NeRFFaceLighting: Implicit and Disentangled Face Lighting Representation Leveraging Generative Prior in Neural Radiance Fields

KAIWEN JIANG, Institute of Computing Technology, CAS and Beijing Jiaotong University, China
SHU-YU CHEN, Institute of Computing Technology, Chinese Academy of Sciences, China
HONGBO FU, School of Creative Media, City University of Hong Kong, China
LIN GAO, Institute of Computing Technology, CAS and University of Chinese Academy of Sciences, China

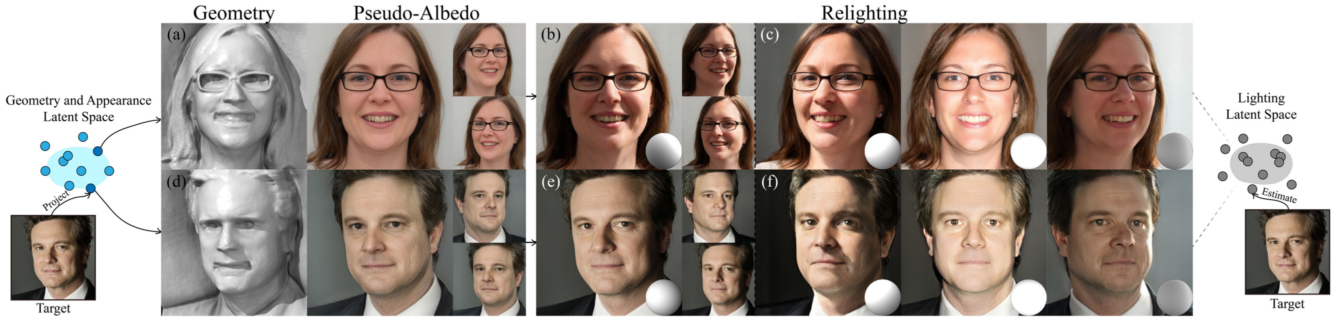

Fig. 1. Our NeRFFaceLighting method achieves disentangled and 3D-aware lighting control with realistic shading and real-time rendering speed. We construct two separated latent spaces: one for geometry and appearance, as shown in the leftmost diagram, and the other for lighting, as shown in the rightmost diagram. Samples are generated by sampling from the geometry and appearance latent space, whose lighting conditions are solely controlled by sampling from the lighting latent space. We demonstrate an example for generated samples in the first row and an example for real portraits in the second row. (a) and (d) show the extracted geometry and the pseudo-albedo. (b) and (e) show the portraits under their own lighting condition with different poses. (c) and (f) show the portraits whose lighting conditions and camera poses are changed simultaneously. The lighting condition in (e) is the same as the input target image. All the lighting conditions are visualized as a sphere placed at the bottom-right corner of the portraits throughout the article. Original image courtesy of Aminatk.

3D-aware portrait lighting control is an emerging and promising domain, thanks to the recent advance of generative adversarial networks and neural radiance fields. Existing solutions typically try to decouple the lighting from the geometry and appearance for disentangled control with an explicit lighting representation (e.g., Lambertian or Phong). However, they either are limited to a constrained lighting condition (e.g., directional light) or demand a tricky-to-fetch dataset as supervision for the intrinsic compositions (e.g., the albedo). We propose *NeRFFaceLighting* to explore an implicit representation for portrait lighting based on the pretrained tri-plane representation to address the above limitations. We approach this disentangled lighting-control problem by distilling the shading from the original fused representation of both appearance and lighting (i.e., one tri-plane) to their disentangled representations (i.e., two tri-planes) with the conditional discriminator to supervise the lighting effects. We further carefully design the regularization to reduce the ambiguity of such decomposition and enhance the ability of generalization to unseen lighting conditions. Moreover, our method can be extended to enable 3D-aware real portrait relighting. Through extensive quantitative and qualitative evaluations, we demonstrate the superior 3D-aware lighting control ability of our model compared to alternative and existing solutions.

CCS Concepts: • **Human-centered computing** → **Graphical user interfaces**; • **Computer systems organization** → Neural networks; • **Computing methodologies** → *Rendering*; Volumetric models;

Additional Key Words and Phrases: Face editing, volume disentangling, lighting manipulation, Neural Radiance Fields, neural rendering

ACM Reference format:
Kaiwen Jiang, Shu-Yu Chen, Hongbo Fu, and Lin Gao. 2023. NeRFFaceLighting: Implicit and Disentangled Face Lighting Representation Leveraging Generative Prior in Neural Radiance Fields. *ACM Trans. Graph.* 42, 3, Article 35 (June 2023), 18 pages.
https://doi.org/10.1145/3597300

1 INTRODUCTION

Recent advances in the field of **Neural Radiance Fields (NeRF)** [Mildenhall et al. 2021] combined with **Generative Adversarial Networks (GAN)** [Goodfellow et al. 2014], known as 3D GAN,

K. Jiang is also with the Beijing Jiaotong University. L. Gao is also with the University of Chinese Academy of Sciences.

This work was supported by grants from the National Natural Science Foundation of China (No. 62061136007 and No. 62102403), the Beijing Municipal Natural Science Foundation for Distinguished Young Scholars (No. JQ21013), China Postdoctoral Science Foundation (No. 2022M713205), and Chow Sang Sang Group Research Fund (No. 9229119).

Authors' addresses: K. Jiang, S.-Y. Chen, and L. Gao (corresponding author), Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences; emails: kevinjiangedu@gmail.com, {chenshuyu, gaolin}@ict.ac.cn; H. Fu, School of Creative Media, City University of Hong Kong; email: hongbofu@cityu.edu.hk.

0730-0301/2023/06-ART35 $15.00
https://doi.org/10.1145/3597300

have achieved remarkable 3D-aware generation for human faces with great details (e.g., [Chan et al. 2022; Deng et al. 2022; OrEl et al. 2022]). Such high-quality 3D generation is promising for designing amateur-level 3D realistic characters as virtual avatars. However, how to enable efficient and disentangled semantic manipulation (e.g., for separately modifying the geometry, appearance, or lighting) remains an open question, especially for lighting. With the manipulation of lighting, artists can create various effects to convey hints and moods in an evocative manner. Make-up artists may also use the lighting to beautify the appearance.

Several methods have been proposed to tackle the problem of manipulating the face lighting in a disentangled manner. For example, ShadeGAN [Pan et al. 2021] uses the Lambertian shading model and explicitly exerts calculated lighting effects from a specific light direction to the predicted color before volume rendering for the manipulation of lighting. GAN2X [Pan et al. 2022] utilizes a 2D GAN prior to obtaining paired images of various viewpoint and lighting conditions and then uses the Phong shading model to solve an inverse rendering problem for the decomposition of albedo, diffuse, and specular all together needed for relighting. These methods achieve reasonable results but are limited to a specific light direction. VoLux-GAN [Tan et al. 2022] also uses the Phong shading model but accepts an environment map as the lighting condition enabled by an augmented dataset obtained from Pandey et al. [2021] with predicted albedo, diffuse, and specular components as adversarial supervision. However, all the above explicit shading modelling methods cannot handle some challenging lighting cases in an efficient way, such as the shadow cast by a pair of glasses, as shown in the top-light case of Figure 6. Besides, ground-truth albedo, diffuse, and specular for as many as hundreds of thousands of subjects are hard to obtain, and using previous methods to predict may also inherit their limitations. There is no denying that an explicit lighting model can achieve highly realistic results under any desired lighting conditions but comes at a great cost of computation, memory, and time consumption, such as the secondary reflected light. It is still an open question about how to explicitly model the lighting in the context of NeRF in an effective and efficient way.

