NeRFFaceShop: Learning a Photo-realistic 3D-aware Generative Model of Animatable and Relightable Heads from Large-scale In-the-wild Videos

Kaiwen Jiang, Feng-Lin Liu, Shu-Yu Chen, Pengfei Wan, Yuan Zhang, Yu-Kun Lai, Hongbo Fu, and Lin Gao*

***Abstract*—Animatable and relightable 3D facial generation has fundamental applications in computer vision and graphics. Although animation and relighting are highly correlated, previous methods usually address them separately. Effectively combining animation methods and relighting methods is nontrivial. In terms of explicit shading models, animatable methods cannot be easily extended to achieve realistic relighting results, such as shadow effects, due to prohibitive computational training costs. Regarding implicit lighting representations, current animatable methods cannot be incorporated due to their inharmonious animation representations, i.e., deforming spatial points. This paper, armed with a lightweight but effective lighting representation, presents a compatible animation representation to achieve a disentangled generative model of 3D animatable and relightable heads. Our represented animation allows for updating and control of realistic lighting effects. Due to the disentangled nature of our representations, we learn the animation and relighting from large-scale, in-the-wild videos instead of relying on a morphable model. We show that our method can synthesize geometrically consistent and detailed motion along with the disentangled control of lighting conditions. We further show that our method is still compatible with morphable models for driving generated avatars. Our method can also be extended to domains without video data by domain transfer to achieve a broader range of animatable and relightable head synthesis. We will release the code for reproducibility and facilitating future research.**

***Index Terms*—Face animation, face relighting, volume disentangling, neural radiance fields, neural rendering**

* Corresponding Author is Lin Gao (gaolin@ict.ac.cn).

This work was supported by Kuaishou Technology, and sponsored by Beijing Municipal Science and Technology Commission (No. Z231100005923031), National Natural Science Foundation of China (No.62322210 and No.62472407), and Innovation Funding of ICT, CAS (No. E461020). The authors would like to acknowledge the Nanjing Institute of InforSuperBahn OneAiNexus for providing the training and evaluation platform.

Kaiwen Jiang is with the CSE Department at the University of California, San Diego (e-mail: k1jiang@ucsd.edu).

Feng-Lin Liu, Shu-Yu Chen, and Lin Gao are with the Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences. Feng-Lin Liu and Lin Gao are also with the University of Chinese Academy of Sciences, China (e-mail: liufenglin21s@ict.ac.cn; chenshuyu@ict.ac.cn; gaolin@ict.ac.cn).

Pengfei Wan, Yuan Zhang are with the Kuaishou Technology, China (e-mail: wanpengfei@kuaishou.com; zhangyuan03@kuaishou.com).

Yu-Kun Lai is with the School of Computer Science and Informatics, Cardiff University, U.K. (e-mail: LaiY4@cardiff.ac.uk).

Hongbo Fu is with the Division of Arts and Machine Creativity, Hong Kong University of Science and Technology, Hong Kong (e-mail: fuplus@gmail.com).

I. Introduction

Generating animatable and relightable 3D heads is a long-standing problem in computer graphics and vision. Consistent and photo-realistic generation of such heads opens up rich possibilities for downstream applications, including AR/VR telepresence, virtual avatar design, etc.

Recently, unconditional photo-realistic generation of 3D heads (e.g., [6], [8], [16]) has achieved huge success by combining Neural Radiance Fields (NeRFs) [39] and Generative Adversarial Networks (GANs) [19]. Built upon the unconditional generation, several works (e.g., [4], [52], [61], [64]) deform their 3D representations with the guidance of morphable models [34] to achieve the animation control but lack the understanding of illumination. Some works (e.g., [11], [27], [46]) address the relighting control with either explicit shading models or implicit lighting representations but still cannot achieve animation control. We argue that animation and relighting are highly correlated because the lighting effects need to be correctly updated during the animation. To achieve a disentangled generation of photo-realistic 3D heads, we propose to solve animation and relighting simultaneously.

If we perceive the output of generation as pure images, a trivial solution to the above problem is to use an animatable method to generate a sequence of images and then apply a single-shot 2D relighting method to adjust the lighting conditions of the generated images. However, as discussed in [27], maintaining consistency across different camera parameters and expressions is challenging.

In contrast, this paper presents a disentangled generative model of 3D animatable and relightable heads built upon the tri-plane representation [6]. We observe the potential of implicit lighting representation [27] and augment it with a compatible animation representation to enable the animation and relighting simultaneously as in Fig. 1.

Instead of updating the coordinates of 3D points to represent the animation in previous works (e.g., [4], [52], [61]), we propose deforming the convolutional features in the generator [30] to ensure the consistent updating of lighting effects during the animation. Furthermore, unlike previous works (e.g., [4], [52], [61]) that rely on a statistical model constructed from accurate 4D scan data [34] to enable the animation, we get rid of this indirect reliance on the 4D scan data and learn the animation and relighting from collected large-scale in-the-

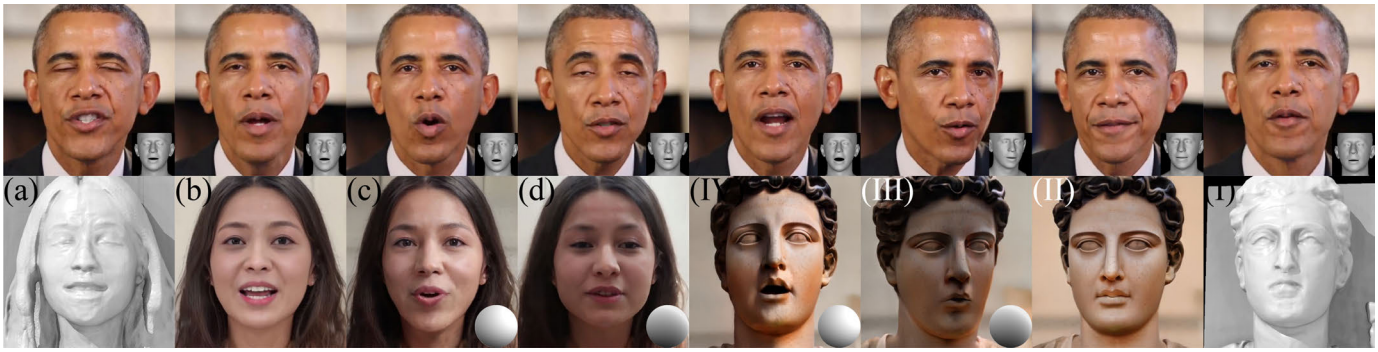


Fig. 1: We learn a photo-realistic 3D-aware generative model of animatable and relightable heads from large-scale in-the-wild videos. Given the driving frames in the first row, we use an off-the-shelf estimator [17] to detect expressions as sets of coefficients, which are visualized as meshes at the bottom right of the corresponding frames. The detected expressions are then mapped to the animation space of our model to synthesize correspondingly driven heads in the second row. (a-d) demonstrate the synthesized animatable and relightable human heads, while (I-IV) show the synthesized statue heads. (a) and (I) show the synthesized geometry. (b) and (II) display the generated pseudo-albedo. (c-d) and (III-IV) demonstrate the driven results rendered under different lighting conditions, which are visualized as spheres at the bottom right of the corresponding synthesized images. Note that our animation can be controlled through the adopted expression estimator and can re-interpret its coefficients to achieve detailed motion synthesis, which is more consistent with the driving frames, such as pouting in (c). Please refer to the accompanying video for the animation effects.

wild videos, which are more easily accessible and extensible. Our animation space is learned from scratch and is then able to capture detailed animation and synthesize smooth motions. By self-supervision, we could learn an encoder to map coefficient spaces from other morphable models (e.g., [34]) to our animation space for driving. However, qualified video data for a broader range of heads, such as cartoon characters and statues, are hard or impossible to access. Armed with the domain transfer techniques [1], [31], [69], we can finetune our trained model on human heads into novel domains.

In summary, the contributions of our work include:

- We propose compatible animation and lighting representations in a 3D GAN for consistently animating and relighting heads such that textures, including lighting effects, are correctly updated during the animation.
- We propose to train the representations by in-the-wild videos to realize detailed motion synthesis. We show that our animation representation is still compatible with existing morphable models for controlling motion synthesis.
- We conduct extensive experiments and comparisons to show our method achieves state-of-the-art 3D-aware animatable and relightable human head synthesis.

