AvatarArtist: Open-Domain 4D Avatarization

Supplementary Material

Appendix

In the supplementary materials, we first discuss the limitations of our proposed method (Sec. A). Following that, we explore another 4D representation, providing a detailed analysis of the parametric triplane (Sec. B).We then explored different model architectures to validate the superiority of our DIT + render approach. We provide additional implementation details, including the domains used during training, the specific training procedures for each model, and other relevant training configurations (Sec. D). We provide additional comparisons and visual results to further demonstrate the effectiveness of our method (Sec. E). Last but not least, we present more results in the supplementary video.

A. Limitations

While our method can handle inputs from various domains and generate high-fidelity avatars, it does not adequately separate the head region from the background, nor does it decouple neck rotation from the camera pose, which limits the realism of the final results. The 4D representation we employ uses a mesh as the primary driving signal. Although we incorporate motion embeddings as a supplementary motion signal, the process of obtaining the mesh is both time-consuming and imprecise, which adversely affects the overall efficiency and accuracy of the avatar generation.

B. Exploration of the 4D Representation

In Portrait4D [4], a 4D GAN (GenHead) based on a deformation field representation [9] achieved impressive generative results. Specifically, the GenHead G consists of a part-wise triplane generator G_{ca} for synthesizing the canonical triplane and a part-wise deformation field D for morphing the canonical head. It generates the 3D deformation field based on FLAME [8] expression coefficients and synthesizes the canonical triplane using the shape parameter from FLAME. During inference, the canonical triplane can be driven by applying the deformation field to compute the offset for each point in the triplane with the corresponding Flame parameters.

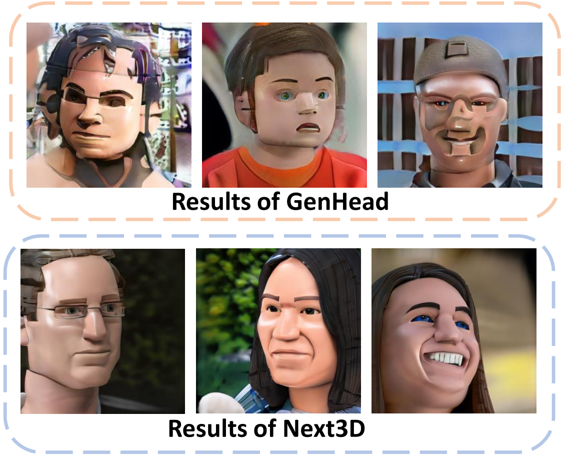

Figure S1. Visualization of generation results of different 4D GANs, including Next3D [11] and GenHead [4], on the unrealistic domain. We use the domain of Lego here. GenHead tends to produce artifacts, whereas Next3D achieves much better results, generating more plausible content.

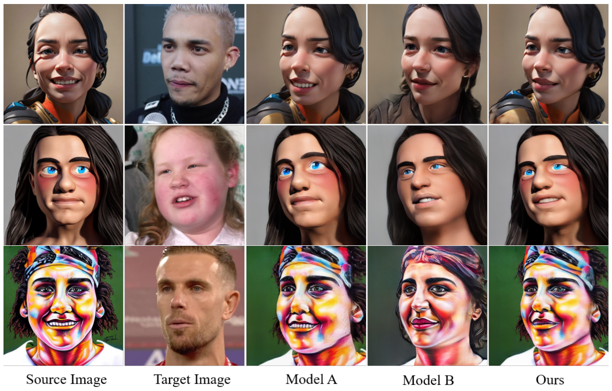

Figure S2. Visualization of different model results. Model A and Model B are two different end-to-end method which not use the Dit. For more details, please refer the Sec. C.

