RODIN: A Generative Model for Sculpting 3D Digital Avatars Using Diffusion Supplementary Material

Appendix

A. Background of Diffusion Models

Diffusion models produce data by reversing a gradual noising process. The forward noising process is a Markov chain that corrupts the data by gradually adding random noises for steps $t = 1, \cdots, T$. Each step in the forward process is a Gaussian transition $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := \mathcal{N}(\sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t\boldsymbol{I})$, where $\{\beta_t\}_{t=0}^T$ are usually predefined variance schedule. Furthermore, the noisy latent variable $\boldsymbol{x}_t$ can be derived from $\boldsymbol{x}_0$ directly as:

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\boldsymbol{x}_0 + \sqrt{1-\alpha_t}\boldsymbol{z}, \quad \boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \tag{1}$$

where $\alpha_t := \prod_{s=1}^t (1-\beta_s)$. When T is large enough, α_T gets closer to 0 and the last latent variable $\boldsymbol{x}_T$ is nearly an isotropic Gaussian distribution.

To sample data from the given distribution, we can reverse the noising process by learning a denoising model $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$. The denoising model $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$ starts from the Gaussian noise $\boldsymbol{x}_T$ and iteratively reduces the noise for $t = T-1, \cdots, 0$. Specifically, it takes the noisy latent variable $\boldsymbol{x}_t$ at each timestep t and predicts the corresponding noise $\boldsymbol{\epsilon}$ with a minimal mean square error:

$$\min_\theta \mathbb{E}_{\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0), \boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), t} ||\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{z}||_2^2. \tag{2}$$

With the learned denoising model, the data can be sampled with the following reverse diffusion process:

$$\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\left(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\right) + \sigma_t\boldsymbol{z}, \tag{3}$$

where $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is a randomly sampled noise, and σ_t is the variance of the added noise.

B. Implementation Details

B.1. Architectural Design and Training Details

Our base diffusion model adopts the U-Net architecture from [3] with a channel number of 192, while we make several modifications including tri-plane roll-out and 3D-aware convolution, as discussed in Section **??**. To orchestrate the tri-plane generation and enable semantic editing, we also introduce a condition encoder, a fixed CLIP ViT-B/32 image encoder, to map a reference image to a semantic latent vector. The upsample diffusion model is a U-Net-like model but we apply only one upsample layer that directly upscales the feature maps from 64 to 256 for efficiency, as shown in Figure 1. The tri-plane roll-out and 3D-aware convolution are utilized in each residual block. When training the upsample model, we apply condition augmentation on the tri-planes to reduce the domain gap as described in Section **??**. Specifically, we degrade the ground-truth 256×256 tri-planes with a random combination of downscale, Gaussian blur, and Gaussian noise.

We utilize AdamW optimizer [4] with a batch size of 48 and a learning rate of $5e$-5 for the base diffusion model, and with a batch size of 16 and a learning rate of $5e$-5 for the upsample diffusion model. We also apply the exponential moving average (EMA) with a rate of 0.9999 during training. We set the diffusion steps as 1,000 for the base model, and 100 for the upsample model, with a linear noise schedule. During inference we sample 100 diffusion steps for both the base model and the upsample model. All the experiments are performed on NVIDIA Tesla 32G-V100 GPUs.

B.2. Tri-plane Fitting

Our framework learns the 3D avatar generation from explicit 3D representations obtained from fitting multi-view images. However, a multi-view consistent, diverse, high-quality and large-scale dataset of face images is difficult and expensive to collect. Images collected from the Web have no guarantee of multi-view consistency and suffer privacy and copyright risks. Regarding this, we turn to synthetic techniques that can randomly render novel 3D portraits by randomly combining assets manually created by artists. We leverage the Blender synthetic pipeline [7] that generates human faces along with random sampling from a large collection of hair, clothing, expression and accessory. Hence, we create 100K synthetic individuals independently and for each render 300 multi-view images with a resolution of 256×256.

For tri-plane fitting, we learn $256 \times 256 \times 32 \times 3$ spatial features for each person along with a lightweight MLP

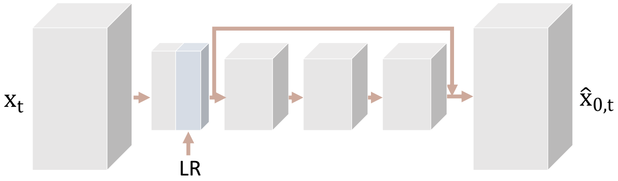

Figure 1. Architecture of the upsample diffusion model.

