Mimic3D: Thriving 3D-Aware GANs via 3D-to-2D Imitation – Supplementary Material

Xingyu Chen* Yu Deng* Baoyuan Wang
Xiaobing.AI

I. More Implementation Details

I.1. Network Structure

Figure IV illustrates our network designs, including the 3D super-resolution module $\mathcal{S}^{3D}$ and the 3D-aware block in the tri-plane generator $\mathcal{E}$.

For $\mathcal{S}^{3D}$ (Fig. IV(a)), we use two modulated 2D convolution blocks [4] to upsample the tri-planes.

For the 3D-aware block (Fig. IV(b)), we re-organize the tri-planes according to Fig. 4 in the main text, and apply modulated 2D convolutions for each of the three planes. We use different affine layers to generate style codes for the three modulated convolutions, respectively.

I.2. Training Details

We randomly sample latent code z from the normal distribution and camera pose $\boldsymbol{\theta}$ from those of the training datasets to synthesize fake images, following EG3D [1]. For each viewing ray, we sample 96 points to calculate the volume rendering equation, including 48 points with stratified sampling and 48 points with importance sampling. The learning rates of the generator and the two discriminators are set to 0.0025 and 0.002, respectively. We train the 2D branch with 25M images in total, and then jointly train the whole framework with additional 15M images. The batch size during training is set as 32. Other training settings are identical to those of EG3D [1].

I.3. Patch Scale

To reduce GPU memory costs and enable training at high resolution, we render 64^2 patches for the 3D-to-2D imitation. Thus, the patch scale is $1/4$ or $1/8$ of the whole image for the 256^2 or 512^2 experiments, respectively. The patch center is uniformly sampled from the whole image space.

I.4. The necessity of 2D super-resolution module

The function of the 2D super-resolution in the 2D branch is to provide stable and high-quality guidance for the 3D branch. Previous studies have attempted to directly learn in 3D space without 2D super-resolution via the adversarial loss. However, due to the restriction of modern GPU memory, they either adopted more efficient 3D representations (*e.g.*, radiance manifolds [2] or MPI [7]) or used patch-wise loss (*e.g.*, EpiGRAF [6]), yet these strategies often lead to worse diversity and image quality due to the instability of the GAN loss. By contrast, our imitation with the 2D branch via LPIPS loss provides stable gradients for learning the 3D representation, and thus supports patch-wise training without sacrificing the generation quality, which is the key to our superior results. Furthermore, our strategy also avoids troublesome training tricks (*e.g.*, the annealed strategy in EpiGRAF [6]) thus easier to be adapted to other frameworks.

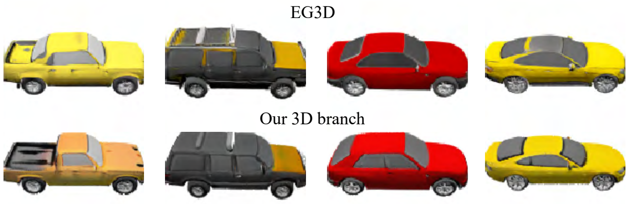

Figure I. Comparison with EG3D on ShapeNet-Cars.

I.5. Training time/memory of 3D-to-2D imitation

Our method requires 31 GB memory at 256^2 resolution with a batch size of 32 when trained on 8 GPUs, compared to 27 GB memory without the 3D-to-2D imitation. Also, our training time is 1.5 times longer than that of EG3D.

II. More Results and Comparisons

II.1. End-to-end 3D-to-2D imitation learning

Our initial motivation for the two-stage training is to leverage the powerful prior of an existing 2D generator (with 2D super-resolution) to guide our 3D branch. In fact, the overall framework (including both 2D and 3D branches) can be trained end-to-end from scratch. We conduct a simple experiment on FFHQ at 256^2 with identical hyper parameters as described in the main paper and achieve an FID of 5.03 for the 3D branch, which is comparable to the two-stage training result.