This canonical tri-plane and deformation field can also form a type of 4D representation. However, it is not suitable for our task. First, the deformation field changes according to different facial expressions, making it an unstable representation. In contrast, our representation only varies based on the subject's identity, ensuring consistency across different expressions for the same individual. Additionally, we found that GenHead does not perform well in open-domain generation. We suspect that this representation requires highly precise canonical space modeling, which is particularly challenging for non-realistic domains. In contrast, NeTX3D's representation focuses more on motion modeling while delegating identity preservation to a separate CNN. Compared to GenHead, this representation is more implicit and better suited for generating characters across different domains. (See Figure S1).

C. Effectiveness of model design

To demonstrate the clear effectiveness of using a DiT model for triplane generation, we conduct experiments comparing it with two feedforward approaches, as illustrated in Fig. S2. Model A, similar to Portrait4D, uses only 4D RGB data, preserving identity well but struggling with motion transfer due to the absence of a unified 4D representation and limitations of cross-attention for cross-domain motion retargeting. Model B, which operates without cross-attention, uses an encoder-decoder to convert input images into parametric triplanes and a ViT decoder to refine animated features. While effective at transferring expressions, the encoder-decoder based feedforward model fails to reconstruct accurate triplanes, leading to identity loss and making it more challenging for the ViT decoder to bridge the identity gap. In contrast, similar to VASA-1 [13], our diffusion + renderer pipeline leverages the target parametric triplanes fitting ability of a powerful generative model. This enables our method to simultaneously maintain both motion and identity, achieving the highest quality results.

D. More implementation details

D.1. Training Domains

As mentioned in our main paper, we used 28 domain images during training, including the original realistic domain. We categorize our domains into two types. The first type uses the official Stable Diffusion 2.1 model [10] as the generative model. For this type, the text prompts used are shown in Table S1, and we generate images in 20 different domain styles, with 6,000 images per domain. The second type, as shown in Table S2, utilizes third-party models in Civitai [1] as the generative models, where each model corresponds to a specific style. For these models, the same text prompt is used across all models, and we set the prompt as ”masterpieces, portrait, high-quality”.

D.2. 4D GAN

The 4D GANs (Next3D) for different domains were fine-tuned from the original FFHQ GAN. Similar to DATID-3D [7], the training was stopped once the GAN had seen 200,000 images. We set the batch size to 32 and used 8 A100 GPUs to fine-tune the model for 2 hours. A learning rate of 0.002 was used for both the generator and discriminator. For the discriminator's input, we applied image blurring, progressively reducing the blur degree as described in [2, 6], and we did not employ style mixing during training. We used the ADA loss combined with R1 regularization, with the regularization coefficient set to $\lambda = 5$. Additionally, the strength of the density regularization was set to $\lambda_{\text{den}} = 0.25$.

D.3. VAE

We follow the LVDM [5] and use a lightweight 3D autoencoder as our VAE. This VAE consists of an encoder E and a decoder D. Both the encoder and decoder comprise multiple layers of 3D convolutions. During training, we render the parametric triplane to obtain both depth maps and rendered images, and compute the L_1 and LPIPS losses separately. We also add a KL divergence loss to ensure that the latent feature distribution is similar to the Gaussian prior $p(h) = \mathcal{N}(0, 1)$. The weight of L_1 loss in triplane and depth is 1, the weight of LPIPS loss in the image is 1, and the weight of KL loss is 1×10^{-5}. We randomly sample camera poses during rendering, with the sampling ranges set to pitch in $[-0.25, 0.65]$ radians, yaw in $[-0.78, 0.78]$ radians, and roll in $[-0.25, 0.25]$ radians. The visual results of our VAE are shown in Figure S3.

D.4. DiT

The VAE compresses the triplane into $z_t \in \mathbb{R}^{64\times64\times4\times8}$. The DiT reshapes z_t to $64 \times 256 \times 8$, adds positional embeddings, and then flattens it before feeding it into the Transformer for training. Following the approach in Direct3D [12], at each DiT block, we concatenate DINO tokens with the flattened z_t and pass them through a self-attention mechanism to capture the intrinsic relationships between the DINO tokens and z_t. Afterward, we discard the image tokens, retaining only the noisy tokens for input to the next module. Moreover, to reduce the number of parameters and computational cost, we adopt adaLN-single, as introduced in PixArt [3]. This method predicts a set of global shift and scale parameters $P = [\gamma_1, \beta_1, \alpha_1, \gamma_2, \beta_2, \alpha_2]$ using time embeddings. A trainable embedding is then added to P in each block for further adjustment. During training, the batch size is set to 1536, and the training is conducted over 48 Tesla A100 GPUs (batch size 32 for each GPU), each with 80GB of memory, for a total of 5 days.