Analogous to relighting, synthesizing face images with realistic appearance and lighting in an efficient way was also a tricky problem before. However, with the introduction of implicit representation as generative neural networks, several methods (e.g., [Karras et al. 2020b, 2021a, b]) are able to achieve this goal in a nearly perfect way. Accordingly, some works (e.g., [Abdal et al. 2021; Deng et al. 2020; Shoshan et al. 2021; Tewari et al. 2020]) have also tried to model the lighting in an implicit way. Specifically, they try to disentangle the lighting in the latent space of 2D GAN from the identity, expression, pose, and other factors. However, since the latent space for 2D GAN is entangled for illumination, shape, and appearance, the modification of lighting condition often leads to the disturbance of geometry and appearance, especially when the camera pose is also changed simultaneously.

In this work, we introduce *NeRFFaceLighting* to explore the implicit lighting representation in the generative NeRF to address all the above limitations. We use **spherical harmonics (SH)** [Ramamoorthi and Hanrahan 2001; Sloan et al. 2003] to describe the white light and enable the control of the lighting effects without affecting the geometry and appearance. We can achieve realistic and efficient lighting results in certain challenging cases (e.g., the shadow cast by a pair of glasses or lighting effects on the hair) and obtain reasonable pseudo-albedo with a general dataset of single-view face images only, as shown in Figure 1. Our work is built upon a recently proposed pretrained 3D GAN, EG3D [Chan et al. 2022], and formulates the decomposition of appearance and lighting as a distillation problem. Specifically, the appearance and lighting are fused together at the start, and we aim at distilling the appearance and lighting into their separated and disentangled representations with efficient fine-tuning. Our approach is based on the following three key insights:

Our first key insight is that the high-level latent codes roughly control the illumination while affecting the geometry and appearance at the same time, though. Thus, to build a disentangled representation for lighting, we construct a separated latent space solely for lighting and append several synthesis blocks that are conditioned on lighting latent codes to construct a shading tri-plane along with the original tri-plane (Figure 3). A shading decoder is designed to predict shading from features sampled from the shading tri-plane and apply the predicted shading to the color predicted by the original decoder from features sampled from the original tri-plane. Thus, the images rendered from colors before applying shading are assumed to be pseudo-albedo, and the images rendered from colors after adding shading are assumed to be portraits with shading, while the images rendered from shading directly are assumed to be the shading component. Besides the original discriminator, we use another discriminator conditioned both on the camera poses and spherical harmonics coefficients to supervise the lighting effect.

However, the major challenge in our implicit representation is the ambiguity caused by the lack of ground truth for the decomposed components (e.g., albedo). In fact, for a specific portrait with shading, there could exist many possible combinations of albedo and shading components. Trivially ignoring such an issue results in degraded lighting accuracy (Table 4) and more shading remaining in the pseudo-albedo components as the residual shading compared to ours (Figure 9(a)). To reduce the ambiguity, our second key insight is that the shading should *generalize* well to different pseudo-albedos when the geometry is held. We observe that EG3D is capable of generating diverse facial images covering wide ranges of lighting and appearance conditions while keeping the geometry roughly unchanged through style-mixing, as shown in Figure 4. This generative prior unleashes the potential of using samples generated by style-mixing to provide supervision for each other so the reflectance on the surface of the pseudo-albedo generally transforms to a desired averaged state, i.e., the residual shadings are removed. The optimization of this disambiguity regularization leads to the removal of many challenging lighting phenomena (e.g., specular lights, over-exposure, directional light, shadows) in the pseudo-albedo, as shown in Figure 12. Moreover, we discover that our lighting results are faithful reflections of the underlying geometry, as shown in Figure 16. However, without an explicit lighting model, we do not rule out the possibility of modelling the appearance in the shading component, which causes the drop of color in the pseudo-albedo components, as shown in Figure 9(b). To alleviate this problem, we introduce similarity regularization, explicitly

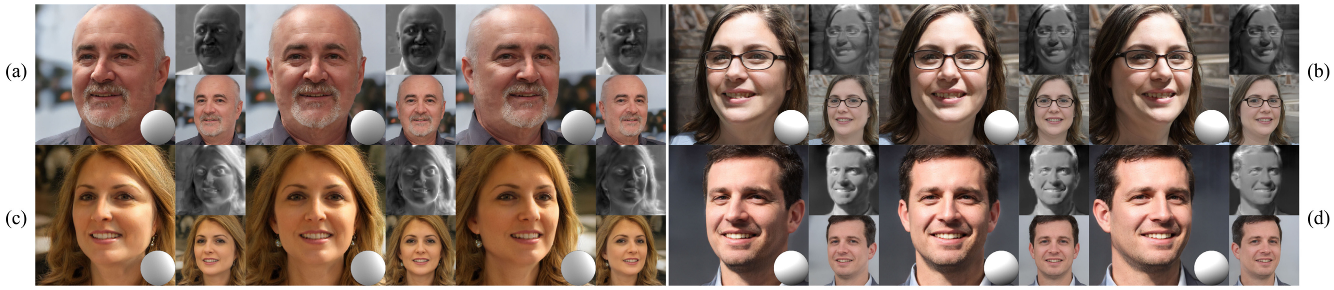

Fig. 2. Demonstration of the lighting control achieved by our method. For each portrait, we place its corresponding shading (upper one) and pseudo-albedo (lower one) components next to it. Notice the specular lights at the top of the head in (a), the shadow cast by a pair of glasses in (b), realistic lighting effects on the hair in (c), and the shadow cast by the head on the neck in (d). Besides these, our method also preserves the geometry and appearance well when comparing the pseudo-albedo components and the final portraits with shading.

requiring that the pseudo-albedo should be visually similar to the portraits with shading based on the ratio image-based rendering algorithm [Shashua and Riklin-Raviv 2001].

Another challenge for the implicit representation is due to the distribution of the training data. For those unseen lighting conditions or those seen ones but at the margin of the distribution, the implicit representation struggles to perform well on them, as shown in the first row of Figure 10. To improve the ability of generalization, our third key insight is that the model should be robust to small *disturbance* of lighting conditions. We thus integrate some disturbance to the above similarity regularization. Our method also can be used for 3D-aware real portrait relighting by projecting real portraits into the latent space of our generator, as shown in the bottom row of Figure 1.