*These authors have contributed equally to this work.

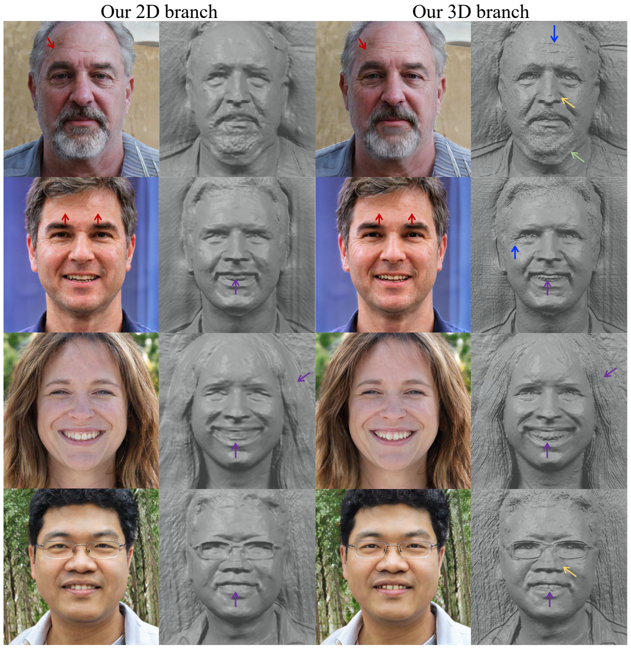

Figure II. Comparison of 2D and 3D branches. (Zoom in for better visualization.)

II.2. More results on faces

Figures V and VI illustrate more visual comparisons. Compared to EG3D [1], we have more detailed geometry and smoothly tilted strips in spatiotemporal texture images, indicating better 3D consistency. Similar to ours, EpiGRAF and GMPI also generate high-resolution images via direct rendering. Yet, we have superior image quality as shown in Fig. VI.

Figures VII and VIII show more of our results on FFHQ and AFHQ -v2 Cats datasets, respectively.

Referring to the supplemental video for animations.

II.3. Results on general objects.

Our method can handle general objects with wider range of camera views. In Fig. I, we compare our 3D branch with EG3D on ShapeNet-Cars (128^2) and achieve comparable image generation quality.

II.4. Comparison of our 2D and 3D branches

Our 3D branch can generate fine details comparable to the 2D branch. In Fig. II (red arrows), we show details produced by the 3D branch that are not visible in the 2D branch.

Our 3D branch clearly produces finer geometry details compared to the alternatives with 2D super-resolution (see Fig. II). As shown, the finer geometry details are not random noises but features of hair, teeth, wrinkles, etc (purple arrows). Furthermore, we can generate diverse nose shapes (yellow arrows), complex jaws with beards (green arrows), and wrinkles (blue arrows) on the geometries.

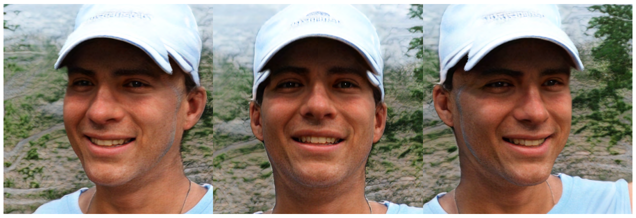

Figure III. Failure case.

III. Limitations and Future Works

We thoroughly discuss the limitations of our method and possible future improvements.

First, our learned 3D branch still has inferior image quality in terms of FID compared to the 2D branch. This may come from the current design of the 3D super-resolution module and the learning strategy. Specifically, our 3D super-resolution module adopts a similar structure to that of the 2D branch in order for a fair comparison, which may not be the optimal solution. More advanced structures, including leveraging 3D-aware convolutions could be further explored for better 3D super-resolution. Besides, the LPIPS loss during 3D-to-2D imitation leverages a pre-trained VGG network which is trained on images of 224^2 resolution. It may not well capture the perceptual information of a small image patch. Leveraging more recent pre-trained models [3, 5] or even multiple feature extractors could be a possible choice. Exploring better discriminators for the patch-level adversarial loss in the 3D branch could also benefit the training process.