D.5. Motion-Aware Cross-Domain Renderer

During the Next3D rendering process in Figure. S4, a CNN is used to refine the dynamic components after rasterization, eliminating artifacts introduced in the rasterization stage (e.g., teeth completion, identity leakage). When training Next3D for different domains, we fine-tune this CNN, as well as the MLPs used in both super-resolution and neural rendering. Therefore, a unified renderer is required to handle parametric triplanes from various domains and mitigate issues caused by rasterization.

As mentioned in our main paper, we find a simple CNN can not handle the cross-domain parametric triplanes, and we propose the motion-aware cross-domain renderer. To train the motion-aware cross-domain renderer, we use the trained 4DGAN to generate the 4D images (i.e., multi-

Table S1. List of full-text prompts corresponding to each domain. The images for these domains were generated using SD-V1.5 as the base model, in combination with corresponding prompts.

Concise Name of Domain	Full text prompt
Pixar	a 3D render of a face in Pixar style
Lego	a 3D render of a head of a lego man 3D model
Greek statue	a FHD photo of a white Greek statue
Elf	a FHD photo of a face of a beautiful elf with silver hair in live action movie
Zombie	a FHD photo of a face of a zombie
Tekken	a 3D render of a Tekken game character
Devil	a FHD photo of a face of a devil in fantasy movie
Steampunk	Steampunk style portrait, mechanical, brass and copper tones
Mario	a 3D render of a face of Super Mario
Orc	a FHD photo of a face of an orc in fantasy movie
Masque	a FHD photo of a face of a person in masquerad
Skeleton	a FHD photo of a face of a skeleton in fantasy movie
Peking Opera	a FHD photo of face of character in Peking opera with heavy make-up
Yoda	a FHD photo of a face of Yoda in Star Wars
Hobbit	a FHD photo of a face of Hobbit in Lord of the Rings
Stained glass	Stained glass style, portrait, beautiful, translucent
Graffiti	Graffiti style portrait, street art, vibrant, urban, detailed, tag
Pixel-art	pixel art style portrait, low res, blocky, pixel art style
Retro	Retro game art style portrait, vibrant colors
Ink	a portrait in ink style, black and white image

Table S2. List of models used for each domain. The images for these domains were generated using specific models as base models. All models were sourced from Civitai [1], an AI-Generated Content (AIGC) social platform.

Concise Name of Domain	Model Name
3D-Animation	3D Animation Diffusion-V1.0
Toon	ToonYou-Beta6
AAM	AAM Anime Mix
Counterfeit	Counterfeit-V3.0
Pencil	Pencil Sketch
Lyriel	Lyriel-V1.6
XXM	XXMix9realistic

view, multi-expression images of the same individual), and we are able to simultaneously obtain the corresponding depth, parametric triplane, and rendering features. The data is separated into static and dynamic parts similar to Portrait4D [4], as mentioned in our main paper. The overall training objective of our renderer is defined as follows:

$$\mathcal{L} = \mathcal{L}_{\text{re}} + \mathcal{L}_{\text{f}} + \mathcal{L}_{\text{tri}} + \mathcal{L}_{\text{depth}} + \mathcal{L}_{\text{opa}} + \mathcal{L}_{\text{adv}}, \tag{S1}$$

where $\mathcal{L}_{\text{re}}$ represents a combination of the LPIPS and L_1 distances between the generated image I_o and its corresponding ground truth. $\mathcal{L}_{\text{tri}}$ measures the L_1 difference between the generated triplane features and their ground truth. $\mathcal{L}_{\text{f}}$ computes the L_1 difference between the generated rendering features and their respective ground truth. $\mathcal{L}_{\text{depth}}$ evaluates the L_1 difference between the generated depth map and its ground truth counterpart. $\mathcal{L}_{\text{opa}}$ is the L_1 difference between the predicted opacity and the ground truth. Finally, $\mathcal{L}_{\text{adv}}$ represents the adversarial loss between I_o and the ground truth image, utilizing the discriminator from the Next3D model.