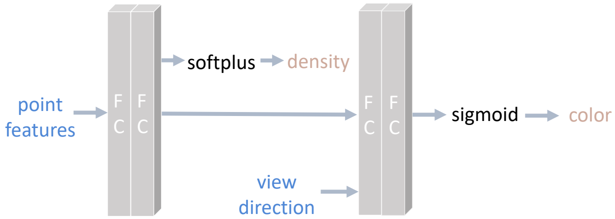

Figure 2. Architecture of the MLP decoder.

decoder consisting of 4 fully connected layers as shown in Figure 2. We randomly initialize the tri-plane feature and MLP weights. During fitting, we apply random rescaling (downsample to a resolution in $[64, 256]$ followed by an upsampling to 256) to ensure that the MLP decoder is robust to multi-resolution tri-plane features. To enable scalable and efficient fitting, we first optimize the shared 4-layer MLP decoder when fitting the first 1,000 subjects, and this decoder is fixed when fitting the following subjects. Thus different subjects are fitted separately in distributed servers.

For multi-view images $\{\boldsymbol{x}\}_{N_v}$ for the given subject, where $\boldsymbol{x} \in \mathbb{R}^{H_0 \times W_0 \times 3}$, we minimize the mean squared error $\mathcal{L}_{\text{MSE}}$ between the rendered image via volumetric rendering, *i.e.*, $\hat{\boldsymbol{x}} = \mathcal{R}(\boldsymbol{c}, \sigma)$ and the corresponding ground truth image. Moreover, we introduce additional regularizers to improve the fitting quality. To be specific, we manage to reduce the "floating" artifact by enforcing the sparsity loss $\mathcal{L}_{\text{sparse}}$ which penalizes the ℓ_1 magnitude of the predicted density, the smoothness loss $\mathcal{L}_{\text{smooth}}$ [2] that encourages a smooth density field, as well as the distortion loss $\mathcal{L}_{\text{dist}}$ [1] that encourages compact rays with localized weight distribution.

B.3. Text-based Avatar Customization

As shown in Section **??**, the RODIN model can edit generated avatars with text prompts. For a generated avatar with a conditioned latent $\boldsymbol{z}_i$, we can obtain an editing direction $\boldsymbol{\delta} = E_T^{clip}(T_{tgt}) - E_T^{cilp}(T_{src})$ in the text embedding space of CLIP based on prompt engineering. For instance, we can choose the source text T_{src} from some general descriptions such as "a photo of a person" and "a portrait of a person", and use the target text T_{tgt} such as "a photo of a person with blond hair" and "a photo of a smiling person ". As we assume colinearity between the CLIP's image and text embedding, we can obtain the manipulated embedding as $\boldsymbol{z}_i + \boldsymbol{\delta}$, which is used to generate edited avatars.

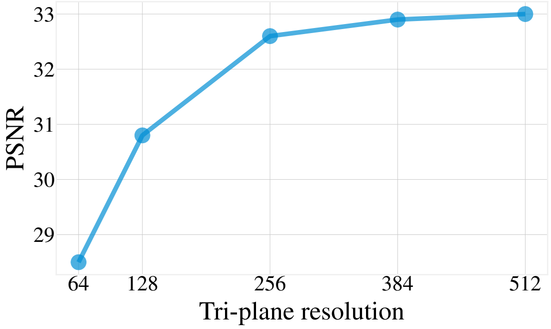

Figure 3. Effect of tri-plane resolution for tri-plane fitting.

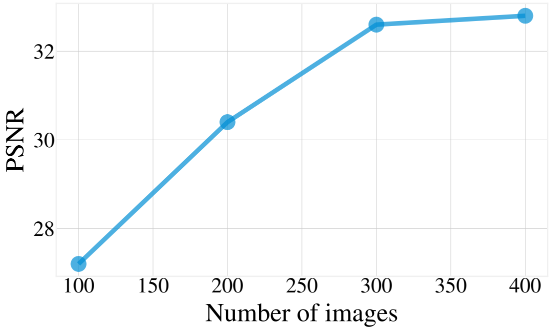

Figure 4. Effect of image numbers for tri-plane fitting.

B.4. Latent Diffusion for Unconditional Sampling

As discussed in Section **??**, our base diffusion model supports both unconditional generation and conditional generation. To account for full diversity during unconditional sampling, we additionally train a diffusion model to model the distribution of the latent $\boldsymbol{z}$. The latent diffusion adopts a 20-layer MLPs network [5] with the hidden channel of 2048 that iteratively predicts the latent code $\boldsymbol{z} \in \mathbb{R}^{512}$ from random Gaussian noise. We set the diffusion steps as 1,000 with a linear noise schedule. We utilize AdamW optimizer with a batch size of 96 and a learning rate of $4e-5$, and also apply exponential moving average (EMA) with a rate of 0.9999 during training.