II. Related Work

A. 3D-aware Portrait Synthesis

The combination of NeRFs [39] and GANs [19] leads to great success in learning 3D-aware unconditional generative models of photo-realistic heads from in-the-wild image datasets (e.g., [29]). Recent years have witnessed rapid improvement over the quality, diversity, and speed of unconditional synthesis by developing the representations and training strategies (e.g., [6]–[8], [16], [20], [33], [40], [41], [48], [50], [65], [71]). Subsequent works (e.g., [4], [13], [14], [27], [28], [46], [51], [52], [61]) targeting disentanglement choose to build upon 3D-aware unconditional generative models to enjoy their synthesis quality. Our method chooses to build upon the tri-plane representation [6] and enables the simultaneous control of animation and lighting.

B. Facial Animation and Relighting in 3D GANs

Animating and relighting heads have been well-explored by editing latent codes (e.g., [2], [15], [55], [68]) in GANs. Extending the method of editing latent codes to 3D GANs (e.g., [54]), however, comes with the problem of inconsistent geometry during animation and relighting. Existing generative methods typically achieve geometrically consistent animation and relighting separately despite their correlation.

To achieve consistent animation, some works incorporate morphable models [23], [34], [44] into their synthesis networks. AniFaceGAN [61], GNARF [4], and OmniAvatar [64] update 3D points based on the morphing of corresponding meshes. Next3D [52] and InvertAvatar [72] embed a morphable mesh into the tri-plane representation [6] with separated static and dynamic modeling for consistency. However, their deformation is equivalent to altering the locations of 3D points with fixed features. In their representations, appearance is decoded mainly from features, leading to the fixed lighting effects during the animation, but relighting usually requires a global understanding of geometry, e.g., the visibility effects. Recently, a new group of auto-encoding methods (e.g., [10], [14], [35], [56], [57]) also observes the potential of video data. They build upon [59] and finetune the model with video data to enable the lifting from 2D images into 3D while allowing expressions to change. However, they cannot handle the relighting as well and they are not innate generative models.

For consistent relighting, some works (e.g., [11], [25], [42], [43], [46], [53]) choose to apply explicit shading models

Fig. 2: Comparison of synthesized motions for different methods given the driving frame in the leftmost column. The result of our method is the most consistent with the driving frame.

in their synthesis networks for relighting. However, there is always a trade-off between required computation and achieved realism. Another type of work [5], [27] uses an implicit representation with careful regularization for solving the relighting. It features lightweight computation and highly realistic relighting results under natural lighting conditions.

Combining the aforementioned animation methods with explicit shading models seems an option to enjoy both the consistent animation and relighting. However, to achieve accurate relighting results (e.g., visibility effects, subsurface scattering), the necessary adversarial training is still computationally prohibitive, especially for dynamic avatars, as discussed in [11]. Therefore, we propose to augment the implicit lighting representation [27] with a compatible animation representation to enable the animation while preserving its lightweight but realistic relighting capability. We learn this animation representation from data, i.e., large-scale in-the-wild videos, since the 3D points deformation guided by morphable models [34] cannot be easily incorporated. This choice also gives us the advantage of synthesizing more detailed motion, as shown in Fig. 2.

Additionally, recent works have explored ways to alter the convolutional features. Learning animation directly from in-the-wild videos might cause inconsistent geometry for static facial regions like neck, resulting in noisy facial shapes. PV3D [63] proposes to alter the features with synthesized residual features. Their approach is effective to represent animation but cannot ensure geometric consistency. In contrast, our method is able to preserve geometric consistency and effectively achieve both relighting and animation control.

C. Domain Transfer in GANs

Transferring learned generative models from one domain to another is especially helpful for modeling domains with limited data, such as cartoon characters, animals, etc. The recent advance (e.g., [1], [18], [31], [69]) can desirably preserve the latent space property while adapting the domains. To overcome the domain gap issue, DATID3D [31] proposes to leverage a diffusion model [47] to translate the human face or cat images into images of another domain but with the same viewpoint. They successfully generate the data in this way for adversarial training. We show that our method is compatible with the domain transfer as well. By learning from human heads with rich in-the-wild video data, our method is able to generalize to new domains whose video data can be hard or even impossible to access, such as cartoon characters, statues, etc.

III. Method

We first briefly discuss our backbone [27] in Sec. III-A. We then discuss our augmented compatible animation representation in Sec. III-B. After that, we illustrate how to train our pipeline with large-scale in-the-wild video data in Sec. III-C. Additionally, we show how to train an encoder to drive the generated identities using learned animation latent space in Sec. IV-A. Lastly, we introduce how to apply domain transfer techniques in our pipeline in Sec. IV-B.

A. Preliminaries

The approach of [27] uses a generator G [30] to transform Gaussian noises and spherical harmonic (SH) [45] coefficients into 3D head volumes, which can be used for rendering any views with the volume rendering [39] given camera parameters. It possesses two latent spaces: an albedo latent space and a lighting latent space. Gaussian noises $z \in \mathbb{R}^{512}$ are first transformed into samples in the albedo latent space as $\omega \in \mathbb{R}^{512}$, and SH coefficients $sh \in \mathbb{R}^9$ are transformed into samples in the lighting latent space as $l \in \mathbb{R}^{512}$. The lighting control is therefore achieved by fixing ω and adjusting SH coefficients only.

Specifically, G is composed of 9 consecutive synthesis blocks [30] $B = \{B^4, B^8, ..., B^{256}, B^{256}_{S^{(1)}}, B^{256}_{S^{(2)}}\}$, where the superscript denotes the resolution and we use this convention in the following content, as shown in Fig. 3 , each of which convolves a multi-channel feature map (referred as "convolutional features") from the previous block by conditional adaptive instance normalization [26]. The first 7 blocks are conditioned on ω, while the remaining 2 blocks are conditioned on both ω and l. By 1×1 convolution, the convolutional features in the first 7 blocks $F(\omega) = \{F^4(\omega), F^8(\omega), ..., F^{256}(\omega)\}$ are transformed into albedo tri-planes $\in \mathbb{R}^{96\times256\times256}$, and the convolutional features in the last 2 blocks $F_S(\omega, l) = \{F^{256}_{S^{(1)}}(\omega, l), F^{256}_{S^{(2)}}(\omega, l)\}$ are transformed into shading tri-planes $\in \mathbb{R}^{96\times256\times256}$ as the lighting representation. By sampling features from the albedo tri-planes and shading tri-planes, 3D head volumes can be determined with two additional lightweight decoders. With the volume rendering [39], photo-realistic facial images I can be rendered from 3D head volumes given camera parameters. An image discriminator D_{image} then discriminates whether I is real or fake with the conditions of camera parameters and SH coefficients for the adversarial training [19], [30]. The training dataset is labeled with ground-truth camera parameters and lighting conditions.

Notably, the lighting representation depends on the convolutional features in the first 7 blocks but not on the albedo tri-planes. Without special note, the implementation details agree with [27]. Please refer to [27] for full details.

B. Compatible Animation Representation

We observe that the albedo tri-planes in [27] contain both identity and animation information. We aim to distill the animation control from the generation of albedo tri-planes. To achieve this, we propose first to construct a separate representation for animation.