The main contributions are summarized as follows:

- We propose a disentangled implicit representation for portrait lighting described as **spherical harmonics (SH)** in the tri-plane-based generative NeRFand further extend it to enable 3D-aware real portrait relighting.
- We propose the regularization leveraging generative prior to reduce the ambiguity and enhance the ability of generalization for lighting control.
- We conduct extensive experiments and comparisons to show our method achieves state-of-the-art 3D-aware lighting manipulation results.

2 RELATED WORK

Our work is closely related to several topics, including relightable neural implicit representations, 3D-aware neural face image synthesis, and portrait relighting.

2.1 Relightable Neural Implicit Representations

Neural radiance field (NeRF) [Mildenhall et al. 2021] is an emerging technique and used in various applications such as static or dynamic 3D scene reconstruction (e.g., [Li et al. 2021; Müller et al. 2022; Park et al. 2021; Wang et al. 2021]) because of its highly realistic and 3D-consistent rendering results.

However, despite its successful usage for reconstruction, how to manipulate reconstructed scenes is crucial for actual deployment in industries. Lighting is one of the core factors that deserve to be controlled. Several methods (e.g., [Boss et al. 2021; Li et al. 2022; Rudnev et al. 2022; Srinivasan et al. 2021; Wang et al. 2022b; Zhang et al. 2021; Zhao et al. 2022b]) have been proposed to enable relighting in a reconstructed object or scene. However, they are specific to one scene or one object and require retraining whenever the scene or object is changed. EyeNeRF [Li et al. 2022] focuses on modeling human eyes with complex reflectance and fine-scale geometry and achieves reasonable results but is limited to eyes. NeLF [Sun et al. 2021] takes in several face images at different poses and under the same but unknown lighting conditions. It is able to conduct relighting and novel view synthesis. However, it cannot deal with complex structures (e.g., long hair) and needs more than one input face image.

Another group of methods [Pan et al. 2021, 2022; Tan et al. 2022] accomplish the relighting task based on 3D GANs. However, they are either limited to a specific directional light or require an augmented dataset with albedo, diffuse, and specular components, which are not error-free by predicting using previous methods [Pandey et al. 2021] for decomposition. Our work is also based on the 3D GAN for general 3D-aware relighting dealing with full faces (including long hair and neck) and needs only one face image for reconstruction and relighting. We are able to deal with some challenging lighting effects (e.g., shadows cast by the head on the neck, specular lights on the forehead), as shown in Figure 2, and only require a single-view 2D face image dataset [Karras et al. 2021b] with estimated lighting conditions described as SH by an off-the-shelf lighting predictor [Zhou et al. 2019].

2.2 3D-aware Neural Face Image Synthesis

The recent incorporation of **Generative Adversarial Networks (GAN)** [Goodfellow et al. 2014] into NeRF [Mildenhall et al. 2021] allows for 3D-aware face image synthesis.

One group of methods (e.g., [Chan et al. 2021; Schwarz et al. 2020]) for enabling generation in NeRF are to use conditional coordinate-based NeRF, but they suffer from low-resolution outputs. To generate high-resolution images, a group of methods [Gu et al. 2022; Niemeyer and Geiger 2021; Zhou et al. 2021] proposed to output low-resolution features and then pass them into 2D convolution for up-sampling. They further optimize the up-sampling technique to improve the 3D consistency but still have a low-quality geometry representation.

Recently, alternative models and representations (e.g., [Chan et al. 2022; Deng et al. 2022; OrEl et al. 2022; Rebain et al. 2022;

Schwarz et al. 2022; Sun et al. 2022a; Xiang et al. 2022; Xu et al. 2022; Zhao et al. 2022a]) have been explored to feature better geometry, image quality, and faster inference speed. They provide foundations for future works, such as editing attributes (e.g., lighting, geometry, appearance) in a semantic way (e.g., [Jiang et al. 2022; Sun et al. 2022a, b]). Our method enables the disentangled control of lighting with the tri-plane representation [Chan et al. 2022].

2.3 Portrait Relighting

2D portrait relighting based on deep learning is a well-studied domain. Existing methods (e.g., Hou et al. [2021, 2022]; Nestmeyer et al. [2020]; Pandey et al. [2021]; Sengupta et al. [2018]; Sun et al. [2019]; Zhou et al. [2019]) typically require explicit supervision in the image level, such as the normal or multiple images for a single identity under different known lighting conditions generated by synthetic methods or using a professional capture setup. This strict requirement is partly due to the fact that some of these methods (e.g., [Pandey et al. 2021]) recover the 3D information from 2D images for better estimation of lighting and the supervision from multiple images for a single identity under different lighting conditions facilitates the relighting effects in an implicit way (e.g., [Sun et al. 2019; Zhou et al. 2019]) or the decomposition of albedo, geometry, and reflectance components in an explicit way (e.g., [Nestmeyer et al. 2020; Pandey et al. 2021]). The main problem with the above methods is that they either capture a limited number of subjects to provide portraits under different lighting conditions, thus limiting their ability of generalization, or synthesize fake ground-truth images as supervision, which limits their relighting accuracy and realism. Recently, Yeh et al. [2022] proposed to use a virtual light-stage based on advanced computer graphics to mitigate such an issue and fill in the gap between synthetic and real data, achieving reasonable results.

However, the recent advent of 3D GANs opens a new yet promising direction. Its innate 3D representation greatly eliminates the need of estimating the 3D information from 2D images for relighting. However, how to use this 3D representation for relighting remains an open problem. HeadNeRF [Hong et al. 2022] presents a parametric head model conditioned on the lighting latent code. Its disentanglement is achieved by strong supervision provided by its multi-lighting-condition dataset. However, it only allows for limited lighting adjustment because of the insufficient coverage of lighting in its dataset. 3DFaceShop [Tang et al. 2022] tries to incorporate the guidance of parametric head models into a neural face representation and achieves highly consistent disentangled control over expression, lighting, and so on, through volume blending. However, its generation is limited to the expressive power of parametric head models, possessing sub-optimal **Frechet Inception Distance (FID)** [Heusel et al. 2017] values compared to ours. Besides, its disentangled control is only guaranteed for the face region, excluding hair, glasses, and so on. SURF-GAN [Kwak et al. 2022] decouples semantic attributes (including lighting) in the latent space through unsupervised training, but its disentangled control is relatively rough and it is hard to adjust the lighting condition into a specified state.