B.5. Text-to-avatar Generation

As shown in Section **??**, we perform text-to-avatar generation by training a text-conditioned diffusion model that generates an image embedding from a text embedding in the CLIP space. We adopt the network architecture from [6] and train it on a subset of the LAION-400M dataset, containing 100K portrait-text pairs. We set the diffusion steps as 1,000 with a linear noise schedule. We utilize AdamW optimizer with a batch size of 96 and a learning rate of $4e-5$, and also apply exponential moving average (EMA)

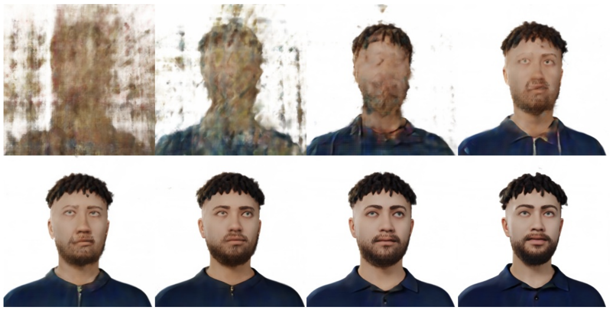

Figure 5. Visualization of intermediate generation results of different time steps.

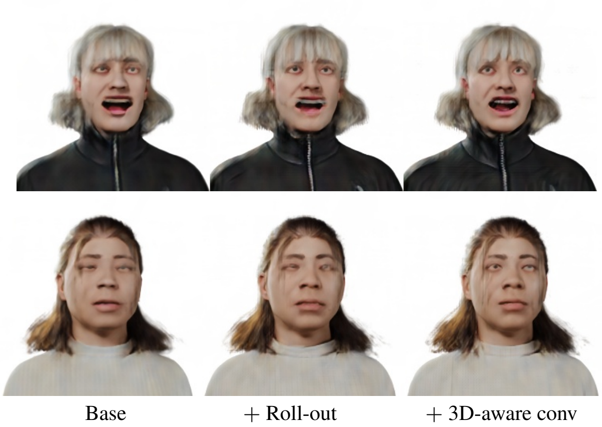

Figure 6. Both tri-plane roll-out and 3D-aware convolution are crucial for high-fidelity results.

with a rate of 0.9999 during training.

C. Additional Ablation Study and Analysis

C.1. Tri-plane Settings

Choices of Tri-plane resolution. To analyze the impact of tri-plane resolution, we experiment with different tri-planes from a set of $\{64, 128, 256, 384, 512\}$ to fit 1024×1024 images and show the results in Figure 3. Overall, the fitting quality increases with the tri-plane resolution. Empirically, we find that the 256×256 tri-plane is strong enough to represent a subject. Considering the memory cost, we thus choose to utilize 256×256 tri-planes in our experiments.

Number of images for fitting. We also explore how many images are needed to achieve a high-quality fitting. As shown in Figure 4, the fitting quality get almost saturated when using 300 different views for the neural tri-plane reconstruction.

C.2. Visualization of Different Diffusion Steps

Diffusion models generate samples by gradually removing noises for $t \in [T, 0]$, and analyzing these intermediate results would reach an in-depth understanding of the generation process. We thus demonstrate the generated results over the reverse process in Figure 5, where we render the predicted tri-plane of our base diffusion, $\hat{x_0}$, at each time step t. Notwithstanding that our diffusion is performed in tri-plane feature space, the reverse process is similar to that in image space, where the coarse structure appears first and fine details appear in the last iterative steps.

Scale	w/o CFG	1.2	1.5	3.0	6.0
PSNR	24.06	**24.21**	24.07	24.05	24.15
SSIM	**0.795**	0.794	0.792	0.782	0.775
LPIPS	0.128	**0.121**	0.133	0.141	0.146

Table 1. Quantitative results of conditional avatar reconstruction.

C.3. Effect of 3D-aware Convolution

By rolling out tri-plane feature maps and applying 3D-aware convolution, the RODIN model performs 3D-aware diffusion using an efficient 2D architecture. As analyzed in Section **??**, tri-plane roll-out and 3D-aware convolution are crucial for high-fidelity results, especially for thin structures such as hair strands and clothing details, by enhancing cross-plane communication. To validate the impact of these designs in high-quality tri-plane, we modify the upsample diffusion model with different configurations and remove the convolution refinement with the base diffusion fixed. Figure 6 demonstrates with rollout and 3D-aware convolution, the full model shows a clear improvement compared to the base model.

C.4. Nearest Neighbors Analysis

The RODIN model enables a hassle-free creation experience of an unlimited number of avatars from scratch, each avatar being distinct. Figure 7 shows the nearest neighbors of some generated samples in the main paper, which indicates that the model does not simply memorize the training data.

C.5. Conditional Avatar Generation

Quantitative metrics. On top of unconditional generation, we can also hallucinate a 3D avatar from a single reference image by conditioning the base generator with the CLIP image embedding for that input image. We evaluate the conditional generation on 1K test data where each subject contains 300 images from different views. Table 1 reports the metrics between reconstructed images and ground-truth synthetic images.

