative-3D-Face-Models

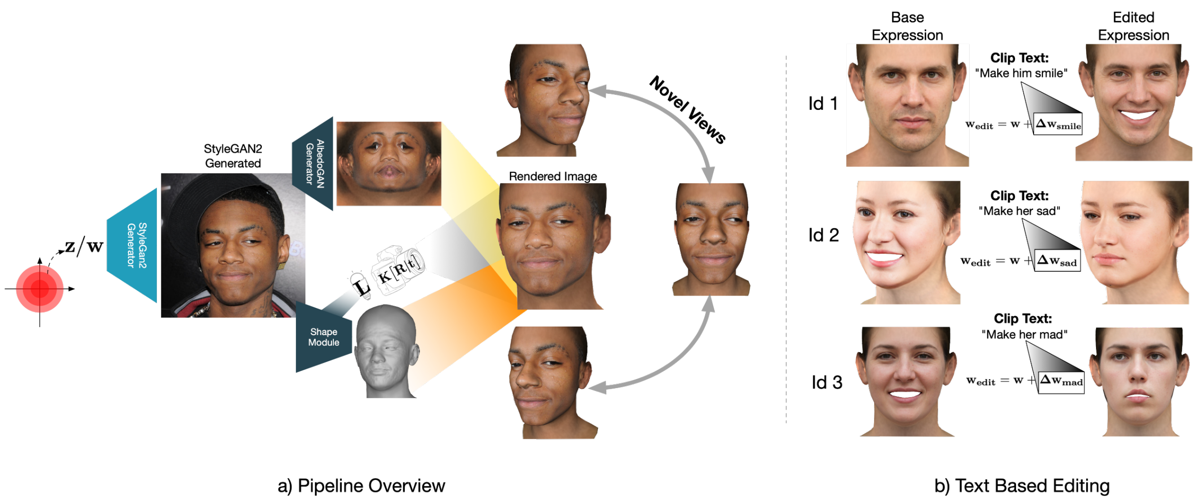

Figure 1. 3D generative face model. a) High-resolution 3D shape and albedo recovered from a StyleGAN2-generated image. Novel views can be rendered using the estimated face model. b) Editing of 3D faces with text. This method allows for 3D expression manipulation through guidance with the CLIP model.

A. Appendix 1

To compare the performance of our method along various poses, we calculated the ID loss between the GT-2D image and rendered face generated from various methods. We randomly sampled $10K$ $z \in \mathbf{R}^{512}$ and generated 2D images using the StyleGAN2 generator and corresponding mesh and albedo from our pipeline. We used the $2D$ images to reconstruct rendered $3D$ face using all other methods. Then we computed the ID loss between the 2D face and the rendered face for the given poses from 45° to 135° (90° being the frontal pose). The observations are shown in Fig. 2. As can be seen from the figure, our method not only performs better in capturing the ID for all the poses but also produces less variability. Hence, supporting our claim of better pose-invariance in texture generation.

B. Appendix 2 (Visualizations)

In this section, we show more results of our pipeline.

* equal contribution

B.1. Sample mesh generated from our model and the corresponding rendered images

Fig. 3 shows the reconstructed mesh and rendered images for randomly sampled $w \in \mathrm{R}^{18\times512}$.

B.2. Pose control

One of the benefits of having 3D representations of faces is direct control over pose and illumination. Fig. 5 shows the pose variation of 3D faces generated from our method. Note despite having non-frontal 2D images, our method can generate complete texture from the corresponding $z \in \mathrm{R}^{512}$, producing good rendered images at all angles. Watch ”teaser.mp4” in supplementary for pose videos.

B.3. Relighting

Fig. 6(a) shows a 3D face model produced by AlbedoGAN lighted with different environment maps. Fig. 6(b) shows two AlbedoGAN 3D face models relighted using the Blinn-Phong shading model [2]. Please, see the illumination videos in the supplementary material.