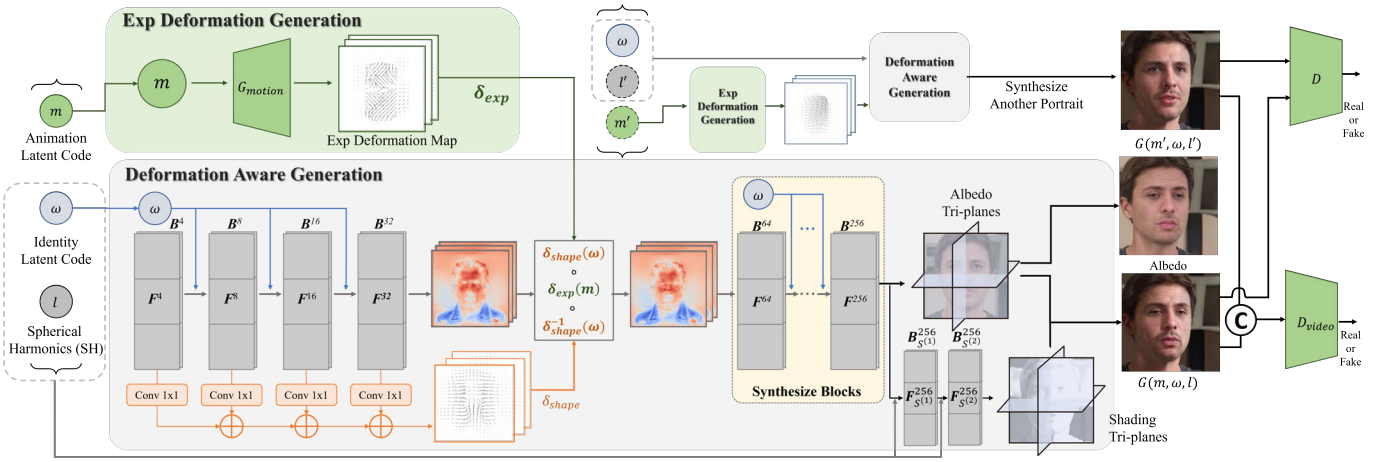


Fig. 3: An overview of our framework. Our pipeline is built on the tri-plane generator [6] G, but we roll out its convolutional features as $F^4, F^8, ..., F^{256}, F^{256}_{S(1)}, F^{256}_{S(2)}$. In particular, we have three latent spaces: identity latent space, animation latent space, and lighting latent space. For the animation representation, we append 1×1 convolutions [30] to the first four blocks B^4, B^8, B^{16}, B^{32} in G to generate a shape deformation map $\delta^{32}_{\text{shape}}$, which is conditioned on an identity latent code ω. We then construct a separate generator G_{motion} to synthesize an expression deformation map δ^{32}_{exp}, which is conditioned on an animation latent code m. $\delta^{32}_{\text{shape}}(\omega) \circ \delta^{32}_{\text{exp}}(m) \circ (\delta^{32}_{\text{shape}})^{-1}(\omega)$ is then applied to warp F^{32}, which represents the output convolutional features of B^{32}, to model the animation. For the lighting representation, the warped convolutional features are first passed into the next three blocks B^{64}, B^{128} and B^{256}, which are conditioned on ω, to synthesize albedo tri-planes. The output convolutional features are then passed into the last two blocks $B^{256}_{S^{(1)}}$ and $B^{256}_{S^{(2)}}$, which are conditioned on both ω and a lighting latent code l to synthesize shading tri-planes. Given certain camera parameters, portraits are rendered by volume rendering from the albedo tri-planes and shading tri-planes. We sample two animation latent codes and two lighting codes for an identity latent code to synthesize two portraits, which are concatenated channel-wise to pass into the video discriminator D_{video} and separately passed into the image discriminator D_{image}. The albedo is not passed into the discriminator.

A potential choice would be to apply spatial deformation to the volume formed by the albedo tri-planes and shading tri-planes. Different from the classic rendering pipeline, which applies lighting effects after the geometric deformation, a common problem with neural radiance fields is that the shading and albedo are fused into the emission. Therefore, the features associated with each point remain fixed in the spatial deformation. And the albedos and shadings are decoded deterministically from the features, the appearance of rendered 3D faces, therefore, remains unchanged, and the lighting effects such as the visibility are unchanged. The animation representation is then required to alter the convolutional features to update the features of the tri-planes during the animation.

Another observation is that the animation refers to the geometric changes, and the [30] architecture enjoys the property of auto-disentanglement of geometry and appearance in the consecutive synthesis blocks. Therefore, to achieve animation control, we only need to alter the convolutional features $F_{\text{geo}}(\omega) = \{F^4(\omega), F^8(\omega), F^{16}(\omega), F^{32}(\omega)\}$ in the first 4 synthesis blocks $B_{\text{geo}} = \{B^4, B^8, B^{16}, B^{32}\}$, which are responsible for the geometry. Moreover, inspired by [27], to control different animations, we construct an additional animation latent space to parameterize the alteration. The original albedo latent space is then transformed into an identity latent space, which contains all information expected for animation and lighting.

We propose constructing interpretable alteration for convolutional features. Observing the connection between tri-planes and three orthogonal views as in [49], [52], we expect to extend this connection to the convolutional features such that the alteration can be interpreted as altering three orthogonal views. We roll out the convolutional features as in [60] to establish pixel-wise alignment between the convolutional features and tri-planes, as shown in Fig. 3. We then alter the convolutional features through 2D deformations, which can be interpreted as 2D deformations on three orthogonal views. We condition the generation of such 2D deformations on a motion latent code $m \in \mathbb{R}^{128}$ sampled from the standard Gaussian distribution.

Since the convolutional features F^{32} lead to the generation of 3D identities, they are assumed to already contain the shape information. The 2D deformations which represent the animation on the convolutional features therefore should be relative to both the conditioned motion latent codes and the underlying shape in the convolutional features. It then raises a concern that due to the additional conditioning from the underlying shape, the same motion latent codes may not correspond to similar motions on different identities.

To resolve this issue, we explicitly decompose the deformation into an expression deformation in an assumed template space shared by all identities and an invertible transformation that transforms the template to each instance and vice versa. Such assumed template space and the decomposition are conceptually similar to the shape and expression decomposition in the morphable models [34], but we deal with a learned implicit animation representation instead of explicit

vertices deformation. Specifically, we propose to decompose the 2D deformations on the convolutional features for each identity as the composition of 2D expression deformation maps $\delta^{32}_{\text{exp}} : \mathbb{R}^2 \mapsto \mathbb{R}^2$ in an assumed template space shared by all identities, along with 2D shape deformation maps $\delta^{32}_{\text{shape}} : \mathbb{R}^2 \mapsto \mathbb{R}^2$, which transforms the template to each instance. For each identity, we need to first transform the convolution features from the instance space into the template space, then impose the expression deformation, and finally transform the deformed convolution features from the template space into the instance space. Therefore, the final fused 2D expression deformation maps for each identity is then given as $\delta^{32}_{\text{shape}} \circ \delta^{32}_{\text{exp}} \circ \delta^{32^{-1}}_{\text{shape}}$, as shown in Fig. 3. Notice that, δ^{32}_{exp} depends solely on the animation latent space, while $\delta^{32}_{\text{shape}}$ is dependent solely on the identity latent space. In this way, the same motion latent code is guaranteed to generate the same expression deformation in the template space. As long as the shape deformation does not embed any expression deformation, which is further discussed in Sec. III-C, such decomposition then guarantees that the same motion latent code produces similar expressions on different identities.

In practice, we construct an independent 2D generator G_{motion} [30] to synthesize the 2D deformation map δ^{32}_{exp} from a motion latent code $m \in \mathbb{R}^{128}$ sampled from the standard Gaussian distribution. $\delta^{32}_{\text{shape}}$ is synthesized by transforming first 4 synthesis blocks in G with a newly added 1×1 convolution [30], as shown in Fig. 3. Compared to representing the 2D deformations on the convolutional features without this decomposition, this design choice also gives us benefits to ensure geometry consistency during animation, which will be elaborated on in Sec. III-C3, and improve the generation quality and diversity. We further observe an interesting property of the learned animation: the same δ^{32}_{exp} results in semantically similar but not exactly the same expressions on different identities due to different non-linear shape deformation $\delta^{32}_{\text{shape}}$. It makes it easier to transfer the expression without losing personalities. In contrast, if we leverage off-the-shelf estimators (e.g., [17]) to estimate 3DMM coefficients to act as the motion latent codes, besides less expressive animation space of morphable models, the ability of synthesizing personalized motion with the same motion latent code will be lost.

Due to the skip connection architecture of [30], we also conduct deformation on intermediate tri-planes. Fig. 3 illustrates the whole generation process. For further details, please refer to the supplementary.

C. Supervision from In-the-wild Videos

Compared to 3D spatial deformation, 2D deformation on the convolutional features cannot be trivially guided by external priors (e.g., [34]), and thus is learned from data. However, it is challenging to ensure the learned expression deformations correspond to expression animation instead of random deformation. Image data cannot disambiguate it, and available large-scale 4D scan data cannot be used for photo-realistic generation. Therefore, we leverage video data where differences between two frames for a single identity are mainly composed of expression animation.

To ensure diverse and realistic generation, we collect available large-scale video data in the wild without assuming the existence of neutral expressions or synchronous motions across different identities. We also aim for motion synthesis beyond the expressive scopes of existing morphable models. Therefore, we learn the animation latent space from scratch. We adopt a video discriminator D_{video} for learning the animation besides the original image discriminator D_{image}.

1) Training Objectives: Specifically, for real data, we sample a pair of RGB frames from a video sequence with an adaptive sampling strategy as introduced in Sec. III-C2. For fake data, we randomly sample two different animation latent codes from the standard Gaussian distribution. We then sample two different lighting latent codes and two different camera parameters from the dataset, and an identity latent code from the standard Gaussian distribution to generate a pair of facial images. They are passed separately into D_{image} with camera parameters and SH coefficients as conditions. Each pair of images is concatenated into 6-channel features to be passed into D_{video} with their concatenated camera parameters and SH coefficients as conditions for training, as shown in Fig. 3.