The loss balancing weights for each term in Eq. (S1) are set to 1, 1, 0.1, 1, 1, and 0.01 for $\mathcal{L}_{\text{re}}$, $\mathcal{L}_{\text{f}}$, $\mathcal{L}_{\text{tri}}$, $\mathcal{L}_{\text{depth}}$, $\mathcal{L}_{\text{opa}}$, and $\mathcal{L}_{\text{adv}}$, respectively. For the first 1000K images, $\mathcal{L}_{\text{adv}}$ is not applied, and the parameters in both the neural renderer and super-resolution components are kept fixed. After 1000K images, $\mathcal{L}_{\text{adv}}$ is introduced, and the trainable parameters of the neural renderer and super-resolution modules are unfrozen. We employ volume rendering with 48 coarse samples and 48 fine samples per ray. The initial volume rendering resolution is set to 64^2 for the first 1000K

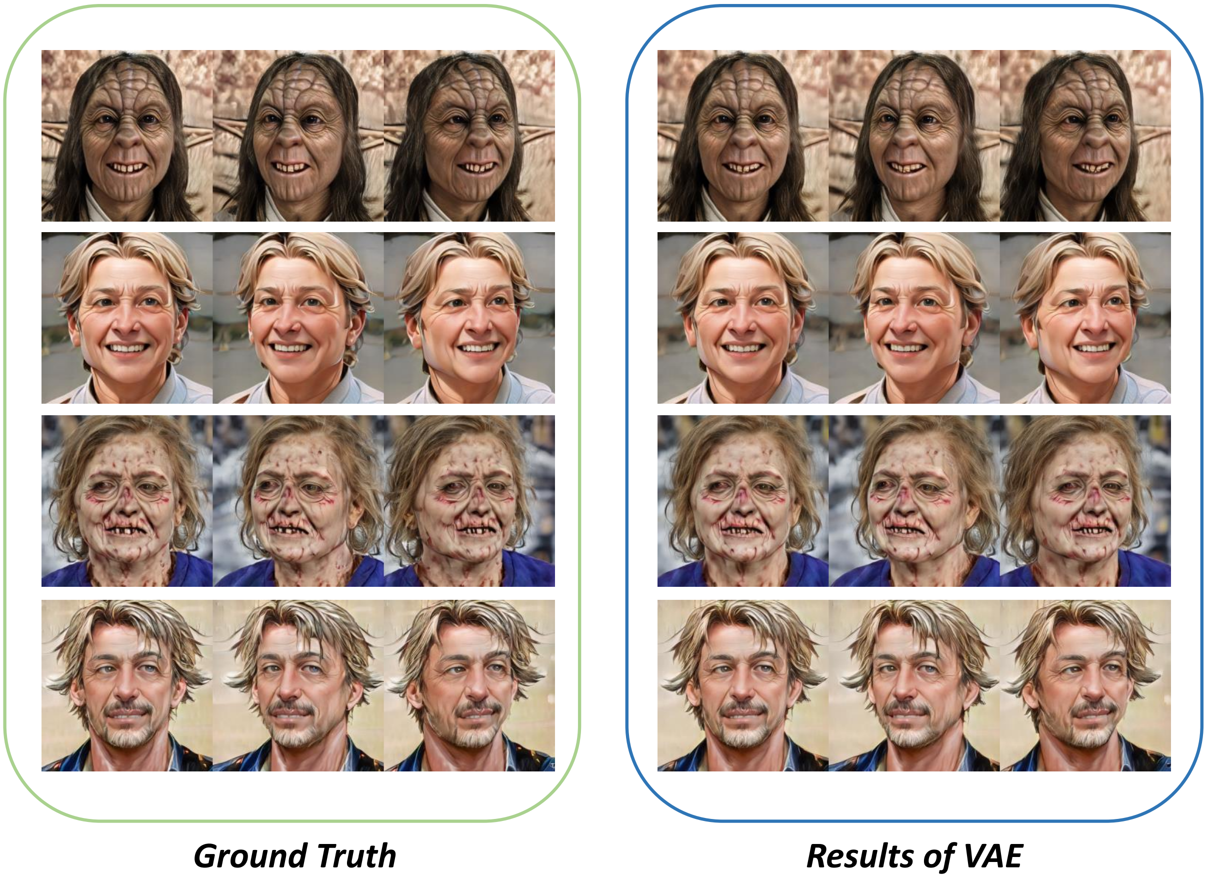

Figure S3. Visualization of reconstruction results of our VAE. The domain is Yoda, 3D-Animation, Zommbie, and Counterfeit, respectively. The ground truth images are generated with the Next3D.

images, gradually increasing to 128^2 as training progresses. The model is trained on a total of 8 million images. We utilize the Adam optimizer with $(\beta_1, \beta_2) = (0.9, 0.999)$ and a learning rate of 1×10^{-4} across all networks. The batch size is set to 96, with an even split between dynamic and static data. The training is conducted over 24 Tesla A100 GPUs, each with 80GB of memory, for a total of 4 days.

E. Additional Comparisons and Visual Results

E.1. User Study