Classifier-free guidance. Our model supports classifier-free guidance (CFG) sampling when inference, which is a technique typically used to boost the sampling quality in conditional generation. Table 1 evaluates generation quality with different scales of classifier-free guidance in terms

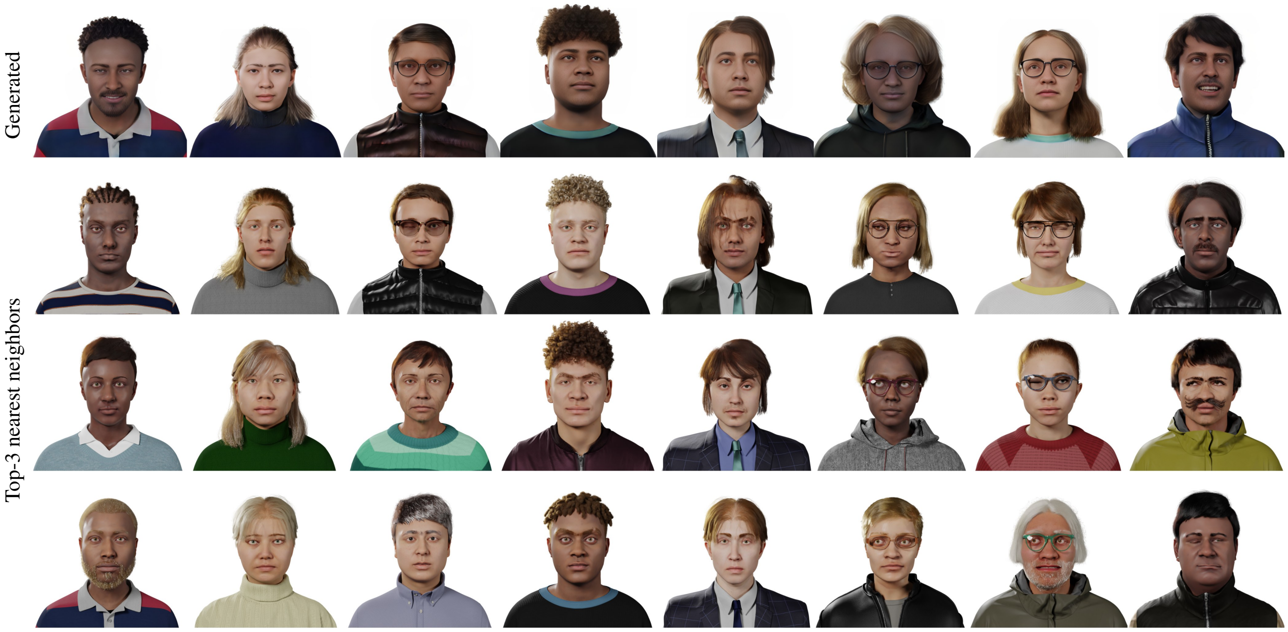

Figure 7. Nearest neighbors in the training data according to CLIP feature similarity.

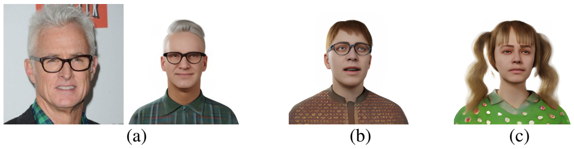

Figure 8. Failure cases.

User study	Ours > GIRAFFE	Ours > EG3D	Ours > Autoencoder
Preference Rate	100%	90.8%	95.4%

Table 2. User study.

of PSNR, SSIM and LPIPS.

C.6. User Study

We perform a user study to evaluate the randomly sampled avatars, where each participant is given a pair of results from different methods at once and asked to select a better one. As shown in Table 2, Rodin outperforms other methods by a large margin.

D. Failure Cases

As shown in Fig. 8, our method still has some limitations: (a) The current model may not generate old people or children well due to data bias. (b) Complex patterns in clothes are challenging to generate. (c) Sometimes there are floating NeRF artifacts.

E. Additional Visual Results

Figure 9 and Figure 10 show more random samples generated by the RODIN model, showing the capability to synthesize high-quality 3D renderings with impressive details. To reflect the geometry, we also extract the mesh from the generated density field using marching cubes, which demonstrates high-fidelity geometry. Figure 11 gives uncurated generated samples, which possess visually-pleasing quality and diversity. We also explore the interpolation of the latent condition z between two generated avatars, as shown in Figure 12, where we observe consistent interpolation results with smooth appearance transition. Figure 12 shows additional results of creating 3D portraits from a single reference image.

F. Societal Impact

The RODIN model aims to enable a low-cost, fast and customizable creation experience of 3D digital avatars that refer to the traditional avatars manually created by 3D artists, as opposed to photorealistic avatars. The reason for focusing on digital avatars is twofold. On the one hand, digital avatars are widely used in movies, games, the metaverse, and the 3D industry in general. On the other hand, the available digital avatar data is very scarce as each avatar has to be painstakingly created by a specialized 3D artist using a sophisticated creation pipeline, especially for modeling hair and facial hair.