Second, our method can produce incorrect geometries in certain cases. As shown in Fig. III, a typical failure case is geometry discontinuity, where the face region is not smoothly connected with the head region, leading to obvious artifacts at side views. These artifacts also occur in the original EG3D. We believe this problem can be alleviated by introducing more profile images for training, as currently the training data are mostly frontal images so that the planes for depicting side-view features may not be well-trained. In addition, certain generated geometry structures such as hairs and cat whiskers are stuck to the surfaces instead of correctly floating in the volumetric space, as shown in Fig. VII and VIII. We conjecture this is due to that the random sampling strategy with limited queries during volume rendering is hard to model thin structures, as also indicated by a previous method [2]. Therefore, a more advanced 3D representation that could efficiently capture these complex structures is worthy of ongoing exploration.

Finally, our training strategy also requires training the 2D branch in advance, which increases the overall training time compared to the original EG3D. A possible way to reduce the training time is to jointly train the 2D and 3D branches from scratch. We leave it for our future work.

References

[1] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3D generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2, 4

[2] Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. GRAM: Generative radiance manifolds for 3D-aware image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2

[3] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 2

[4] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of styleGAN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 1

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, 2021. 2

[6] Ivan Skorokhodov, Sergey Tulyakov, Yiqun Wang, and Peter Wonka. EpiGRAF: Rethinking training of 3D GANs. In *Proceedings of the Advances in Neural Information Processing Systems*, 2022. 1, 5

[7] Xiaoming Zhao, Fangchang Ma, David Güera, Zhile Ren, Alexander G Schwing, and Alex Colburn. Generative multiplane images: Making a 2D GAN 3D-aware. In *Proceedings of the European Conference on Computer Vision*, 2022. 1, 5