For a more comprehensive evaluation, we conducted a user study with 10 participants, who were asked to assess image sharpness, temporal consistency, expression consistency, and identity consistency. They did so by selecting the best method while reviewing 12 cross-ID reenactment results generated by different approaches.

For each evaluation criterion, participants were presented with five videos, each corresponding to the results produced by a different method. They were instructed to rate the videos on a scale from 1 to 5, where 5 indicates the highest quality and 1 the lowest. Multiple methods could receive the same score. As shown in Table S3, our method exhibits significant advantages over the others.

Model	**Trained Domains / Untrained Domains**			
	Sharpness	**Temporal**	**Expression**	**Identity**
LivePortrait	3.625 / 2.5	3.625 / 2.5	3.5 / 1.5	3.5 / 1.5
Xportrait	2.375 / 1	1.625 / 1	2 / 1	1.875 / 1.5
Invertavatar	2.625 / 2.5	2.25 / 2.5	2.375 / 2	2.75 / 2.5
Portrait4D	1.875 / 3.5	2.5 / 3	2.125 / 2.5	2.375 / 2
Ours	**4.625 / 5**	**4.25 / 5**	**4.375 / 4.5**	**4.125 / 5**

Table S3. User Study.

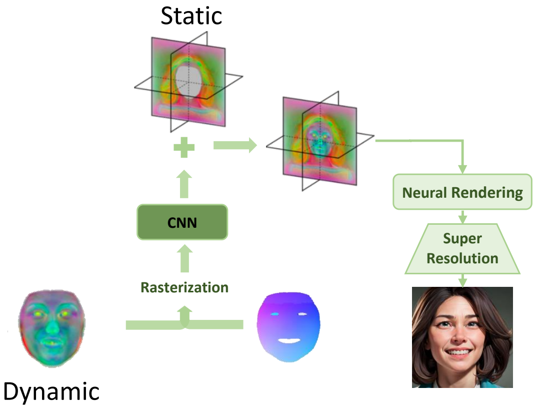

Figure S4. Visualization of rendering process of Next3D. After rasterization, a CNN is employed to remove artifacts introduced during the rasterization process, which is critical for final performance, as mentioned in the Next3D [11].

E.2. Visual Comparisons

In Figure S6, we present additional visual comparisons, demonstrating that our method achieves superior performance. Moreover, we present our geometric results in Figure S5. For more visual results, please refer to our video results.