Same as [27], we use the non-saturating GAN loss with R_1 regularization [38], density regularization, and lighting regularization for training. Besides, we also apply the commonly used minimal constraint over the deformation maps, as in other works [58], [61], defined as:

$$\mathcal{L}_{\text{minimal}} = \alpha||\delta^{32}_{\text{shape}}||_2 + \beta||\delta^{32}_{\text{exp}}||_2,$$

where α is set to 1, and β is set to 0.1.

2) Adaptive Sampling: We observe that, in the in-the-wild video data, the expression distribution is not uniform and biased significantly towards certain modes, such as an expression status when listening. Besides, since the animation is learned through the differences between two sampled frames in video sequences, we argue that the differences between two sampled frames should be significant, such that the effects of the traverse of the animation latent space are obvious.

Specifically, in a video sequence, we calculate the distance between any two frames by measuring the distance of the corresponding reconstructed meshes by [37]. We then apply a greedy algorithm to split frames into groups. Initially, all the frames are ungrouped. To create a new group, we choose the next ungrouped sample as the center for the group, which will include all the frames whose distances are below an empirical threshold $\tau = 1.3$ to the center frame. This process repeats until every frame is in a group. The sampling of the first frame is achieved by firstly sampling a group uniformly and then sampling a frame in the group uniformly, such that the sampling will not over-emphasize the largest group. After the settlement of the first frame, the second frame is sampled with the probability proportional to the distance to the first frame, where we do not use groups anymore here. Notice that such sampling guidance is rough but effective in practice, because we do not want to rule out the possibility of sampling any two frames.

3) Consistent Animation: Despite the benefits given by the video data, it comes with the challenge of noises from uncertainty in estimated camera parameters, especially for cases

where heads have large angles of roll or eyes are closed. Since the discriminators are not truly 3D-aware, there is room for ambiguity in the regions that appear in one frame but disappear in another. Therefore, the learned animation usually possesses undesirable distortions, such as making heads sometimes larger and sometimes smaller, as observed in [63]. Thanks to the interpretability of our animation representation and the assumed template space, the face region in the template space corresponds to a deterministic area in δ^{32}_{exp}. After training, we empirically define a mask M similar to [52] in the template space to retain deformation only on faces across different identities. Specifically, the generated δ^{32}_{exp} is first masked as $\delta^{32}_{\text{exp}} \odot M$, and then fed into the generation process.

IV. Applications

With an animatable and relightable 3D human face generative model in hand, we here discuss two potential applications: 1) cross-reenactment, where we use a reference human face to drive the generated ones; 2) domain transfer, where we extend the human face generative model into other domains.

A. Image-Driven Cross-reenactment.

After the generative model converges, we can then use the learned animation latent space to drive the generated identities. We could learn an additional encoder to map images into motion latent codes for reenactment. However, considering the enormous efforts (e.g., [9], [17]) devoted before to map images or audios into the coefficients of the morphable model, we illustrate a simpler design here to learn an additional encoder to map coefficients of the morphable model [34] into our animation latent space. Notice that neither should this be perceived as the only way to drive the generated identities, nor should the conclusion that our animation space is confined within the animation space of 3DMM be assumed. However, it turns out that we can even enrich the expressiveness of the animation space of morphable model with our animation latent space.

Specifically, we first use the generative model to generate a batch of frontal images I with randomly sampled identity latent codes, lighting latent codes, and animation latent codes m. We then leverage an off-the-shelf estimator [17] to estimate their corresponding expression and jaw coefficients c from images. The coefficients are then passed to our encoder E to predict the animation latent codes $\hat{m}$. With the same identity and lighting latent codes, but different animation latent codes $\hat{m}$, the generative model then produces frontal images $\hat{I}$. The training objective is then defined as:

$$\mathcal{L} = ||I - \hat{I}||_1 + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}}(I, \hat{I}),$$

where $\lambda_{\text{LPIPS}} = 1$ and $\mathcal{L}_{\text{LPIPS}}$ denotes denotes the perceptual loss [70].

It is noteworthy that even though the original estimator is unable to represent certain expressions like pouting, the learned encoder is able to express them through mapping its biased coefficients into our animation latent space thanks to our pure image-based supervision, as shown in Fig. 1 and Fig. 4. Notice that while we take FLAME [34] as an example here, the paradigm generalizes to other morphable models as long as the corresponding estimators exist.

B. Adaptation to Novel Domains

After training on a domain with rich video data, we draw inspirations from [31], [69] to extend the converged model into novel domains, such as cartoon characters and statues, with pure generated images.

For domains supported by [66], we first ask the generative model to generate a batch of $1,000$ images $I_a = G(\omega, l, G_{\text{motion}}(m))$ in the source domain with randomly sampled identity latent codes ω, lighting latent codes l, animation latent codes m, and camera parameters. They are then transformed into images in the target domain as I_b by a style transfer method [66]. To preserve the 3D-aware, animatable, and relightable properties of the generator, transformed images in the target domain should match the source images in aspects of expression, lighting conditions, and camera parameters. Since the style transfer method is not robust at preserving the lighting conditions, we relabel the lighting conditions of images in the target domain as $\hat{l}$ with an estimator [73], which we empirically find it to be robust on humanoid domains. We then apply the reconstruction and adversarial loss in [69] to fine-tune G only, while fixing G_{motion}, to achieve the domain transfer. The training objective is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{LPIPS}}(G(\omega, \hat{l}, G_{\text{motion}}(m)), I_b) + \lambda_{\text{Adv}}\mathcal{L}_{\text{Adv}},$$

where $\lambda_{\text{Adv}} = 0.05$, and $\mathcal{L}_{\text{Adv}}$ denotes the non-saturating adversarial loss [30].

For other domains, as in [31], we leverage a diffusion model [47] with a prompt "an FHD photo of face of {target domain}" to transform generated images into target domains. However, we find that the diffusion model is not robust to maintain expression and lighting conditions. The lighting conditions of images in the target domain can be similarly relabeled with a robust estimator [73], but the expression constraint cannot be easily met. What introduces more challenges is that for certain domains, such as statues, it is almost impossible to have transformed images with expressions, except for the neutral one, even though we explicitly insert words, such as "smiling", into the prompt. Therefore, we modify the initial generation such that the sampled animation latent code m is fixed and corresponds to the neutral expression. Also, the prompt used in the diffusion model is explicitly modified as "an FHD photo of neutral face of {target domain}". After applying filtering similar to [31], we can then apply the same training objective to fine-tune G only.

V. Experiments

A. Implementation Details

To facilitate convergence, we first train an unconditional generation backbone of [6], where convolutional features are rolled out, on the FFHQ dataset [29] with the Frechet Inception Distance (FID) [24] of 4.7.

We then fine-tune on our trained backbone to train our proposed animatable and relightable model with a combined human head dataset of VFHQ [62] and CelebV-Text [67],

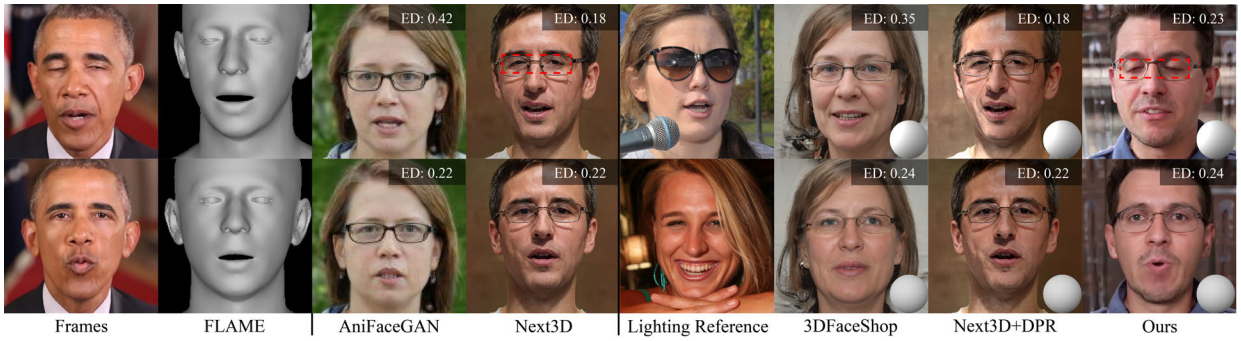


Fig. 4: Qualitative evaluation for animation and relighting. Given the driving frames in the first column, we show their correspondingly reconstructed meshes based on an off-the-shelf detector [17] in the second column. We then demonstrate driven results for animation-only methods in the next two columns. For both animatable and relightable methods, given lighting references in the fifth column, we display driven results rendered under novel lighting conditions in the last three columns. Red rectangles emphasize on the animated eyes region. The calculated expression distances (ED) against the driving frames are shown on the top-right corners of generated results.