To achieve arbitrary lighting manipulation, a common practice is to decompose the image into the albedo and shading components. However, the major challenge for faithful lighting control based on such decomposition is the lack of appropriate datasets for supervision. ShadeGAN [Pan et al. 2021] tackles this problem by explicitly modelling the lighting effects and then applying the calculated shading to the predicted pre-cosine albedo to render the final portraits passed into the discriminator. However, ShadeGAN is limited to a specific light direction and a simplified Lambertian shading model. GAN2X [Pan et al. 2022] approaches this problem by leveraging a 2D GAN [Karras et al. 2020b] prior. It uses a 2D GAN to generate multi-view and multi-lighting-condition face images for a single identity and then solves an inverse rendering problem with the Phong shading model. This method achieves reasonable results and is able to lift a 2D GAN to the 3D GAN with disentangled control of lighting but still limited to a specific lighting direction. VoLux-GAN [Tan et al. 2022] uses a 2D relighting model [Pandey et al. 2021] to generate predicted albedo, diffuse, and specular components for a general single-view dataset [Karras et al. 2021b] as augmentation. It uses this supervision in its adversarial losses to achieve plausible decomposition. Nonetheless, all the above methods still fail to deal with some challenging lighting problems, such as the shadow cast by a pair of glasses.

In contrast, our method is able to generate faithful 3D-aware face images along with plausible pseudo-albedo and shading components with the disentangled control of the lighting condition specified by the SH. Different from their explicit modelling, such as the Lambertian shading model or Phong shading model, we adopt an implicit representation for lighting, which is directly optimized using real images with a careful design of regularization to reduce ambiguity and enhance the ability of generalization. Thus, we are able to deal with those challenging cases in an implicit yet efficient and effective way. Besides, we only need a general single-view dataset, which is also a great advantage, considering the difficulty of obtaining truly ground-truth albedo, geometry, and reflectance components. One application of our work is to project real portraits into our latent space and change their lighting conditions for relighting. We train an encoder to facilitate this process and explore some regularization for better projection and relighting performance. Our method further supports rotating the relit faces, since our model is 3D-aware.

3 METHODOLOGY

In this section, we first introduce the preliminaries of our backbone, EG3D [Chan et al. 2022], in Section 3.1. Then, we formalize the structure of our proposed framework in detail to disentangle the lighting with an implicit representation in Section 3.2. The entire pipeline is trained end-to-end by fine-tuning with two dual-discriminators, using the non-saturating GAN loss function [Goodfellow et al. 2014] with R1 regularization [Mescheder et al. 2018], following the training scheme in EG3D. To reduce the ambiguity and enhance the generalization ability, we introduce the regularization in Section 3.3. Furthermore, we introduce how to use our pipeline to conduct 3D-aware portrait relighting in Section 3.4.

3.1 Preliminaries

Since our framework is built on the pretrained EG3D, we first briefly summarize its pipeline here. It uses StyleGAN2 [Karras et al. 2020b] as a backbone to generate multi-channel (96 channels) 2D

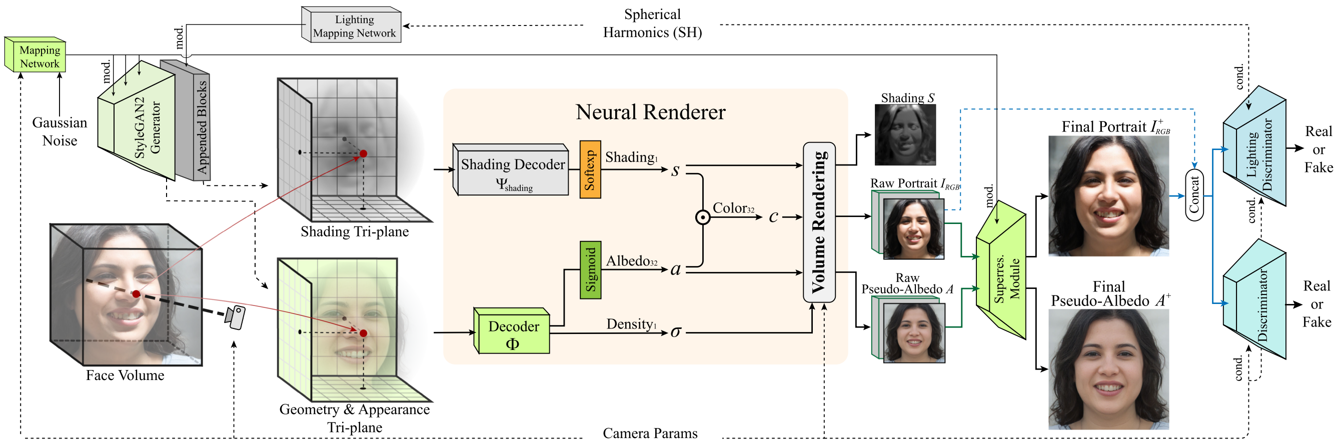

Fig. 3. An overview of our framework. Our pipeline is built on the original pretrained tri-plane representation. We append several synthesis blocks, which are conditioned on the lighting latent codes translated from SH by the lighting mapping network to a generator for constructing shading tri-plane, while the original tri-plane is assumed to be geometry and appearance tri-plane. Features sampled from the original tri-plane are passed through the original decoder Φ to predict density σ and albedo features $a \in \mathbb{R}^{32}$. Features sampled from the shading tri-plane are passed through the shading decoder Ψ_{shading} to predict shading $s \in \mathbb{R}^1$. The final color features are composited as $c = s \odot a$. By volume rendering, the neural renderer produces the shading image, raw portrait with shading, and raw pseudo-albedo. The raw portraits and pseudo-albedo are passed into a super-resolution module for fetching high-resolution portraits and pseudo-albedo. The original dual-discriminator, along with another dual-discriminator that is conditioned on both the camera parameters and lighting conditions, discriminates over the portraits with shading.

features from latent codes w translated from random Gaussian noises and reshape it into three planes (a tri-plane for short) p as an efficient and powerful representation of 3D volume.

Given a specific set of camera parameters, EG3D casts rays into this 3D volume formed by a tri-plane. For each queried position, it retrieves three features (32 channels for each) by projecting the queried position onto each of the three planes via bilinear interpolation and reduces them into a single feature vector (32 channels) by summation. A lightweight decoder, implemented as a small MLP Φ, interprets the reduced feature vectors into densities and color features. These quantities are rendered into feature images $I_{\text{F}} \in \mathbb{R}^{H_l \times W_l \times 32}$ in low resolution through volume rendering. Its first three channels are taken as low-resolution RGB images $I_{\text{RGB}} \in \mathbb{R}^{H_l \times W_l \times 3}$. Furthermore, a super-resolution module takes in the feature images I_{F} and outputs the upsampled high-resolution images $I^+_{\text{RGB}} \in \mathbb{R}^{H_h \times W_h \times 3}$.

The rendering camera parameters are omitted in all following discussions for simplicity. Its dual-discriminator discriminates both I_{RGB} and I^+_{RGB} to enhance the view-consistency.

3.2 Disentangled and Implicit Lighting Representation

It has been known that the fine-level latent codes of StyleGAN2 roughly control the appearance. We observed that the lighting is embedded in it as well, as shown in Figure 4, which indicates the potential for synthesizing highly realistic lighting effects (e.g., shadows from occlusion) in an implicit way, i.e., learning from the dataset. However, the control of lighting is highly entangled with other attributes, especially for the appearance through mixing the fine-level latent codes. Thus, previous techniques such as StyleFlow [Abdal et al. 2021] proposed to manipulate the latent codes in a way such that the lighting is changed as desired but the geometry and appearance are well-preserved. However, such solutions are not very effective at changing the lighting and might disturb the geometry and appearance, as shown in Figure 6. Another group of ideas (e.g., [Deng et al. 2020; Jiang et al. 2022; Shi et al. 2022; Sun et al. 2022a]) is to form separated latent spaces, each of which controls a specific set of attributes. We adopt a generally similar idea and adapt it in the tri-plane representation for lighting control.