References

[1] Civitai. https://civitai.com/. 2, 3

[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] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. 2

[4] Yu Deng, Duomin Wang, Xiaohang Ren, Xingyu Chen, and Baoyuan Wang. Portrait4d: Learning one-shot 4d head avatar synthesis using synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7119–7130, 2024. 1, 3

[5] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*, 2(3):4, 2022. 2

[6] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *Proc. NeurIPS*, 2021. 2

[7] Gwanghyun Kim and Se Young Chun. Datid-3d: Diversity-preserved domain adaptation using text-to-image diffusion for 3d generative model, 2022. 2

[8] 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):194:1–194:17, 2017. 1

[9] Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5865–5874, 2021. 1

[10] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2

[11] Jingxiang Sun, Xuan Wang, Lizhen Wang, Xiaoyu Li, Yong Zhang, Hongwen Zhang, and Yebin Liu. Next3d: Generative neural texture rasterization for 3d-aware head avatars. In *CVPR*, 2023. 1, 5

[12] Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Jingxi Xu, Philip Torr, Xun Cao, and Yao Yao. Direct3d: Scalable image-to-3d generation via 3d latent diffusion transformer. *arXiv preprint arXiv:2405.14832*, 2024. 2

[13] Sicheng Xu, Guojun Chen, Yu-Xiao Guo, Jiaolong Yang, Chong Li, Zhenyu Zang, Yizhong Zhang, Xin Tong, and Baining Guo. VASA-1: Lifelike audio-driven talking faces generated in real time. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2

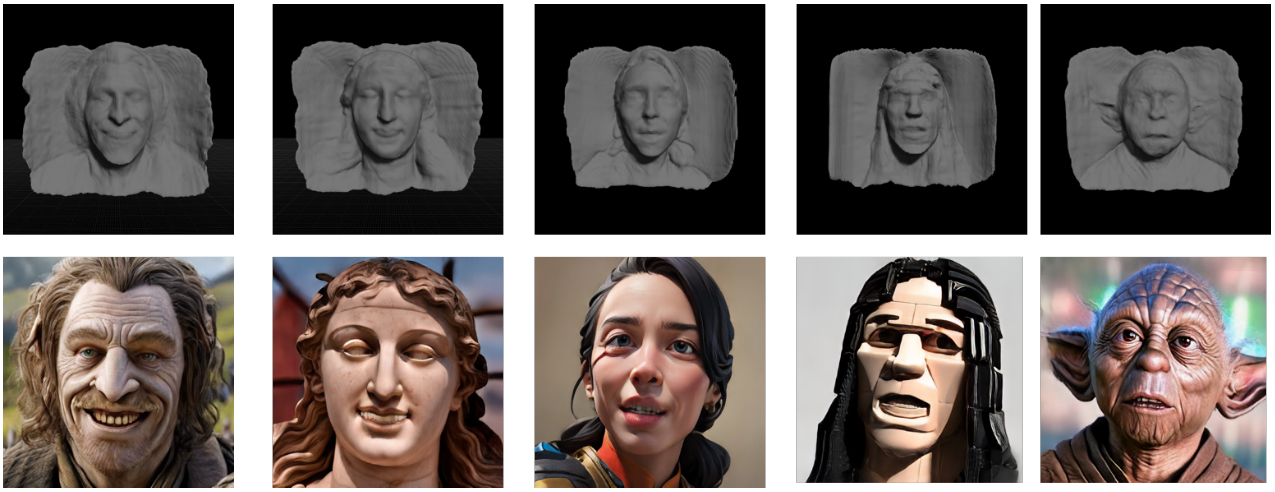

Figure S5. The geometry results of our method.