Fig. 5: Visualization of the ablation study for the temporal consistency during the linear interpolation of the top and bottom expressions. Textures along straight fixed lines on the hairline (red) and neck (blue) areas are displayed. Ideally, they should be the same vertically.

containing $\sim 60k$ high-quality in-the-wild video sequences. Following the convention of training generative models [6], [29], [30], we do not split the dataset into training and validation set. Besides, during evaluation, we do not use these datasets seen in the training anymore.

We combine scripts in [3], [6], [27] to process the dataset. Specifically, given hundreds of thousands of video sequences, we first apply [21], [22] to crop and estimate camera parameters for each frame. Specially, frames where the resolution of detected largest face is less than 512 are abandoned since we only want high-quality images. Frames where visible hands are detected by [36] with confidence over 0.75 are also abandoned. We then apply [37] to detect the FLAME coefficients [34] for each frame. Videos whose standard deviation of expression coefficients for all frames is less than 1.5 or standard deviation of jaw coefficients for all frames is less than 0.02 are abandoned since we only want to retain video sequences with apparent motions. Finally, as in [27], we use [73] to label the lighting conditions for each remaining frame. The optimizer setup [32] is the same as [6], [27], and we train it for $\sim 5M$ steps with 4 Tesla A100 GPUs. The batch size is set to 16. The encoder in Sec. IV-A is trained by [32] with a learning rate of 0.001 for one day with a batch size of 4 on an RTX 3090 GPU. As to the domain transfer, we have experimented with the domains of cartoon characters and statues. The optimizer and training setup are the same as that [69]. We train it for $\sim 20k$ steps with 4 Tesla A100 GPUs and set the batch size to 16.

Method	Animation Metrics			Lighting Metrics	
	AED↓	APD↓	ID↑	LE↓	LS↓
AniFaceGAN	<u>0.25</u>	0.049	<u>0.73</u>	/	/
Next3D	*0.23*	*0.032*	0.71	/	/
GenHead	0.29	0.047	0.56	/	/
ShadeGAN	/	/	/	1.0714	0.2149
EG3D+DPR	/	/	/	0.7424	0.1594
NFL	/	/	/	<u>0.6377</u>	0.1455
FaceLit	/	/	/	0.6420	**0.1207**
3DFaceShop	0.26	0.042	*0.79*	**0.5950**	*0.1208*
EG3D+SF	**0.21**	<u>0.033</u>	0.25	0.9935	0.2191
Ours	<u>0.25</u>	**0.030**	**0.80**	*0.6263*	<u>0.1317</u>

TABLE I: Quantitative evaluation for animation and relighting. "NFL" refers to NeRFFaceLighting, and "EG3D+SF" refers to EG3D+StyleFlow. We bold the best method, italicize the second-best method, and underline the third-best method.

Methods	TI ↓	FID ↓	AED↓	APD↓	ID↑
Ours w/ M.F.	0.0847	<u>13.97</u>	**0.23**	0.033	0.73
Ours w/o Dec.	0.0725	15.61	0.26	<u>0.032</u>	0.76
Ours w/ Uni.	**0.0568**	14.95	0.27	0.035	*0.79*
Ours w/o M.	<u>0.0702</u>	**13.62**	*0.25*	*0.031*	*0.79*
Ours	*0.0674*	**13.62**	*0.25*	**0.030**	**0.80**

TABLE II: Ablation study on different choices of the animation representation for generation quality and animation capability. "TI" denotes the temporal inconsistency. We bold the best method, italicize the second-best method, and underline the third-best method.

B. Quantitative Evaluation

To evaluate the quality of animation, we follow the evaluation protocol in [52] and compare our method with AniFaceGAN, Next3D and GenHead [13], [14], as other state-of-the-art animatable 3D generative models, based on the Average Expression Distance (AED), Average Pose Distance (APD), and Identity Consistency during animation (ID). For each method, we randomly sample 500 identities and animate each with randomly sampled 20 FLAME parameters of expressions and poses from the FFHQ dataset. Then, we estimate the FLAME parameters for these 10000 generated images and the average distances between the driving FLAME parameters and the reconstructed ones. For identity consistency, we randomly sample 2000 poses, 2000 sets of FLAME parameters, and 1000 identities. Then we randomly select two poses and two sets of FLAME parameters for each identity, generating a total of 1000 image pairs. We calculate consistency metric using a pre-trained Arcface model [12] for each image pair and report the average result.

For relighting, we follow the evaluation protocol in [27] and evaluate our method ShadeGAN [43], EG3D+DPR [73], NeRFFaceLighting [27], and FaceLit [46], as other relightable 3D generative models, based on the Lighting Error (LE) and Lighting Stability (LS). For each method, we sample 1000 real images from the FFHQ dataset and ask the model to generate 1 corresponding fake image for each real image with the same lighting condition. We use the off-the-shelf lighting estimator [17] to estimate the lighting conditions for each pair, and measure and average the distance between them for calculating the lighting error. For the lighting stability, we sample 1000 real images from the FFHQ dataset and ask the model to generate 100 corresponding fake images for each real image with the same lighting condition. We use the off-the-shelf lighting estimator [17] again to estimate the lighting conditions for each generated 100 fake images set, and measure and average the standard deviation among them. In terms of both animation and relighting, we use the same metrics introduced before to compare our method with 3DFaceShop [54] and EG3D+StyleFlow [2]. For our method, we transform the coefficients of Flame [34] into our animation latent codes by the proposed encoder in Sec. IV-A.

As shown in Table. V-B, our method achieves the best APD and ID metrics, second-best LE metrics, and third-best AED and LS metrics. As discussed in [27], 3DFaceShop sacrifices relighting capability on regions outside of faces for consistency despite its best lighting error metrics. As shown in Sec. V-C, our method more consistently synthesizes the motion details, which the off-the-shelf detector is, however, not sensitive to.

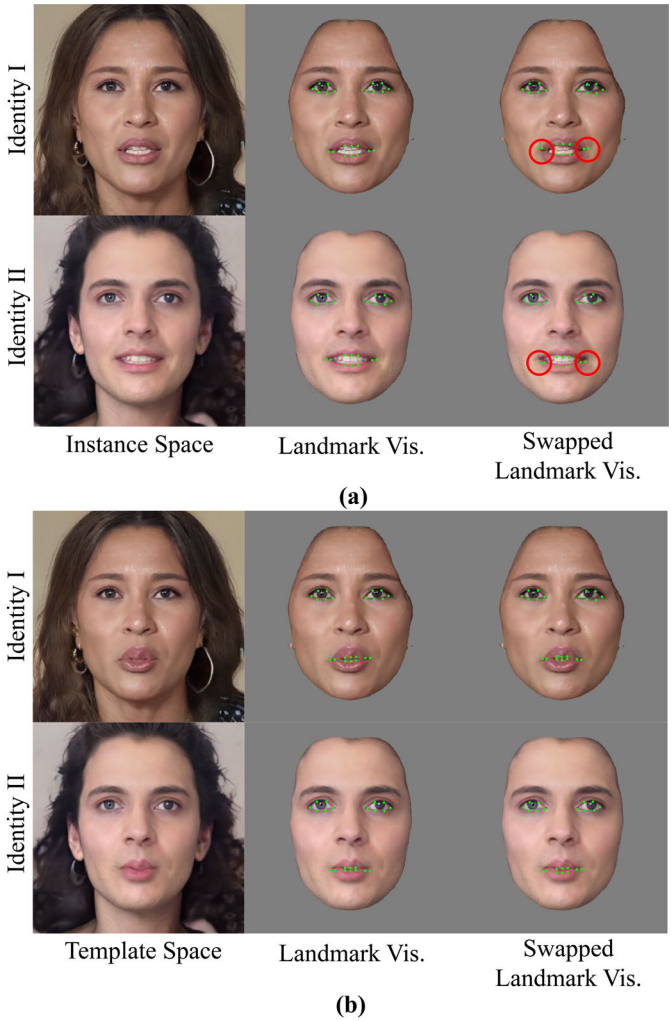


Fig. 6: Demonstration of the learned shape deformation $\delta^{32}_{\text{shape}}$ and assumed template space. We visualize two identities (denoted as "Identity I" and "Identity II"). **(a)** We demonstrate their rendered frontal images without imposing any expression or shape deformations. We further detect and visualize the landmarks of the rendered images in the second column. To put more emphasis on the face region, we mask out the irrelevant regions, including the hair and background, and focus on the eyes and mouth. In the third column, we visualize the landmarks detected on the other identity on the rendered image to highlight the differences between the landmarks of these two identities. Red circles are used to highlight the misalignment of swapped landmarks. **(b)** We impose the $\delta^{32^{-1}}_{\text{shape}}$ alone to transform each identity from the instance space into the template space. As in (a), we visualize the rendered frontal images, landmarks and the swapped landmarks.