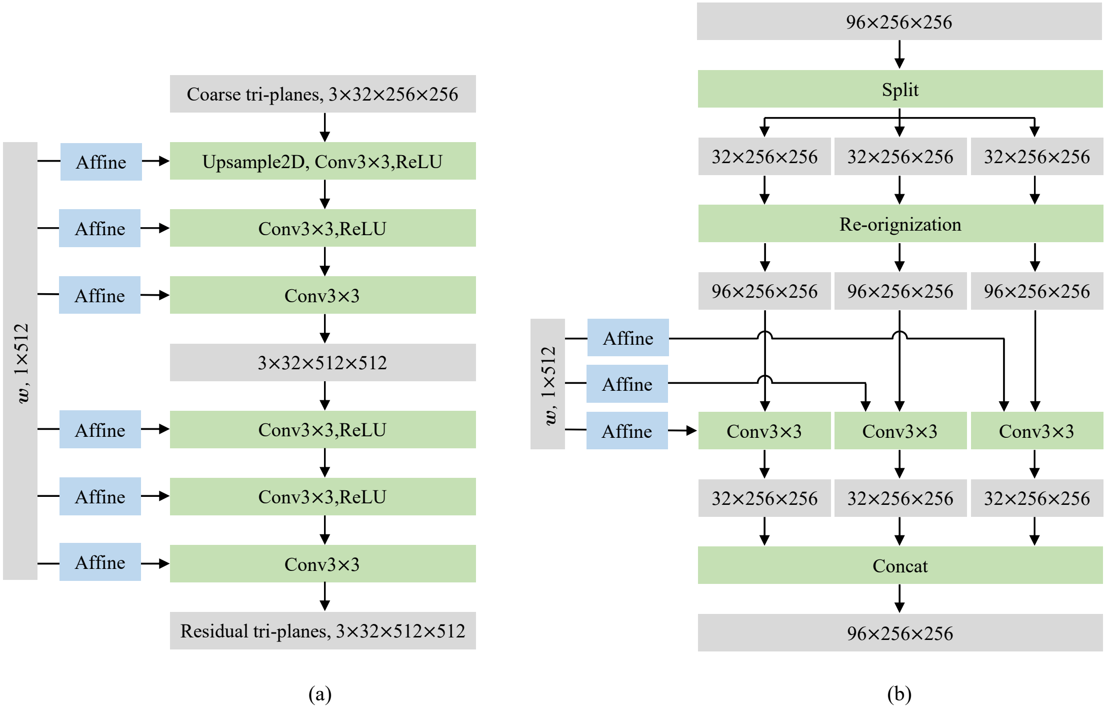

Figure IV. Network designs. (a) 3D super-resolution module $\mathcal{S}^{3D}$. (b) 3D-aware block.

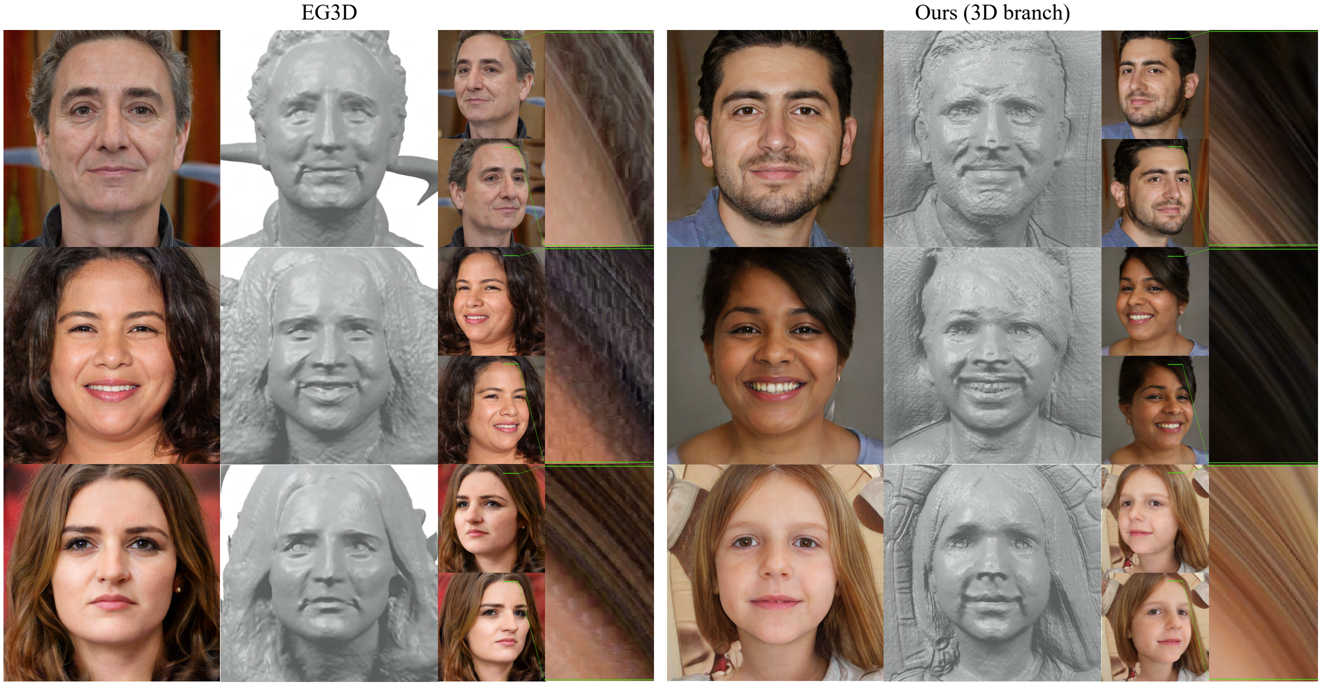

Figure V. Comparison w/ EG3D [1]. Our method generates images with comparable quality to those of EG3D, while producing 3D geometries with finer details and multiview sequences with better 3D-consistency. Referring to the supplemental video for animations.