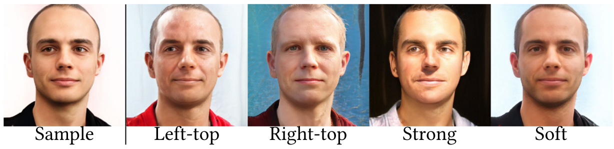

Fig. 4. Demonstration of various residual shadings initially achieved through style-mixing while keeping the geometry roughly unchanged. For a specific portrait, its correlated generated portraits can cover a wide range of residual shadings (e.g., left-top light (Left-top), right-top light (Right-top), strong light (Strong), and soft light (Soft)), but their appearances vary significantly.

Specifically, we construct a separated mapping network that translates SH sh into lighting latent codes w_{sh} to form a latent space that is only responsible for lighting. We then append several synthesis blocks of Karras et al. [2020b] conditioned on w_{sh} to the end of the original generation pipeline of the tri-plane generator to generate *a shading tri-plane* p^{shading} along with the original tri-plane p, as shown in Figure 3. Furthermore, we devise a shading decoder Ψ_{shading} to interpret the features sampled from the shading tri-plane into a shading value $s \in [0, +\infty)$. The original color features predicted by decoder Φ from features sampled from the original tri-plane are viewed as *albedo features* $a \in [0, 1]^{32}$. Thus, the original latent space is expected to be mainly responsible for geometry and appearance. The final color features c are formed as

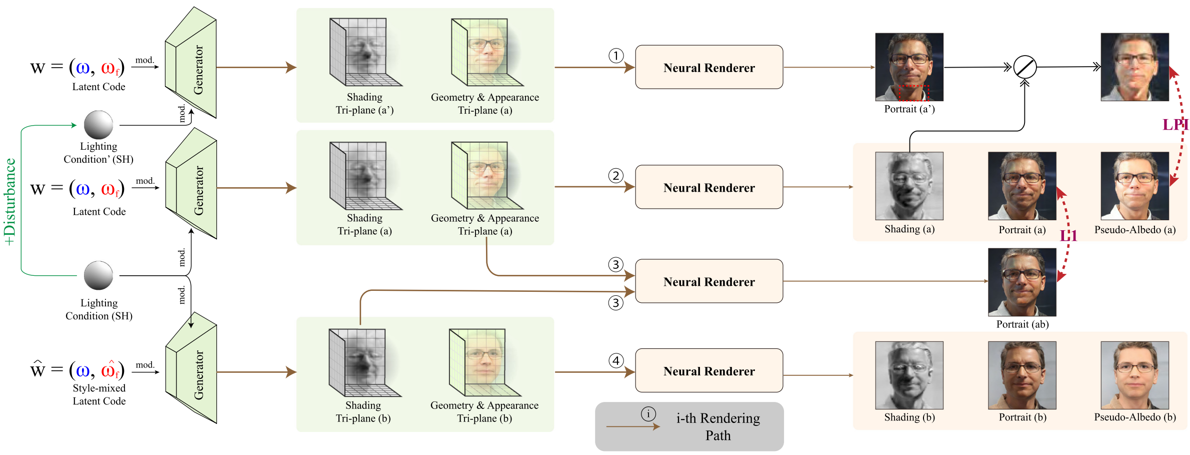

Fig. 5. Illustration of the regularization introduced in Section 3.3. For a latent code w (with ω and ω_f denoting coarse-level and fine-level latent codes, respectively), we first sample another style-mixed latent code $\hat{w}$ (with $\hat{\omega}_f$ denoting randomly sampled fine-level latent codes). We then combine the geometry and appearance tri-plane (a) and the shading tri-plane (b) to generate the portrait (ab) and calculate the L1 metric between it and the portrait (a) to enhance the generalization of the shading component. Besides, we add a small disturbance to the original lighting condition to generate portrait (a'). We apply the ratio image-based rendering algorithm to approximately remove the lighting effects from the portrait (a') with the shading (a) ($\oslash$ denotes the element-wise division) and calculate the LPIPS metric between it and the pseudo-albedo (a). The small disturbance here enhances the generalization ability of our model, since the implicit model could be unstable even for a small deviation from the seen lighting conditions (e.g., the deviated shading on the neck shown in the red rectangle in the portrait (a')).

$c = s \odot a$. Note that the shading tri-plane is dependent on both geometry and appearance, because certain geometric details (e.g., beards, hair details) are missing in the underlying genuine shape (i.e., densities) and modelled in the appearance (i.e., color features) instead. Such effects are shown in Figure 16. More discussion can be found in the supplementary material.

The densities σ predicted by Φ along with these modified color features are rendered into feature images $I_F \in \mathbb{R}^{H_l \times W_l \times 32}$ through volume rendering, which are later passed into a super-resolution module to generate corresponding high-resolution images $I^+_{\mathrm{RGB}} \in \mathbb{R}^{H_h \times W_h \times 3}$. Besides this, the same densities along with the predicted shadings s or albedo features a can be rendered into 2D shading images $S \in \mathbb{R}^{H_l \times W_l \times 3}$ and pseudo-albedo images $A \in \mathbb{R}^{H_l \times W_l \times 3}$, which can be fed into the super-resolution module as $A^+ \in \mathbb{R}^{H_h \times W_h \times 3}$ separately. Note that our training process is based on fine-tuning. Thus, we expect our design can gradually distill the shading information from the original tri-plane into the shading tri-plane. However, since we do not have explicit albedo supervision, the rendered pseudo-albedo from albedo features might be different from traditional concepts.

To ensure the implicitly generated portraits with desired lighting conditions, we utilize the power of conditional GAN. The conditional GAN (e.g., [Karras et al. 2020a; Mirza and Osindero 2014]) can be used to generate samples with specified attributes, such as species (e.g., cats or dogs), in a unified latent space. EG3D uses camera parameters as a condition to alleviate the bias of dataset (e.g., people tend to smile when they face the camera). Similarly, besides the original dual-discriminator D, which is conditioned on camera parameters, we further utilize another dual-discriminator D_{light} [Karras et al. 2020a], which is conditioned on both camera parameters and lighting conditions (namely, SH) to supervise lighting effects, since making the discriminator aware of the camera parameters is crucial to the 3D GAN, as proven in Zhao et al. [2022a] and [Chan et al. 2022]. Following Karras et al. [2020a], we do not use any additional pretrained regressor to supervise the conditions, i.e., SH in our case. During training, these two discriminators are optimized simultaneously. Note that even though in our pipeline, the predicted shading values of the shading decoder are independent of the viewing direction, and we empirically find that the shading tri-plane encodes the view-dependent effects, as shown in the video, by possibly adding a heterogeneous layer in terms of the lightness behind the surface (see the supplementary material for more details).