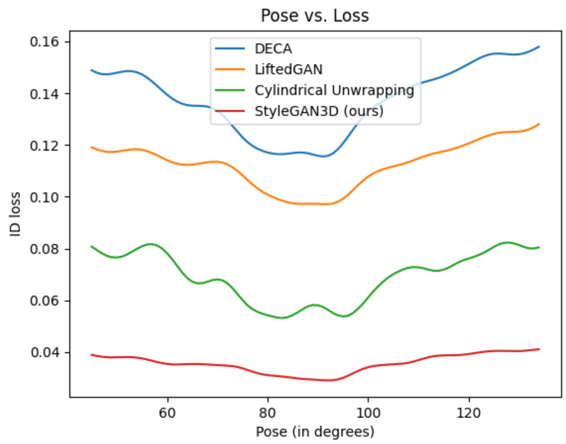

Figure 2. Pose vs. ID loss. 90°is the frontal pose. As can be inferred from the graph, there's a change of $19.4\%, 29.3\%, 18.1\%$ in ID loss between frontal pose and side pose of LiftedGAN [5], DECA [3], and unwrapped texture against 7.5% for our model. Note that the cylindrical unwrapped texture is obtained by orthogonal projection of a mesh [4] fitted on the given 2D image.

Algorithm 1 Training AlbedoGAN to generate albedo from a latent code

Require: N: number of iterations, G_{StlGAN}: pretrained StyleGAN2 generator, 3DMM, G_{Ab}: AlbedoGAN generator, D_{real}: discriminator, and D_{id}: identity discriminator

for $i \leftarrow 1$ to N **do**
    Generate a face image $\mathbf{I}_{StlGAN}$ using StyleGAN2 by randomly sampling a latent code $\mathbf{w}$
    Project the 3DMM onto $\mathbf{I}_{StlGAN}$ to obtain a UV texture orthogonally
    Map input image pixels to UV domain, perform barycentric interpolation to fill missing pixels
    Project flipped $\mathbf{I}_{StlGAN}$ onto the fitted mesh to collect pixels for occluded areas, blend them with original texture
    Extract albedo $\mathbf{A}_{GT}$ and shading map $\mathbf{S}$ from unevenly illuminated texture using SH model
    Generate albedo $\mathbf{A}_{abGAN}$ using AlbedoGAN and same $\mathbf{w}$.
    Train AlbedoGAN G_{Ab} to minimize difference between $\mathbf{A}_{abGAN}$ and $\mathbf{A}_{GT}$. Use D_real and D_id to penalize G_{Ab}
end for
return trained AlbedoGAN G_{Ab} from the final iteration

B.4. Mesh and Rendered comparison with SOTA

Fig. 4 shows the comparison between our method and various other SOTA methods for 3D face reconstruction.

Figure 3. Randomly generated coarse mesh, detailed mesh and rendered faces from our model, corresponding to input face.