Rather than collecting real photos, all our training images are rendered by Blender. Such synthetic data can miti-

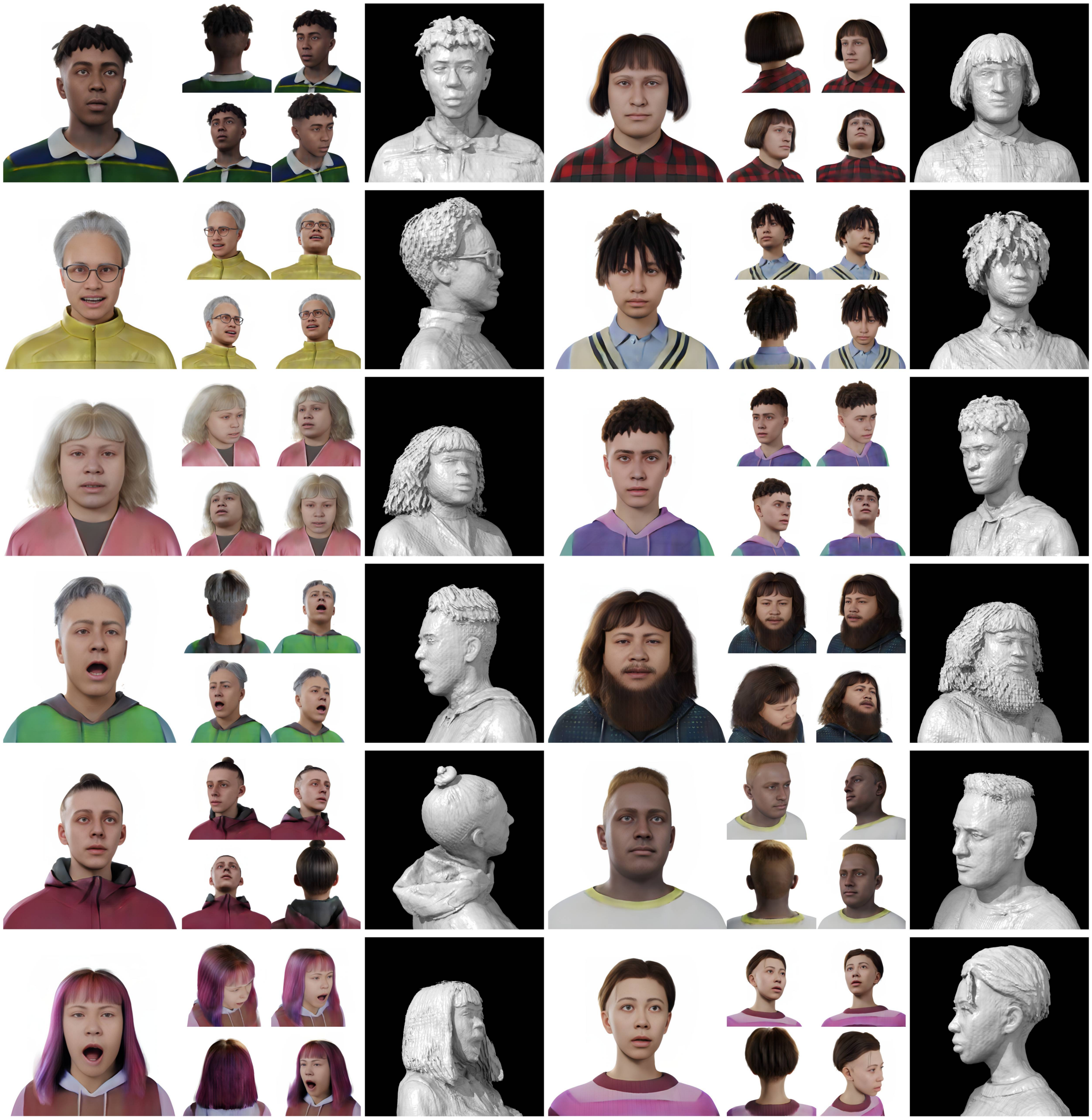

Figure 9. Unconditional generation samples by our RODIN model. We visualize the mesh extracted from the generated density field.

gate the privacy and copyright concerns that existed in real face collection. Another advantage of using synthetic data is that we could have control over the variation and diversity of rendered images, eliminating the data bias in existing face datasets. Also, digital avatars are easier to be distinguished from real people compared with photo-realistic avatars, hindering misuse for impersonating real persons. Nonetheless, the 3D portrait reconstruction and text-based avatar customization could still be misused for spreading disinformation maliciously, like all other AI-based content generation models. We caution that the high-quality renderings produced by our model may potentially be misused and viable solutions so avoid this include adding tags or watermarks when distributing the generated photos.

This work successfully generalizes the power of diffusion models from 2D to 3D and is promising to offer the new design tool for 3D artists which could significantly save the costs of the traditional 3D modeling and rendering pipeline. In the next we intend to explore the possibility of modeling general 3D scenes using the same technique and investigate novel applications such as Lego and architect designs.