3.3 Reduce the Ambiguity and Enhance the Generalization

However, without explicit supervision of the albedo and with only optimizing on the portraits with lighting, the decomposition as the albedo and shading is highly ambiguous. Formally, we optimize on the c only, but how s and a compose into it is indefinite. This problem is thus ill-posed, and colorful lights even worsen it. We choose the light to be white light only to alleviate this issue, but the ambiguity problem still exists, especially for shadows. This uncertainty makes the optimization harder, giving rise to worsened lighting accuracy and stability. To address this issue, our major idea is to use specially generated samples to provide supervision for each other. Figure 5 illustrates the whole regularization introduced in this subsection.

Formally, we are seeking to distill the shading information from our pseudo-albedo components into shading components to reduce uncertainty. We observe that even though, from the perspective of a specific sample, its pseudo-albedo component could suffer from various lighting phenomena (e.g., over-exposure, specular lights) initially, its correlated generated samples, which share roughly the same geometry, cover a wide range of textures, including different appearances and residual shadings, as shown in Figure 4. If the correlated generated distribution of residual shadings is roughly complete or symmetric for a certain geometry (e.g., if there exists a strong lighting case, then there exists a soft lighting case), then the averaged residual shading on the pseudo-albedo component should be canceled out. However, the challenge is that we cannot hold the appearance and change the residual shading only for these correlated generated samples if we want to take the average of them. Such a challenge leads us to an indirect but compact and efficient solution, i.e., averaging the shading components explicitly and averaging the residual shading on the pseudo-albedo components implicitly as consequent.

Initially, under some lighting conditions, the shading component of a certain sample may overfit to the residual shading on the pseudo-albedo component, since we optimize on the portrait to match the lighting condition by the discriminator, as seen in the portrait (a) and pseudo-albedo (a) of Figure 5. If we apply a shading component of one of the correlated generated samples to the pseudo-albedo component of such sample, then it fails to generate the similar portrait as the original shading component, since the residual shading on the pseudo-albedo components of correlated generated samples could be different from that of the original sample, as seen when comparing the pseudo-albedo (a) and pseudo-albedo (b) and portrait (a) and portrait (ab) in Figure 5. Thus, the shading component of a certain sample cannot *generalize* to other correlated samples. However, we require that the shading component of a specific sample *generalizes* well to other correlated generated samples to make the shading component match the averaged pseudo-albedo component implicitly. Along with optimizing the generator to produce accurate lighting effects on the portraits with the discriminator, the shading information remaining in the pseudo-albedo components will gradually be distilled to suit the generalized shading component simultaneously. Please find more details in the supplementary material.

In our implementation, we use style-mixing to approximate the generation of different samples with roughly the same geometry but diverse textures, as utilized in Wang et al. [2022a]. For a batch of samples w and sampled lighting conditions sh, their generated images I^{+}_{RGB} are conditioned on both tri-planes p_w and shading tri-planes $p^{\text{shading}}_{(w,\,w_{\text{sh}})}$. Assume samples $\hat{w}$ are style-mixed with w, the loss is defined as:

$$\mathcal{L}_{\text{Cross}} = \left\| I^{+}_{\text{RGB}}(p_w, p^{\text{shading}}_{(w,\,w_{\text{sh}})}) - I^{+}_{\text{RGB}}(p_w, p^{\text{shading}}_{(\hat{w},\,w_{\text{sh}})}) \right\|_1, \qquad (1)$$

which refers to the calculated L1 metric in Figure 5. Note that the variability of the appearance does not matter in this strategy.

However, without an explicit lighting model, we do not rule out the possibility that the appearance information is distilled from the pseudo-albedo components into the shading components. This could result in a drop of color on the pseudo-albedo components, as shown in Figure 9(b). To alleviate this issue, we expect the pseudo-albedo to be visually similar to the portraits with shading. Because of the discrepancy between these two domains, i.e., the pseudo-albedo and the portraits with shading, we adopt a soft constraint here and use the generated shading component to bridge the connection based on the ratio image-based rendering algorithm [Shashua and Riklin-Raviv 2001]. Formally, for samples w and sampled lighting conditions sh, we define the loss as follows:

$$\mathcal{L}_{\text{Sim}} = \mathcal{L}_{\text{LPIPS}}(A(w), I_{\text{RGB}}(w, w_{sh}) \oslash S(w, w_{sh})), \qquad (2)$$

in which $\mathcal{L}_{\text{LPIPS}}$ denotes the perceptual loss [Zhang et al. 2018]. Moreover, the implicit representation struggles to generalize well to those unseen lighting conditions and seen lighting conditions but lying in the margin of the distribution of the training dataset, as shown in the first row of Figure 10. To enhance the generalization, we take the merits of the idea that similar lighting conditions can be used to supervise each other. Specifically, instead of using I_{RGB} with the same lighting conditions as S in the $\mathcal{L}_{\text{Sim}}$, we introduce small disturbance to the lighting conditions of the portraits, i.e., I_{RGB}, as shown as portrait (a') in Figure 5. Namely, the loss $\mathcal{L}_{\text{Sim}}$ is adjusted to:

$$\mathcal{L}_{\text{Sim}} = \mathcal{L}_{\text{LPIPS}}(A(w), I_{\text{RGB}}(w, w_{sh+z_\sigma \sigma}) \oslash S(w, w_{sh})), \qquad (3)$$

where σ denotes the standard deviation of the lighting conditions in the training dataset and $z_\sigma \sim \mathcal{N}(0, c_\sigma^2)$. This loss refers to the calculated LPIPS metrics in Figure 5. Such disturbance expands the raw distribution, as shown in Figure 11(b) compared to (a), resulting in better generalization performance, as shown in the second row of Figure 10. Note that there could be alternative ways (e.g., applying another loss to directly promote the visual similarity between portraits of similar lighting conditions) of using similar lighting conditions to supervise each other and we choose to incorporate the disturbance into the previous loss, since it is efficient and effective.

In summary, the final regularization loss $\mathcal{L}_{\text{Reg}}$ is defined as the weighted summation of the above three losses. Namely,

$$\mathcal{L}_{\text{Reg}} = \mathcal{L}_{\text{Cross}} + \lambda \mathcal{L}_{\text{Sim}}, \qquad (4)$$

where we empirically set $\lambda = 1$. Besides the main GAN loss function, this regularization loss is implemented as lazy regularization following StyleGAN2 [Karras et al. 2020b] to reduce the computational cost and memory usage.