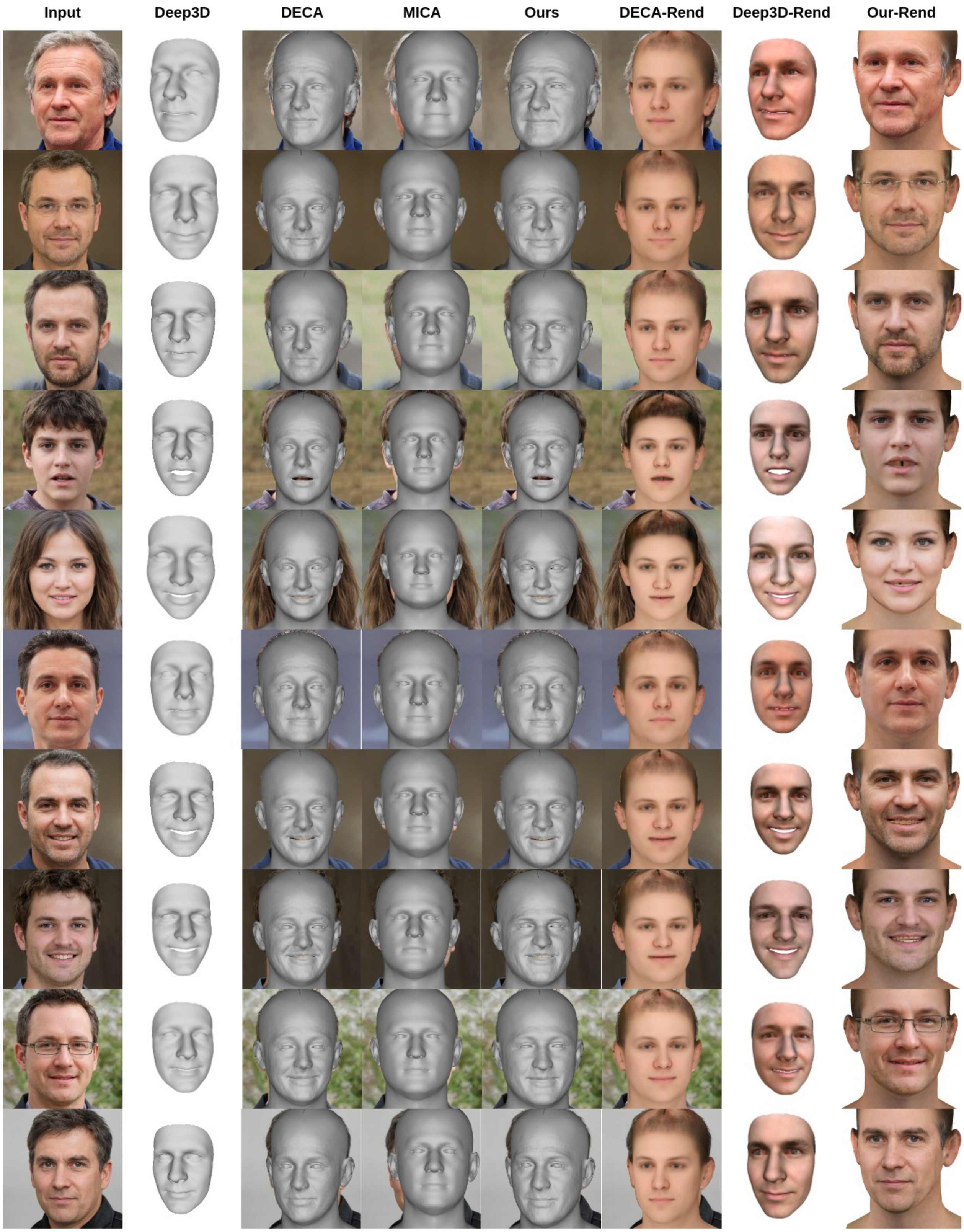

Figure 4. Comparison of reconstructed mesh and rendered faces of our method against Deep3DFaceReconstuction [1], DECA [3], and MICA [6].

Figure 5. Generated albedo and corresponding rendered images at various poses for given input image. Interesting to see our method can generate a complete, symmetrical albedo for non-frontal input images without creating artifacts.

Figure 6. a) Environment maps renderings; b) Relighting example using point lights and Blinn–Phong shading model.

References

[1] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *IEEE Computer Vision and Pattern Recognition Workshops*, 2019. 4

[2] John et al. The mathematics of lighting. In *OpenGL Programming Guide: The official guide to learning OpenGL, version 4.5 with SPIR-V*, 2016. 1

[3] Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. Learning an animatable detailed 3D face model from in-the-wild images. In *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, volume 40, 2021. 2, 4

[4] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017. 2

[5] Yichun Shi, Divyansh Aggarwal, and Anil K Jain. Lifting 2d stylegan for 3d-aware face generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6258–6266, 2021. 2

[6] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Towards metrical reconstruction of human faces. In *European Conference on Computer Vision*, 2022. 4