References

[1] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022. 2

[2] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J 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*, pages 16123–16133, 2022. 2

[3] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1

[4] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations, ICLR*, 2019. 1

[5] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[6] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2

[7] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021. 1

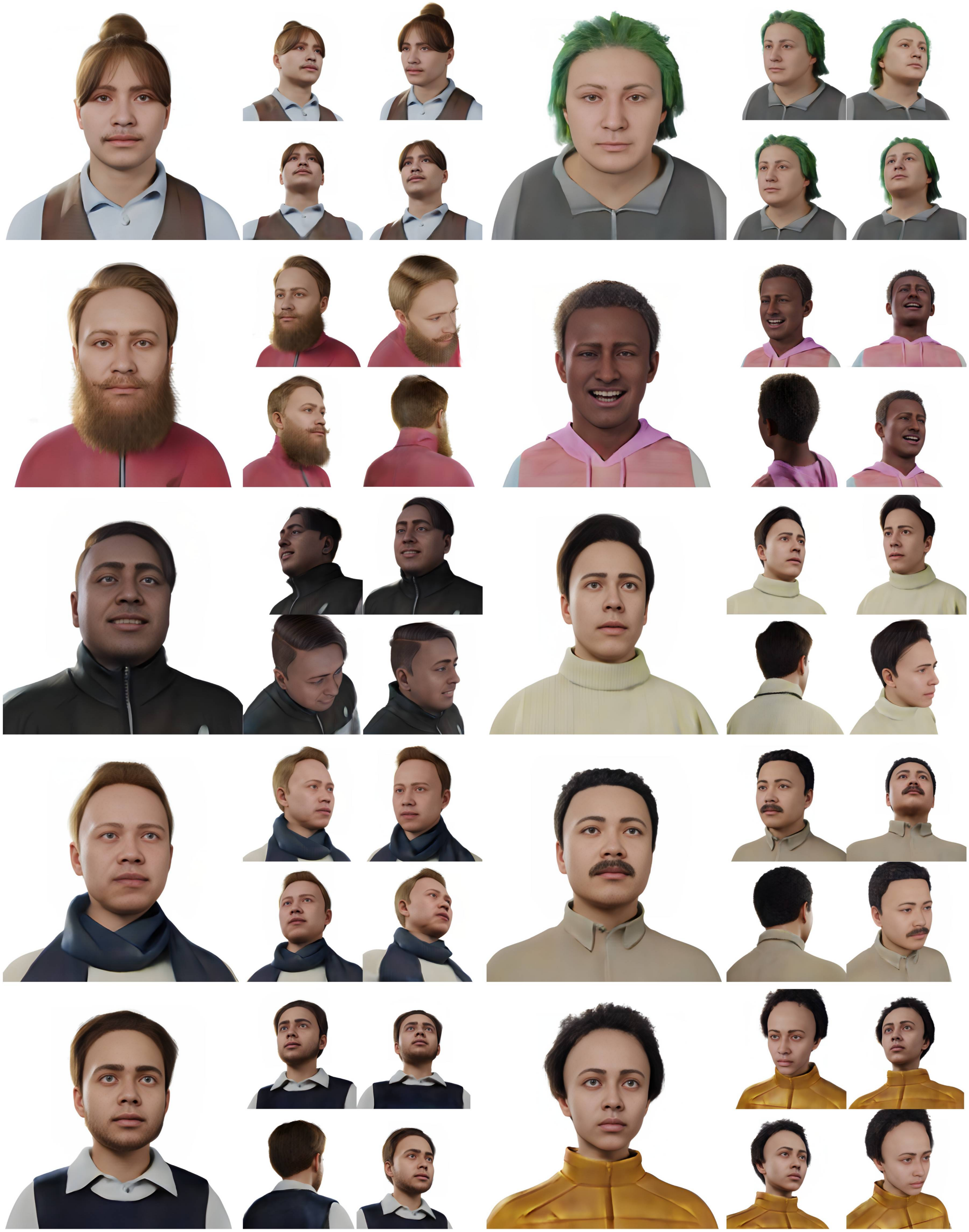

Figure 10. Unconditional generation samples by our RODIN model.