C. Qualitative Evaluation

Fig. 4 provides a qualitative comparison of our method against other methods. For animation-only methods, we com-

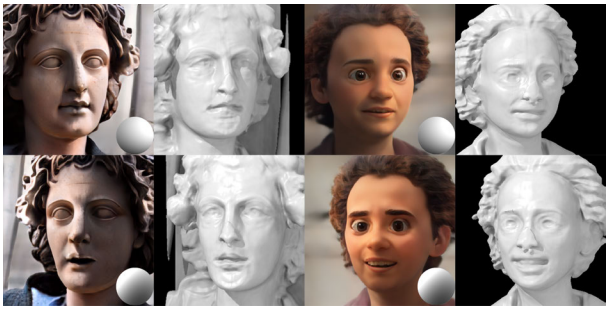


Fig. 7: Demonstration of generated animatable and relightable statues and cartoon characters. The portraits in the same rows have the same expressions. The corresponding geometry is visualized next to the portrait.

pare it with AniFaceGAN and Next3D. For both animatable and relightable methods, we compare it with 3DFaceShop and Next3D+DPR, where we apply [73] on animated results of Next3D. For our method, we encode the driving frames into coefficients by [17] and transform them into our animation latent codes by our proposed encoder.

Our method successfully synthesizes more realistic lighting effects, such as shadows, and more consistent motion details, such as closed eyes and pouted mouth. However, the estimator gives higher expression distances. This bias partly explains our slightly worse metrics in Sec. V-B. We show more results in Figs. 9, 10 and 8, which demonstrate the superior animation and relighting quality of our method.

For the domain transfer, we demonstrate our transferred model on two novel domains, i.e., cartoon characters and statues, in Fig. 7. Our method can synthesize realistic heads in the corresponding domains and achieve effective animation and relighting control. Please find more results in the video.

D. Ablation Study

To validate our proposed animation representation, we compare our design choices against various baselines. Since the task of PV3D [63] is different from ours, we compare with it by replacing our animation representation with PV3D's motion features, denoted as "Ours w/ M.F.". We abandon the proposed decomposition of 2D deformations and directly learn it on the convolutional features, denoted as "Ours w/o Dec.". Instead of the proposed adaptive sampling strategy, we train our model with uniformly sampled frames, denoted as "Ours w/ Uni.". Finally, we evaluate our full model but without applying the masking, denoted as "Ours w/o M.".

Quantitatively, we evaluate these baselines regarding temporal inconsistency while animating the expression, generation quality, and animation metrics introduced before. For the temporal inconsistency, we randomly sample $1,000$ identities and sample 100 expressions for each identity. We then use an estimator [17] to calculate the standard deviation of shape parameters for each identity and report the average result. We evaluate the generation quality based on the FID [24]. Due to the differences of datasets, we do not compare the generation quality with those methods (e.g., 3DFaceShop, AniFaceGAN, Next3D) that rely on an image dataset and cannot benefit from the video dataset. As shown in Table II, our full model achieves the best generation quality and expressive capabilities of representing expressions while maintaining good temporal and identity consistency.

Qualitatively, following [54], we evaluate the temporal consistency during continuous animation in Fig. 5, by comparing our full method with aforementioned baselines. Clearly, only with the template space assumption and the masking can temporal consistency be guaranteed. Besides, in Fig. 6, we also visualize the effects of our proposed decomposition of 2D deformation maps, i.e., decomposing it as the composition of shape and expression deformations. Notice that, for "Identity I" and "Identity II", they have different mouth shapes which incur the misalignment of swapped landmarks. Suppose we only learn a 2D expression deformation map which is invariant of the identity, even though such a deformation guarantees the same expression for all identities, it cannot handle such misalignment of facial parts of different identities. In contrast, after imposing the shape transformation to transform each identity into the template space, their mouth shapes are then roughly aligned, which makes the 2D expression deformation maps valid. Notice that our method automatically learns such a template space without any supervision.

VI. Conclusion, Limitations, and Future Work

Common solutions for consistently animating realistic 3D generative heads based on neural rendering rely on deforming 3D points guided by the classic morphable models, which, however, are incompatible with the lightweight and realistic relighting method. In contrast, we have presented a compatible animation representation with the relighting method to achieve both animatable and relightable photo-realistic 3D head generation. We demonstrate how to train such representations with easily accessed in-the-wild video data and achieve detailed and smooth motion synthesis. We further show that our method is still compatible with the coefficient spaces of other morphable models for driving generated avatars. Our method can be extended to domains with no video data for a broader range of animatable and relightable head generation.

One of our limitations is that our model still cannot enable animation on regions outside of faces, such as hairs, and subtle or extreme expressions, such as iris movement in the leftmost column of Fig. 10. However, since our animation space is learned from easily accessed video data, it has great potential to represent free motion. As future work, it would be interesting to explore how to capture the wider animation scope of human heads. It will also be helpful to explore how to train an encoder to directly map images, which have richer information than pre-computed coefficients, to our animation latent space. It will also be interesting to explore how to train an efficient encoder to achieve real-time animatable and relightable single-shot 3D reconstruction. Besides, static portrait animation, relighting and editing could be misused to generate fake videos to tarnish the reputation or perform other illegal purposes, causing a societal threat. We do not condone such behaviors.

Fig. 8: More results of synthesized animation and relighting results. Given the source identities in the first column, we show the generated results which transfer the motion, pose and lighting in the target driving images in the first row.

REFERENCES

[1] R. Abdal, H.-Y. Lee, P. Zhu, M. Chai, A. Siarohin, P. Wonka, and S. Tulyakov. 3DAvatarGAN: Bridging domains for personalized editable avatars. arXiv preprint arXiv:2301.02700, 2023.

[2] R. Abdal, P. Zhu, N. J. Mitra, and P. Wonka. StyleFlow: Attribute-conditioned exploration of StyleGAN-generated images using conditional continuous normalizing flows. ACM Trans. Graph., 40(3):21:1–21:21, 2021.

[3] S. An, H. Xu, Y. Shi, G. Song, U. Y. Ogras, and L. Luo. PanoHead: Geometry-aware 3D full-head synthesis in 360deg. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 20950–20959, June 2023.

[4] A. Bergman, P. Kellnhofer, W. Yifan, E. Chan, D. Lindell, and G. Wetzstein. Generative neural articulated radiance fields. Advances in Neural Information Processing Systems, 35:19900–19916, 2022.

[5] Z. Cai, K. Jiang, S.-Y. Chen, Y.-K. Lai, H. Fu, B. Shi, and L. Gao. Real-time 3D-aware portrait video relighting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6221–6231, June 2024.

[6] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. D. Mello, O. Gallo, L. J. Guibas, J. Tremblay, S. Khamis, T. Karras, and G. Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022, pp. 16102–16112. IEEE, 2022.

[7] E. R. Chan, M. Monteiro, P. Kellnhofer, J. Wu, and G. Wetzstein. pi-

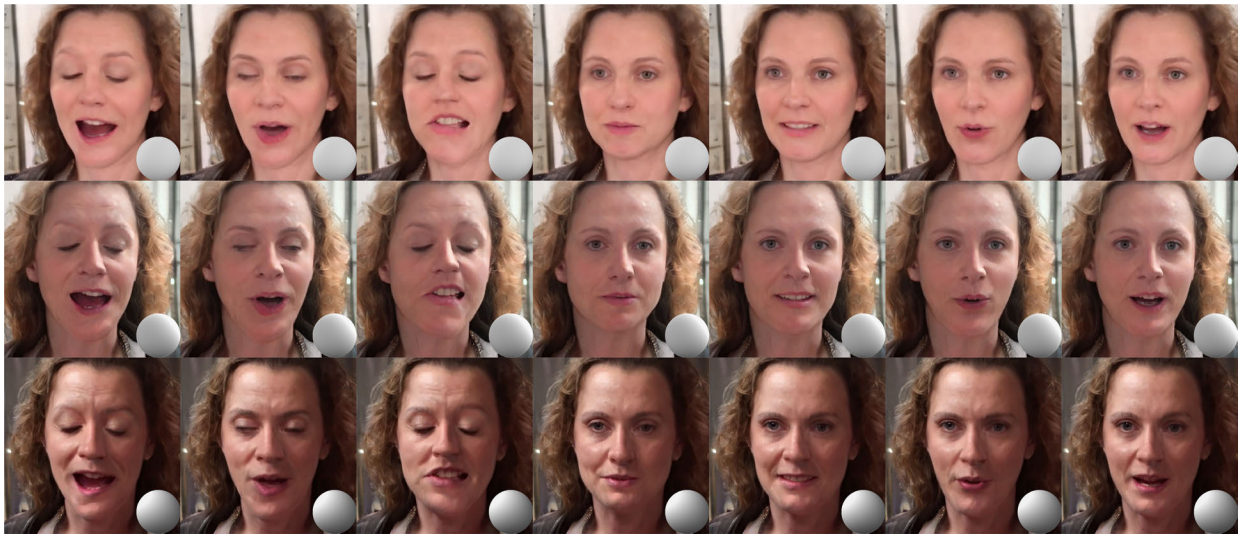


Fig. 9: Selected examples of expressions generated by our method for the same identity under different lighting conditions. For the same rows, the generated results share the same lighting conditions. For the same columns, the expressions are identical, while the camera parameters are permuted.