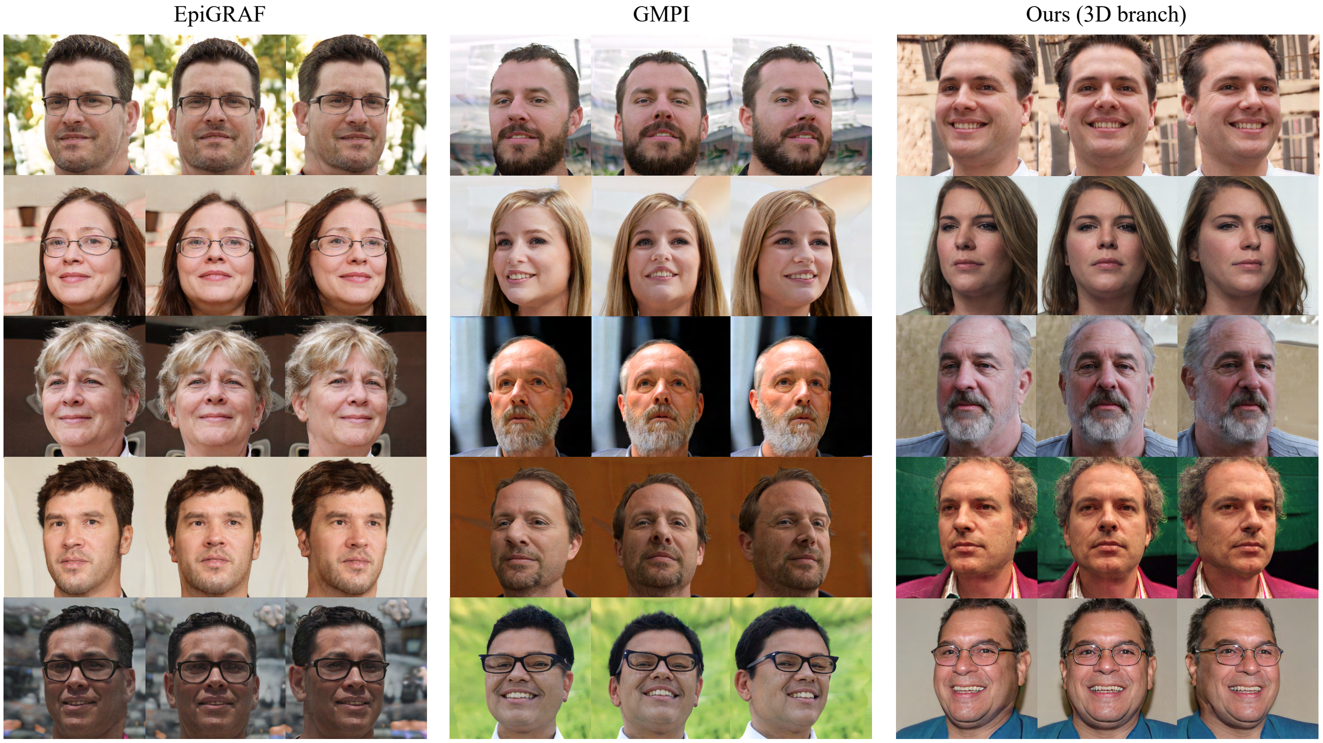

Figure VI. Comparison w/ EpiGRAF [6] and GMPI [7] . Referring to the supplemental video for animations.