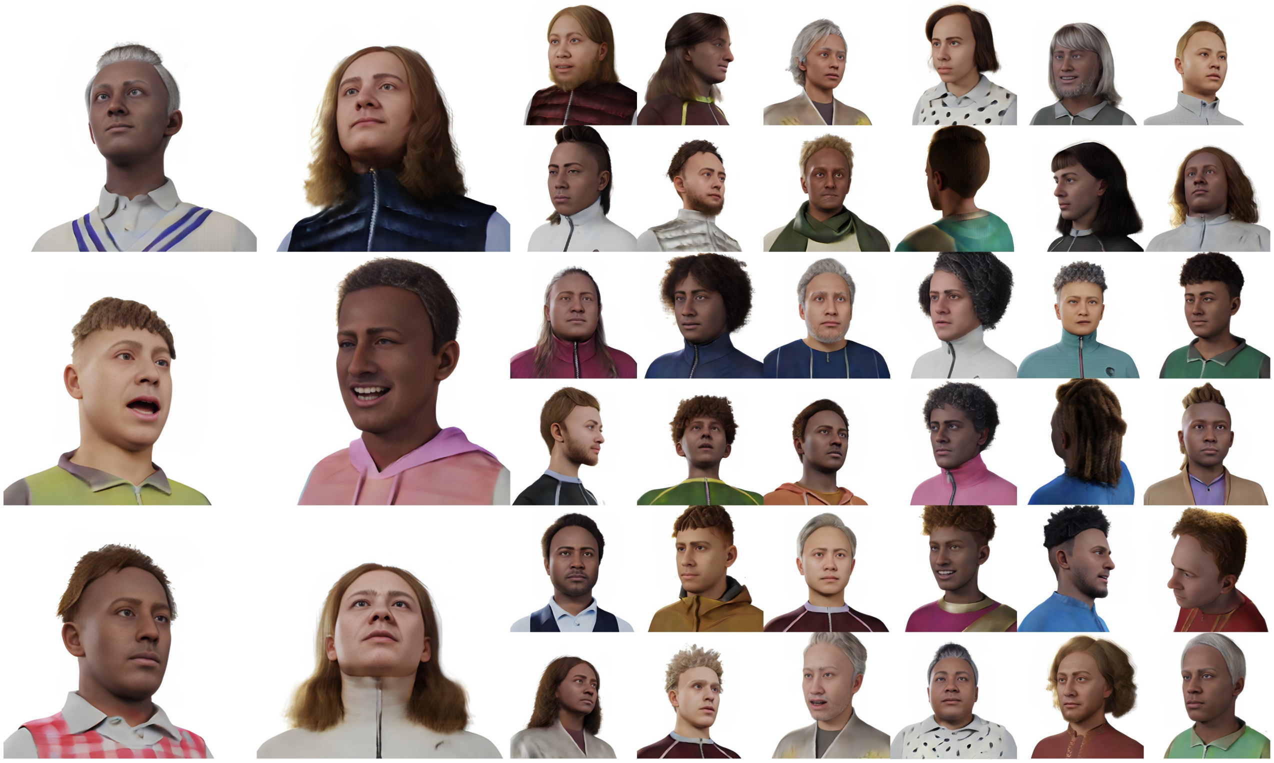

Figure 11. Uncurated generation results by our RODIN model.