Fig. 10: More examples of synthesized animation and relighting results. Given the driving frames in the first row, we show the generated driven results of different identities across row and different lighting conditions across column.

GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5799–5809, 2021.

[8] X. Chen, Y. Deng, and B. Wang. Mimic3D: Thriving 3D-aware GANs via 3D-to-2D imitation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023.

[9] Y. Chen, J. Zhao, and W.-Q. Zhang. Expressive speech-driven facial animation with controllable emotions. In 2023 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 387–392. IEEE, 2023.

[10] X. Chu, Y. Li, A. Zeng, T. Yang, L. Lin, Y. Liu, and T. Harada. GPAvatar: Generalizable and precise head avatar from image(s). 2024.

[11] B. Deng, Y. Wang, and G. Wetzstein. LumiGAN: Unconditional generation of relightable 3D human faces. CoRR, abs/2304.13153, 2023.

[12] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4690–4699, 2019.

[13] Y. Deng, D. Wang, X. Ren, X. Chen, and B. Wang. Learning one-shot 4D head avatar synthesis using synthetic data. arXiv preprint arXiv:2311.18729, 2023.

[14] Y. Deng, D. Wang, and b. Wang. Portrait4D-v2: Pseudo multi-view data creates better 4D head synthesizer. arXiv, 2024.

[15] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong. Disentangled and controllable face image generation via 3D imitative-contrastive learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5154–5163, 2020.

[16] Y. Deng, J. Yang, J. Xiang, and X. Tong. GRAM: generative radiance manifolds for 3D-aware image generation. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022, pp. 10663–10673. IEEE, 2022.

[17] Y. Feng, H. Feng, M. J. Black, and T. Bolkart. Learning an animatable detailed 3D face model from in-the-wild images. ACM Trans. Graph., 40(4), jul 2021.

[18] R. Gal, O. Patashnik, H. Maron, A. H. Bermano, G. Chechik, and D. Cohen-Or. StyleGAN-NADA: CLIP-guided domain adaptation of image generators. ACM Trans. Graph., 41(4), jul 2022.

[19] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial networks. Commun. ACM, 63(11):139–144, 2020.

[20] J. Gu, L. Liu, P. Wang, and C. Theobalt. StyleNeRF: A style-based 3D aware generator for high-resolution image synthesis. In International Conference on Learning Representations, 2022.

[21] J. Guo, X. Zhu, and Z. Lei. 3DDFA. https://github.com/cleardusk/3DDFA, 2018.

[22] J. Guo, X. Zhu, Y. Yang, F. Yang, Z. Lei, and S. Z. Li. Towards fast,

accurate and stable 3D dense face alignment. In Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[23] Y. Guo, J. Cai, B. Jiang, J. Zheng, et al. CNN-based real-time dense face reconstruction with inverse-rendered photo-realistic face images. IEEE transactions on pattern analysis and machine intelligence, 41(6):1294–1307, 2018.

[24] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds., Advances in Neural Information Processing Systems, vol. 30. Curran Associates, Inc., 2017.

[25] A. Hou, M. Sarkis, N. Bi, Y. Tong, and X. Liu. Face relighting with geometrically consistent shadows. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4217–4226, 2022.

[26] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 1501–1510, 2017.

[27] K. Jiang, S. Chen, H. Fu, and L. Gao. NeRFFaceLighting: Implicit and disentangled face lighting representation leveraging generative prior in neural radiance fields. ACM Trans. Graph., 42(3):35:1–35:18, 2023.

[28] K. Jiang, S. Chen, F. Liu, H. Fu, and L. Gao. NeRFFaceEditing: Disentangled face editing in neural radiance fields. In S. K. Jung, J. Lee, and A. W. Bargteil, eds., SIGGRAPH Asia 2022 Conference Papers, SA 2022, Daegu, Republic of Korea, December 6-9, 2022, pp. 31:1–31:9. ACM, 2022.

[29] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, pp. 4401–4410. Computer Vision Foundation / IEEE, 2019.

[30] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of StyleGAN. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, pp. 8107–8116. Computer Vision Foundation / IEEE, 2020.

[31] G. Kim and S. Y. Chun. DATID-3D: diversity-preserved domain adaptation using text-to-image diffusion for 3D generative model. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023, pp. 14203–14213. IEEE, 2023.

[32] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In International Conference on Learning Representations, 2015.

[33] A. Kumar, A. K. Bhunia, S. Narayan, H. Cholakkal, R. M. Anwer, S. Khan, M.-H. Yang, and F. S. Khan. Generative multiplane neural radiance for 3D-aware image generation. arXiv preprint arXiv:2304.01172, 2023.

[34] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. Learning a model of facial shape and expression from 4D scans. ACM Trans. Graph., 36(6):194:1–194:17, 2017.

[35] X. Li, S. De Mello, S. Liu, K. Nagano, U. Iqbal, and J. Kautz. Generalizable one-shot 3D neural head avatar. Advances in Neural Information Processing Systems, 36, 2024.

[36] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, et al. Mediapipe: A framework for building perception pipelines. arXiv preprint arXiv:1906.08172, 2019.

[37] T. Martyniuk, O. Kupyn, Y. Kurlyak, I. Krashenyi, J. Matas, and V. Sharmanska. DAD-3DHeads: A large-scale dense, accurate and diverse dataset for 3D head alignment from a single image. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20942–20952, 2022.

[38] L. Mescheder, A. Geiger, and S. Nowozin. Which training methods for GANs do actually converge? In International Conference on Machine Learning, pp. 3478–3487, 2018.

[39] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. NeRF: representing scenes as neural radiance fields for view synthesis. Commun. ACM, 65(1):99–106, 2022.

[40] M. Niemeyer and A. Geiger. GIRAFFE: Representing scenes as compositional generative neural feature fields. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11453–11464, 2021.

[41] R. Or-El, X. Luo, M. Shan, E. Shechtman, J. J. Park, and I. Kemelmacher-Shlizerman. StyleSDF: High-resolution 3D-consistent image and geometry generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13503–13513, 2022.

[42] X. Pan, A. Tewari, L. Liu, and C. Theobalt. GAN2X: Non-Lambertian inverse rendering of image GANs. In 2022 International Conference on 3D Vision, pp. 711–721, 2022.

[43] X. Pan, X. XU, C. C. Loy, C. Theobalt, and B. Dai. A shading-guided generative implicit model for shape-accurate 3D-aware image synthesis. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds., Advances in Neural Information Processing Systems, vol. 34, pp. 20002–20013. Curran Associates, Inc., 2021.

[44] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3D face model for pose and illumination invariant face recognition. In 2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, pp. 296–301, 2009.

[45] R. Ramamoorthi and P. Hanrahan. An efficient representation for irradiance environment maps. In Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '01, p. 497–500. Association for Computing Machinery, New York, NY, USA, 2001.

[46] A. Ranjan, K. M. Yi, J. R. Chang, and O. Tuzel. FaceLit: Neural 3D relightable faces. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023, pp. 8619–8628. IEEE, 2023.

[47] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, 2021.

[48] K. Schwarz, A. Sauer, M. Niemeyer, Y. Liao, and A. Geiger. VoxGRAF: Fast 3D-aware image synthesis with sparse voxel grids. In Advances in Neural Information Processing Systems, 2022.

[49] R. Shao, Z. Zheng, H. Tu, B. Liu, H. Zhang, and Y. Liu. Tensor4D: Efficient neural 4D decomposition for high-fidelity dynamic reconstruction and rendering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16632–16642, 2023.