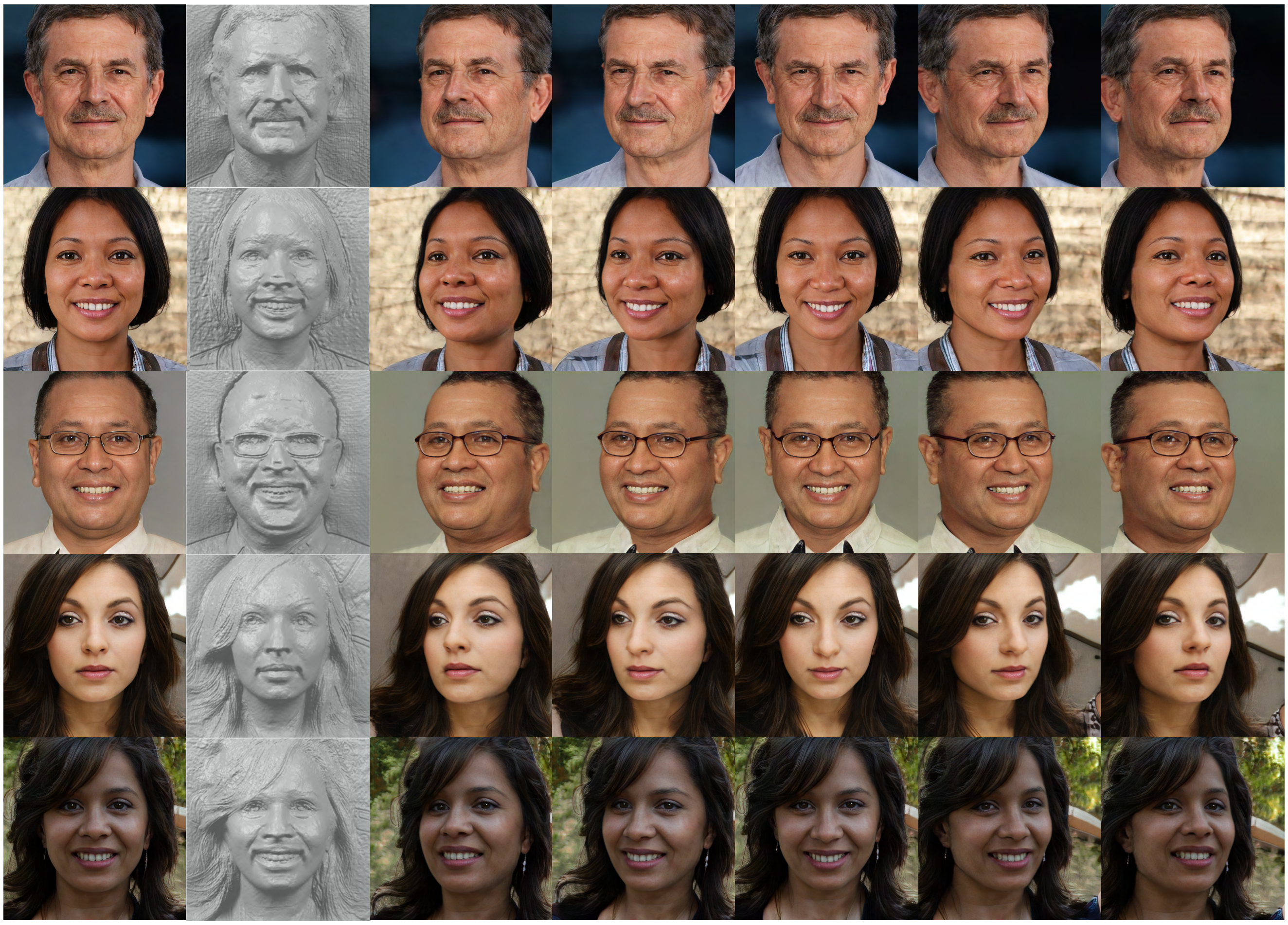

Figure VII. Our results on FFHQ dataset. Referring to the supplemental video for animations.