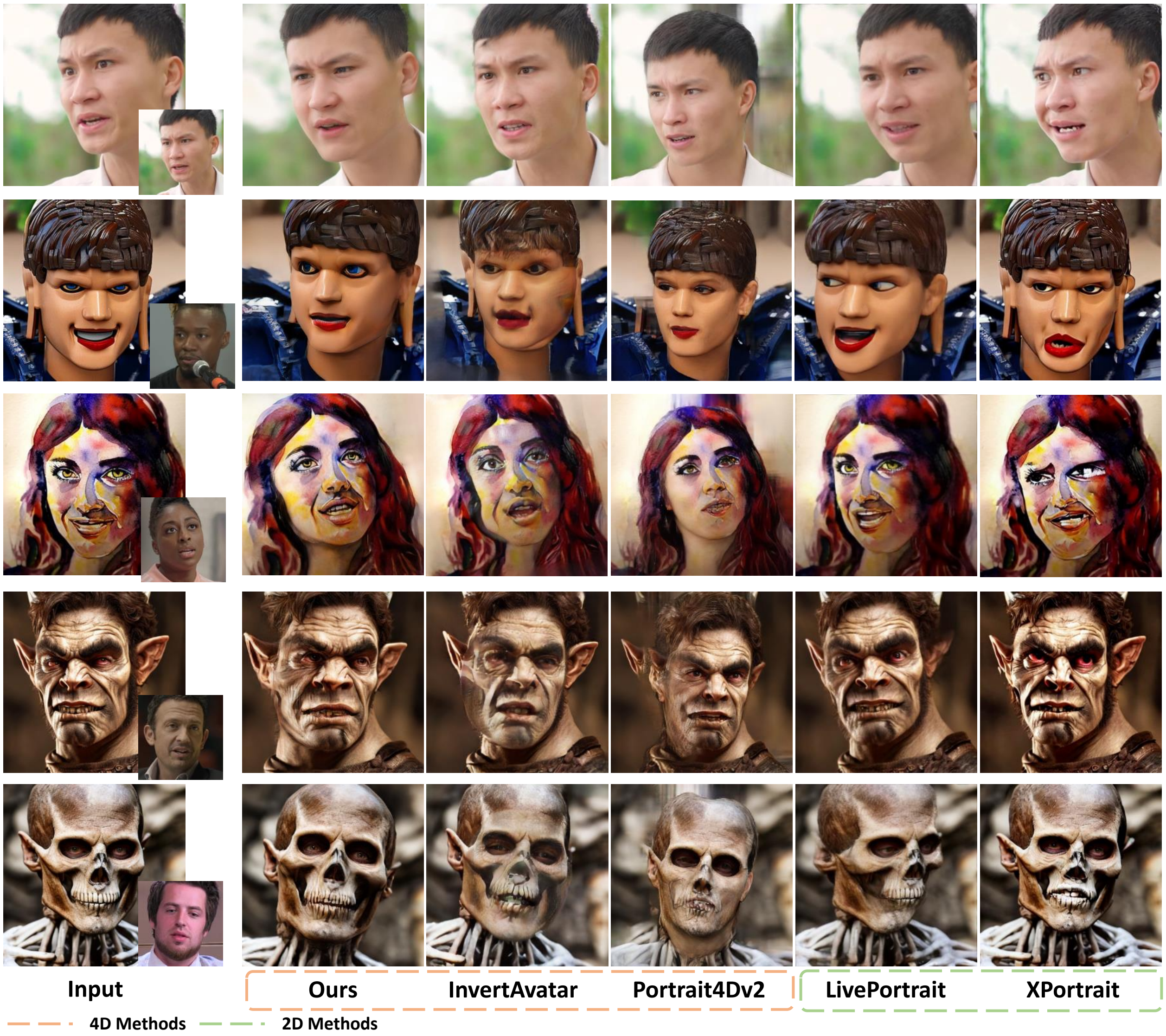

Figure S6. Qualitative comparison with state-of-the-art methods. The leftmost column of the figure presents the input images, with the bottom-right corner indicating the target image. The first row illustrates the results of self-reenactment, while the subsequent rows showcase the results of cross-reenactment. Our method demonstrates superior performance in terms of expression and pose consistency, as well as identity preservation. For more visual results, please refer to our video results.