3.4 3D-aware Portrait Relighting

Our model can manipulate the lighting in a disentangled manner such that the geometry and appearance are well-preserved for each generated sample. It can be further extended to perform 3D-aware portrait relighting on single-view real-face images. To achieve this, the general idea is to project a real face image into our latent space and manipulate the latent codes for lighting to control the lighting condition. The official EG3D utilizes **pivotal tuning inversion (PTI)** [Roich et al. 2022] for projection, which can be roughly summarized as two steps: (1) Find an appropriate latent code whose generated sample is similar to the input real image but not necessarily accurate; (2) Tune the weights of the generator to fit the details of the real image. Our projection generally follows this idea but makes some adjustments at each step for fitting our purpose (i.e., relighting) and pipeline.

3.4.1 Finding Appropriate Latent Codes. Aiming at speeding up and starting a latent code whose generated sample is close to the input real image, we choose the encoder proposed in the encoder-for-editing [Tov et al. 2021] as our backbone. Specifically, we train an encoder E to predict the geometry and appearance latent codes for the real image I_{Real}, namely, $E(I_{\text{Real}}) = w \in \mathcal{W}$. Note that the encoder proposed in Tov et al. [2021] originally predicts latent codes in $\mathcal{W}+$ space, but we modify its architecture to predict latent codes in $\mathcal{W}$ space, following the design of PTI. The camera parameters are extracted via Deng et al. [2019] and assumed to be known and used for the corresponding volume rendering.

However, decomposing a real image into our pseudo-albedo and shading components is an ambiguous task. We prefer to model the appearance information as much as possible in the pseudo-albedo component, and thus conduct the loss directly on the pseudo-albedo component. This loss is formally defined as:

$$\mathcal{L}_{\text{Sim}} = \lambda_{\text{LPIPS}} \cdot \mathcal{L}_{\text{LPIPS}}(A^{+}(w), I_{\text{Real}}) + \lambda_{\text{ID}} \cdot \mathcal{L}_{\text{ID}}(A^{+}(w), I_{\text{Real}}),$$

where $\mathcal{L}_{\text{ID}}(\cdot,\cdot) = 1- < f_{\text{id}}(\cdot), f_{\text{id}}(\cdot) >$ and $f_{\text{id}}(\cdot)$ is the deep identity feature from a face recognition network [Deng et al. 2022], and $< \cdot,\cdot >$ denotes cosine similarity.

Besides, since the pseudo-albedo component is directly required to be similar to the real image, which is not deprived of shading, the encoder is prone to predict the sample whose pseudo-albedo component contains some residual shading, as analyzed in Section 4.5. To alleviate this issue, we take the merits of the idea introduced in Zhang et al. [2020] that human faces tend to be bilaterally symmetric and the asymmetry is mainly caused by the facial shadows, which should not be present on the pseudo-albedo A^{+}. Thus, we require the pseudo-albedo A^{+} to be perceptually similar to the flipped pseudo-albedo $\overline{A^{+}}$. We utilize the same practice to flip the face as in Zhang et al. [2020], leading to the following loss:

$$\mathcal{L}_{\text{Flip}} = \mathcal{L}_{\text{LPIPS}}(A^{+}(w), \overline{A^{+}}(w)).$$

Despite the flipping being specific to the face region, we also find it helpful for the encoder to predict latent codes whose pseudo-albedo components contain less residual shadows on the region out of the face (e.g., neck), as shown in Figure 8, since shadows on the face are usually correlated with shadows on other regions.

In summary, the final objective function is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{Sim}} + \lambda_{\text{Flip}} \cdot \mathcal{L}_{\text{Flip}}.$$

We set the $\lambda_{\text{LPIPS}} = 0.8, \lambda_{\text{ID}} = 0.5, \lambda_{\text{Flip}} = 0.8$. The optimization of this objective function leads to the optimization of the encoder E only. Note that the generator is fixed in this procedure.

To reconstruct the real image faithfully, it is necessary to estimate the lighting condition. We use the same practice to fetch the lighting condition sh for the real image as in labelling the images of the training set. To fit the estimated lighting condition with the predicted pseudo-albedo component by E, we perform optimization only on sh for 100 steps. The optimization loss is formally defined as:

$$\mathcal{L}_{\text{SH}} = \mathcal{L}_{\text{LPIPS}}(I^{+}_{\text{RGB}}(w, w_{\text{sh}}), I_{\text{Real}}).$$

Thus, the execution of this procedure leads to the estimated w and w_{sh} for the input real image I_{Real}.

3.4.2 Fine-tuning the Weights of the Generator. Given the predicted latent codes w and w_{sh}, the reconstructed face image I^{+}_{RGB} is usually not accurate enough compared to the real image I_{Real}, thus motivating us to fine-tune the weights of the generator G. Our generator can be viewed as generation with two steps: the first is to generate a tri-plane for geometry and appearance, and the second is to generate a shading tri-plane for lighting. Considering the missing or inaccurate details both in the geometry and appearance and the lighting, these two steps are all required to be fine-tuned, leading to another highly ambiguous task. The appearance details, especially for those with darkened colors (e.g., beards, black hairs) are prone to overfit in the shading. Besides, the shading details (e.g., shadows) are also possible to be modelled in the appearance.

As in the original PTI process, we use the L1 metrics $\mathcal{L}_{\text{L1}}$ and LPIPS $\mathcal{L}_{\text{LPIPS}}$ metrics to measure and optimize on the distance between the reconstructed face image and the real one. Besides these loss functions, we adopt several regularization terms to constrain the reconstructed face image to not only be accurate but also possess reasonable decomposed components (i.e., the pseudo-albedo A^{+}).

First, we apply the LPIPS loss directly on the pseudo-albedo A^{+} and the real face image I_{Real} to make the pseudo-albedo perceptually similar to the real one. However, this loss function is prone to leak the shading information into the pseudo-albedo. To alleviate this issue, we again take the merits of the flipping idea introduced in Zhang et al. [2020], requiring the pseudo-albedo A^{+} to be perceptually similar to the flipped pseudo-albedo $\overline{A^{+}}$ as $\mathcal{L}_{\text{Flip}}$. Formally, these losses are defined as:

$$\mathcal{L}_{\text{Sim}} = \mathcal{L}_{\text{LPIPS}}(A^{+}, I_{\text{Real}}), \mathcal{L}_{\text{Flip}} = \mathcal{L}_{\text{LPIPS}}(A^{+}, \overline{A^{+}}).$$

To refine the important face details (e.g., eyes), we further apply the facial feature loss to precisely optimize important face components as in He et al. [2021]. We use an off-the-shelf facial image segmentation module [Yu et al. 2018] to parse the real image I_{Real} into its semantic mask M and extract those important components. The refinement is formally defined as:

$$\mathcal{L}_{\text{Refine}} = \sum_{c \in C} \lambda_c \mathcal{L}_{LPIPS}(I_{\text{Real}} \odot M_c, I^{+}_{\text{RGB}} \odot M_c),$$

where $C = \{\text{eye}\}$ and M_c denote the mask for the corresponding facial component.