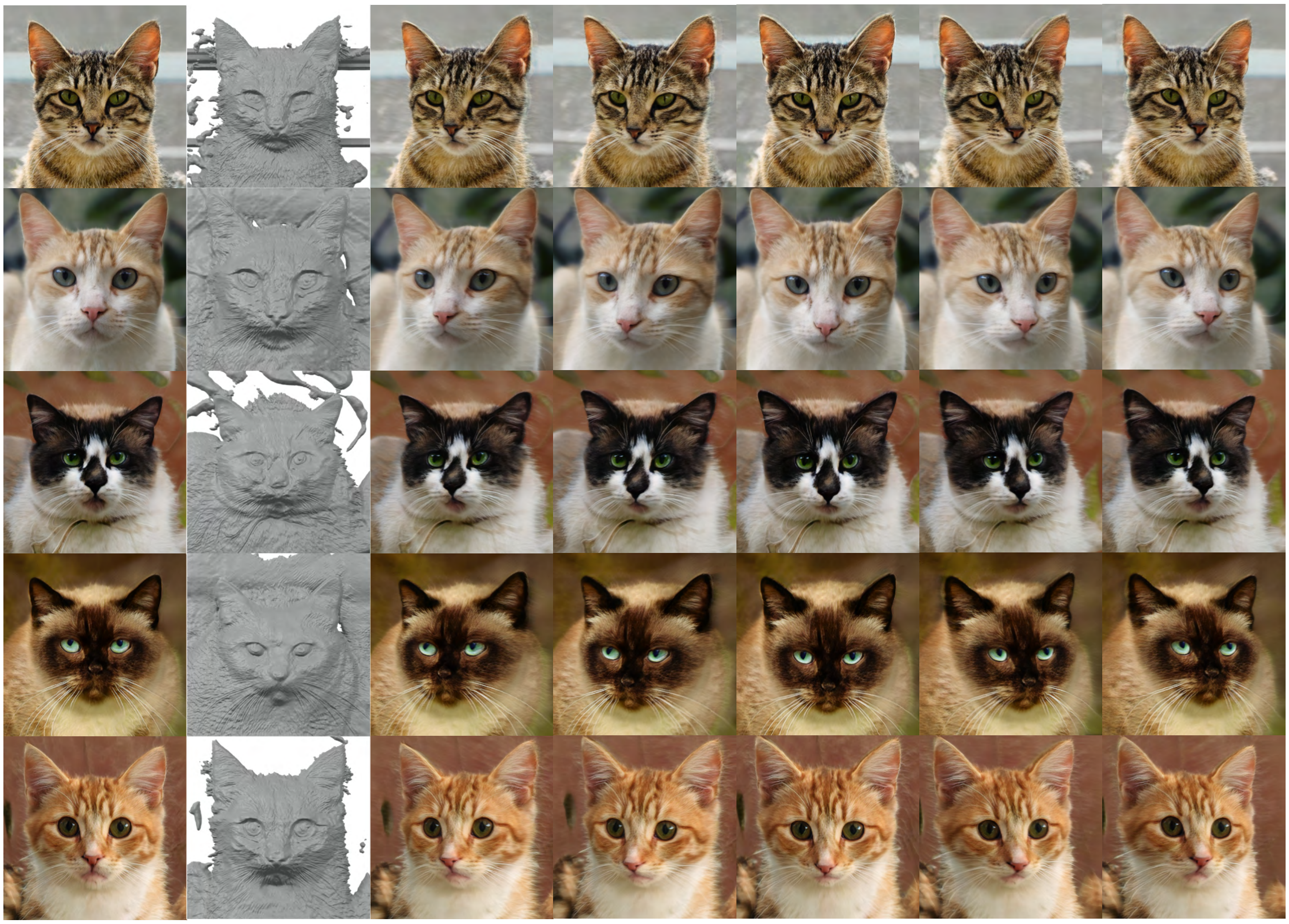

Figure VIII. Our results on AFHQ-v2 Cats. Referring to the supplemental video for animations.