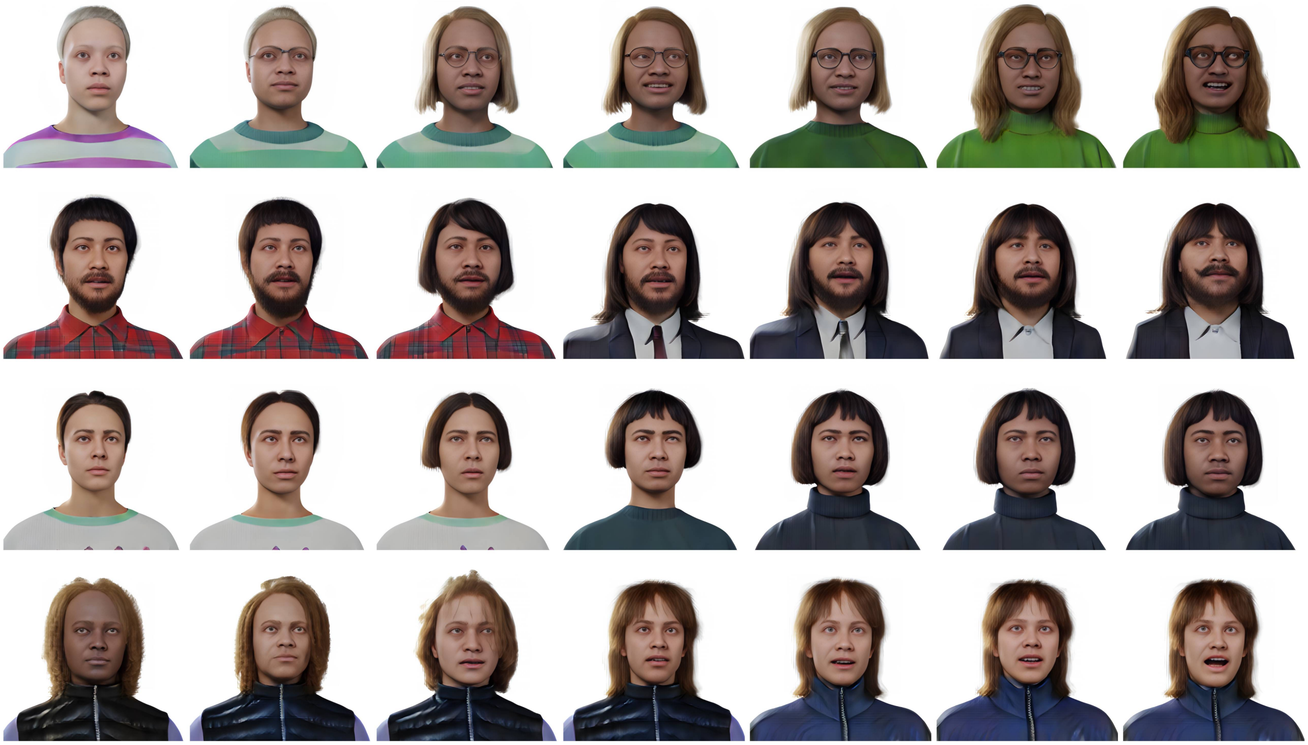

Figure 12. Latent interpolation results for generated avatars.

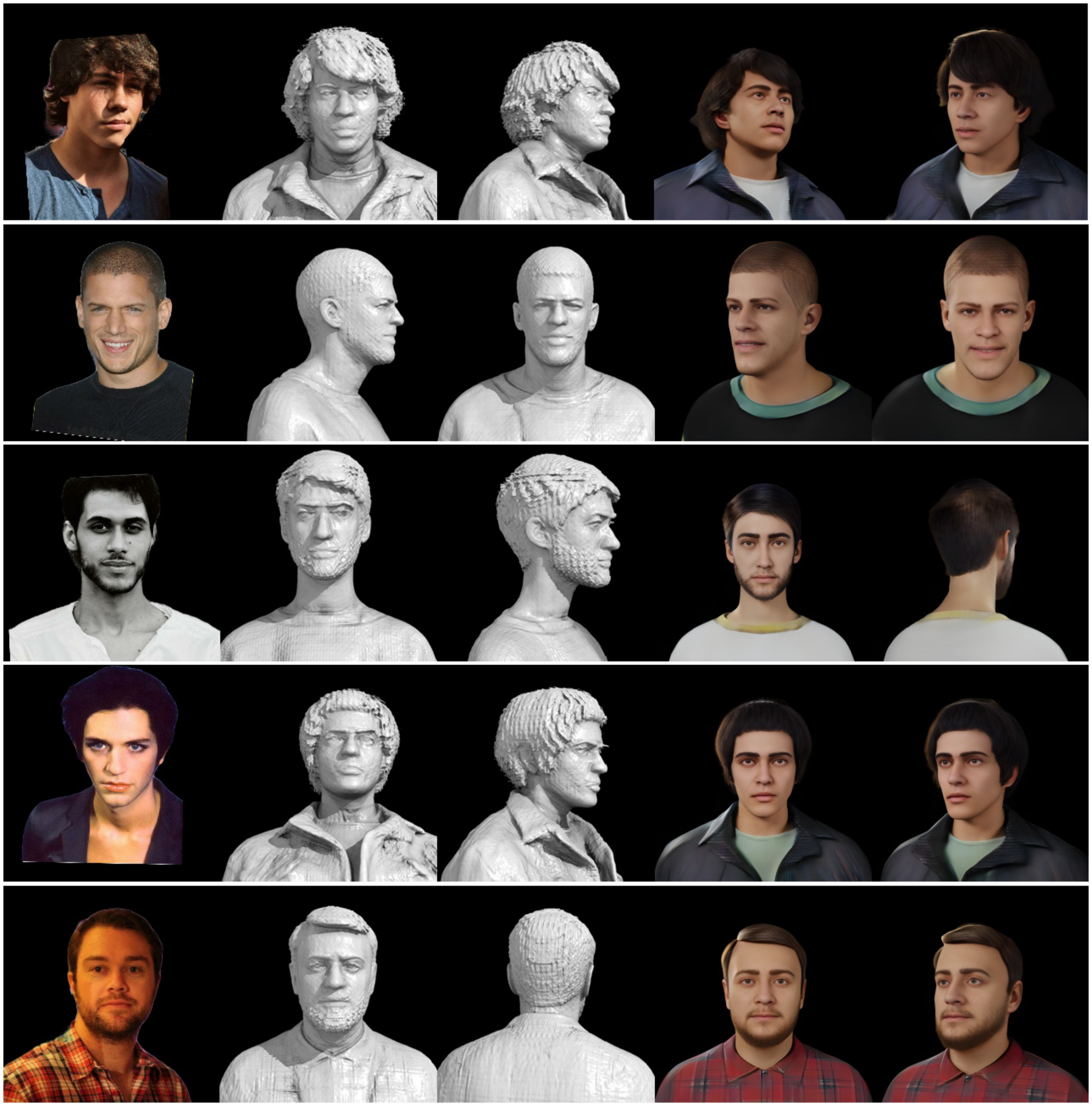

Figure 13. Additional results of portrait inversion.