[50] I. Skorokhodov, S. Tulyakov, Y. Wang, and P. Wonka. EpiGRAF: Rethinking training of 3D GANs. In Advances in Neural Information Processing Systems, 2022.

[51] J. Sun, X. Wang, Y. Shi, L. Wang, J. Wang, and Y. Liu. IDE-3D: interactive disentangled editing for high-resolution 3D-aware portrait synthesis. ACM Trans. Graph., 41(6):270:1–270:10, 2022.

[52] J. Sun, X. Wang, L. Wang, X. Li, Y. Zhang, H. Zhang, and Y. Liu. Next3D: Generative neural texture rasterization for 3D-aware head avatars. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023, pp. 20991–21002. IEEE, 2023.

[53] F. Tan, S. Fanello, A. Meka, S. Orts-Escolano, D. Tang, R. Pandey, J. Taylor, P. Tan, and Y. Zhang. VoLux-GAN: A generative model for 3D face synthesis with HDRI relighting. In ACM SIGGRAPH 2022 Conference Proceedings, SIGGRAPH '22. Association for Computing Machinery, New York, NY, USA, 2022.

[54] J. Tang, B. Zhang, B. Yang, T. Zhang, D. Chen, L. Ma, and F. Wen. 3DFaceShop: Explicitly controllable 3D-aware portrait generation. IEEE Transactions on Visualization & Computer Graphics, (01):1–18, 2023.

[55] A. Tewari, M. Elgharib, G. Bharaj, F. Bernard, H.-P. Seidel, P. Pérez, M. Zollhofer, and C. Theobalt. StyleRig: Rigging StyleGAN for 3D control over portrait images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6142–6151, 2020.

[56] P. Tran, E. Zakharov, L.-N. Ho, L. Hu, A. Karmanov, A. Agarwal, M. Goldwhite, A. B. Venegas, A. T. Tran, and H. Li. VOODOO XP: Expressive one-shot head reenactment for VR telepresence. arXiv preprint arXiv:2405.16204, 2024.

[57] P. Tran, E. Zakharov, L.-N. Ho, A. T. Tran, L. Hu, and H. Li. VOODOO 3D: Volumetric portrait disentanglement for one-shot 3D head reenactment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10336–10348, 2024.

[58] E. Tretschk, A. Tewari, V. Golyanik, M. Zollhöfer, C. Lassner, and C. Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In Proceedings of the IEEE/CVF Conference on International Conference on Computer Vision, pp. 12939–12950, 2021.

[59] A. Trevithick, M. Chan, M. Stengel, E. R. Chan, C. Liu, Z. Yu, S. Khamis, M. Chandraker, R. Ramamoorthi, and K. Nagano. Real-time radiance fields for single-image portrait view synthesis. In ACM Transactions on Graphics (SIGGRAPH), 2023.

[60] T. Wang, B. Zhang, T. Zhang, S. Gu, J. Bao, T. Baltrusaitis, J. Shen, D. Chen, F. Wen, Q. Chen, and B. Guo. RODIN: A generative model for sculpting 3D digital avatars using diffusion. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023, pp. 4563–4573. IEEE, 2023.

[61] Y. Wu, Y. Deng, J. Yang, F. Wei, Q. Chen, and X. Tong. AniFaceGAN: Animatable 3D-aware face image generation for video avatars. In NeurIPS, 2022.
[62] L. Xie, X. Wang, H. Zhang, C. Dong, and Y. Shan. VFHQ: A high-quality dataset and benchmark for video face super-resolution. In The IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2022.
[63] E. Z. Xu, J. Zhang, J. H. Liew, W. Zhang, S. Bai, J. Feng, and M. Z. Shou. PV3D: A 3D generative model for portrait video generation. In The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, 2023.
[64] H. Xu, G. Song, Z. Jiang, J. Zhang, Y. Shi, J. Liu, W. Ma, J. Feng, and L. Luo. OmniAvatar: Geometry-guided controllable 3D head synthesis. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023, pp. 12814–12824. IEEE, 2023.
[65] Y. Xu, S. Peng, C. Yang, Y. Shen, and B. Zhou. 3D-aware image synthesis via learning structural and textural representations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18409–18418, 2022.
[66] S. Yang, L. Jiang, Z. Liu, and C. C. Loy. Pastiche master: Exemplar-based high-resolution portrait style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7693–7702, 2022.
[67] J. Yu, H. Zhu, L. Jiang, C. C. Loy, W. Cai, and W. Wu. CelebV-Text: A large-scale facial text-video dataset. In CVPR, 2023.
[68] H. Zhang, T. Dai, Y. Xu, Y.-W. Tai, and C.-K. Tang. FaceDNeRF: semantics-driven face reconstruction, prompt editing and relighting with diffusion models. Advances in Neural Information Processing Systems, 36, 2024.
[69] J. Zhang, Y. Lan, S. Yang, F. Hong, Q. Wang, C. K. Yeo, Z. Liu, and C. C. Loy. DeformToon3D: Deformable 3D toonification from neural radiance fields. In ICCV, 2023.
[70] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 586–595, 2018.
[71] X. Zhao, F. Ma, D. Güera, Z. Ren, A. G. Schwing, and A. Colburn. Generative multiplane images: Making a 2D GAN 3D-aware. In S. Avidan, G. J. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, eds., Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part V, vol. 13665 of Lecture Notes in Computer Science, pp. 18–35. Springer, 2022.
[72] X. Zhao, J. Sun, L. Wang, J. Suo, and Y. Liu. InvertAvatar: Incremental GAN inversion for generalized head avatars. In ACM SIGGRAPH 2024 Conference Papers, SIGGRAPH '24. Association for Computing Machinery, New York, NY, USA, 2024.
[73] H. Zhou, S. Hadap, K. Sunkavalli, and D. Jacobs. Deep single-image portrait relighting. In 2019 IEEE/CVF International Conference on Computer Vision, pp. 7193–7201, 2019.

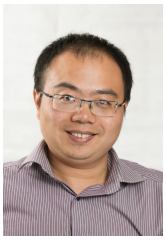
Kaiwen Jiang is a Master student at University of California, San Diego. He received the Bachelor degree in computer science from the Beijing Jiaotong University. His current research interests include computer graphics, and computer vision.

Feng-Lin Liu is a PhD student at Institute of Computing Technology, Chinese Academy of Sciences. His current research interests include computer graphics, computer vision and deep learning.

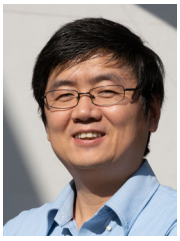
Shu-Yu Chen received the PhD degree in computer science and technology from the University of Chinese Academy of Sciences. She is currently working as an Associate Professor in the Institute of Computing Technology, Chinese Academy of Sciences. Her research interests include computer graphics, computer vision and deep learning.

Pengfei Wan received B.E. degree in electronic engineering and information science from the University of Science and Technology of China (USTC), Hefei, China, and Ph.D. degree in electronic and computer engineering from the Hong Kong University of Science and Technology (HKUST), Hong Kong. His research interests include computer vision, computer graphics and generative models.

Yuan Zhang received the B.S. degree in Automation from Beijing University of Posts and Telecommunications in 2009 and the Ph.D. degree in Pattern Recognition from the Institute of Automation, Chinese Academy of Sciences in 2014. He is currently a Tech Lead Manager at Kuaishou Technology (Kwai Inc.). His research interests include but are not limited to multimodal perception, deep generative models, image & video manipulation, 3D understanding & reconstruction, XR and human digitization.



Yu-Kun Lai received his bachelor's degree and PhD degree in computer science from Tsinghua University in 2003 and 2008, respectively. He is currently a Professor in the School of Computer Science & Informatics, Cardiff University. His research interests include computer graphics, geometry processing, image processing and computer vision. He is on the editorial boards of IEE Transactions on Visualization and Computer Graphics and The Visual Computer.



Hongbo Fu received a BS degree in information sciences from Peking University, China, in 2002 and a PhD degree in computer science from the Hong Kong University of Science and Technology in 2007. He is a Professor in the Division of Arts and Machine Creativity at the Hong Kong University of Science and Technology. His primary research interests fall in the fields of computer graphics and human-computer interaction. He has served as an Associate Editor of The Visual Computer, Computers & Graphics, and Computer Graphics Forum.

Lin Gao received his PhD degree in computer science from Tsinghua University. He is currently an Associate Professor at the Institute of Computing Technology, Chinese Academy of Sciences. He has been awarded the Newton Advanced Fellowship from the Royal Society and the AG young researcher award. His research interests include computer graphics and geometric processing.