The final objective is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{L1}}(I^{+}_{\text{RGB}}, I_{\text{Real}}) + \lambda_1 \mathcal{L}_{\text{LPIPS}}(I^{+}_{\text{RGB}}, I_{\text{Real}}) + \lambda_2 \mathcal{L}_{\text{Sim}} + \lambda_3 \mathcal{L}_{\text{Flip}} + \lambda_4 \mathcal{L}_{\text{Refine}}.$$

We set $\lambda_1 = 1, \lambda_2 = 0.5, \lambda_3 = 0.25, \lambda_4 = 0.1$. The optimization of this objective leads to fine-tuning the weights of the whole generator, excluding the super-resolution module. Note that the latent codes w and w_{sh} are fixed in this procedure. As shown in Figure 7, our method achieves reasonable 3D-aware portrait relighting results.

4 EXPERIMENTS

In this section, we provide implementation details, quantitative evaluation, qualitative evaluation, and ablation study. We explore the generalization of our pipeline to unseen lighting conditions, the effectiveness of distilling the shading information from the pseudo-albedo components, and the degree of leakage of appearance into the shading component. We also provide additional results about the interpolation of latent codes, lighting control by an environment map, and user interface for interactive lighting control.

Table 1. Quantitative Comparison with ShadeGAN, VoLux-GAN, and EG3D Based on Geometry Consistency

Method	Relit Image Identity Similarity↑					Albedo Image Identity Similarity↑				
	−0.5 rad	−0.25 rad	0 rad	0.25 rad	0.5 rad	−0.5 rad	−0.25 rad	0 rad	0.25 rad	0.5 rad
ShadeGAN	0.4814	0.7513	-	0.7628	0.4997	0.4818	0.7582	-	0.7702	0.5091
VoLux-GAN	0.6064	0.7736	-	0.7997	0.5985	0.6389	0.7919	-	0.7863	0.6162
EG3D	**0.6728**	**0.8502**	-	**0.8487**	**0.6810**	-	-	-	-	-
Ours	0.6354	0.8253	-	0.8323	0.6464	**0.6868**	**0.8697**	-	**0.8723**	**0.6973**

Table 2. Quantitative Comparison with EG3D, IDE-3D, and 3DFaceShop on the FFHQ Dataset Based on Frechet Inception Distance

	FID↓	FID$_{\text{CLIP}}$↓
IDE-3D 512^2	4.9	3.3
EG3D 512^2	4.3	3.0
3DFaceShop 512^2	59.0	19.0
Ours 512^2	**4.0**	**2.8**

4.0.1 Implementation Details. We use DPR [Zhou et al. 2019] as in StyleFlow [Abdal et al. 2021] to predict the SH for real images. The shading decoder is implemented as a lightweight MLP with one hidden layer of 64 units. We fine-tune on the official checkpoint trained on the FFHQ [Karras et al. 2021b] of EG3D. The optimizer setup [Kingma and Ba 2015] is the same as EG3D, and we train at the resolution of 64^2 for 5M images with four Tesla V100 GPUs. We again use FFHQ [Karras et al. 2021b] as in EG3D for our training dataset. Our method is implemented with both PyTorch [Paszke et al. 2019] and Jittor [Hu et al. 2020]. More details can be found in the supplementary material.

4.1 Quantitative Evaluation

Our method achieves nearly real-time rendering performance at at 512^2 resolution, since the rendering is achieved based on pure inference. Specifically, on a single Tesla V100 GPU, the frame-per-second is 23.4 without tri-plane caching and 41.8 with tri-plane caching. This allows interactive changes of camera parameters and lighting conditions, as shown in the accompanying video. As to the projecting of a real portrait into the latent space of our generator for relighting, it takes about 3~4 minutes on a single Tesla V100 GPU, and this only needs to be done once for a specific sample.

To evaluate the generation ability of our method with quantitative metrics, we adopt the **Frechet Inception Distance (FID)** [Heusel et al. 2017]. We evaluate the FID on both InceptionNet-v3 [Szegedy et al. 2016], denoted as FID, and CLIP [Radford et al. 2021], denoted as FID$_{\text{CLIP}}$, as suggested by Kynkäänniemi et al. [2022]. We compare our method with our backbone EG3D [Chan et al. 2022], IDE-3D [Sun et al. 2022a], and 3DFaceShop [Tang et al. 2022] as state-of-the-art 3D-aware generative models. Specifically, we sample 50k images as in Karras et al. [2020b]. From Table 2, it is clear that our method has the best generation quality and diversity, showing that our design does not harm the generation ability of our backbone and even improves it.

Besides, we also adopt the geometry consistency proposed in VoLux-GAN [Tan et al. 2022] to measure the 3D consistency of the results by our method. We compare it with ShadeGAN [Pan et al. 2021], VoLux-GAN, and our backbone EG3D as alternative 3D-aware relightable and generative models (except for our backbone). From Table 1, it is clear that our method has the best geometry consistency in terms of both relit images and albedo images compared to the alternative 3D-aware lighting control methods, i.e., ShadeGAN and VoLux-GAN . Our backbone, EG3D, has better geometry consistency in terms of the images with shading than our method. It is possibly because the face recognition network we used is sensitive to the lighting condition. Since we explicitly pass the lighting conditions for sampling in our method, the generated samples cover some marginal lighting conditions, e.g., dark lights. However, the generated samples of our backbone do not necessarily contain these lighting conditions. This speculation also explains why the geometry consistency of our method in terms of the albedo images is better than that of our backbone, EG3D, in terms of the images with shading.

Table 3. Quantitative Comparison with StyleFlow, DisCoFaceGAN, and GAN-Control as 2D Relightable Methods, ShadeGAN, EG3D+Deep Portrait Relighting (EG3D+DPR), EG3D+StyleFlow, and 3DFaceShop as 3D-aware Relightable Methods Based on the Lighting Error and Lighting Stability

Method	Lighting Error↓	Lighting Stability↓
StyleFlow	0.7523	0.1530
DisCoFaceGAN	**0.5860**	0.1335
GAN-Control	0.6647	0.1485
ShadeGAN	1.0714	0.2149
3DFaceShop	0.5950	**0.1208**
EG3D+DPR	0.7424	0.1594
EG3D+StyleFlow	0.9935	0.2191
Ours	0.6377	0.1455

We highlight the best score as boldface, underline the second best, and double-underline the third best.

However, it is notoriously tricky to evaluate the decomposition as the albedo and shading and the relighting performance due to the difficulty of obtaining suitable ground truth. Considering this challenge, we decide to adopt two metrics to measure the relighting performance directly based on an off-the-shelf lighting estimator [Feng et al. 2021; Li et al. 2017], which is not used in training for any involved methods and is different from the one [Zhou et al. 2019] we use for labeling the dataset. It predicts the SH as lighting conditions.

Our proposed metrics are based on a lighting transfer task. Specifically, in this task, the model is required to generate images with the same lighting condition as a given real image. After that, the off-the-shelf lighting estimator estimates the lighting conditions of both real images and generated images. The benefit of this task is that it measures the lighting performance directly and does not require any ground truth, thus excluding the influence of how ground truth is obtained (e.g., from a professional capture

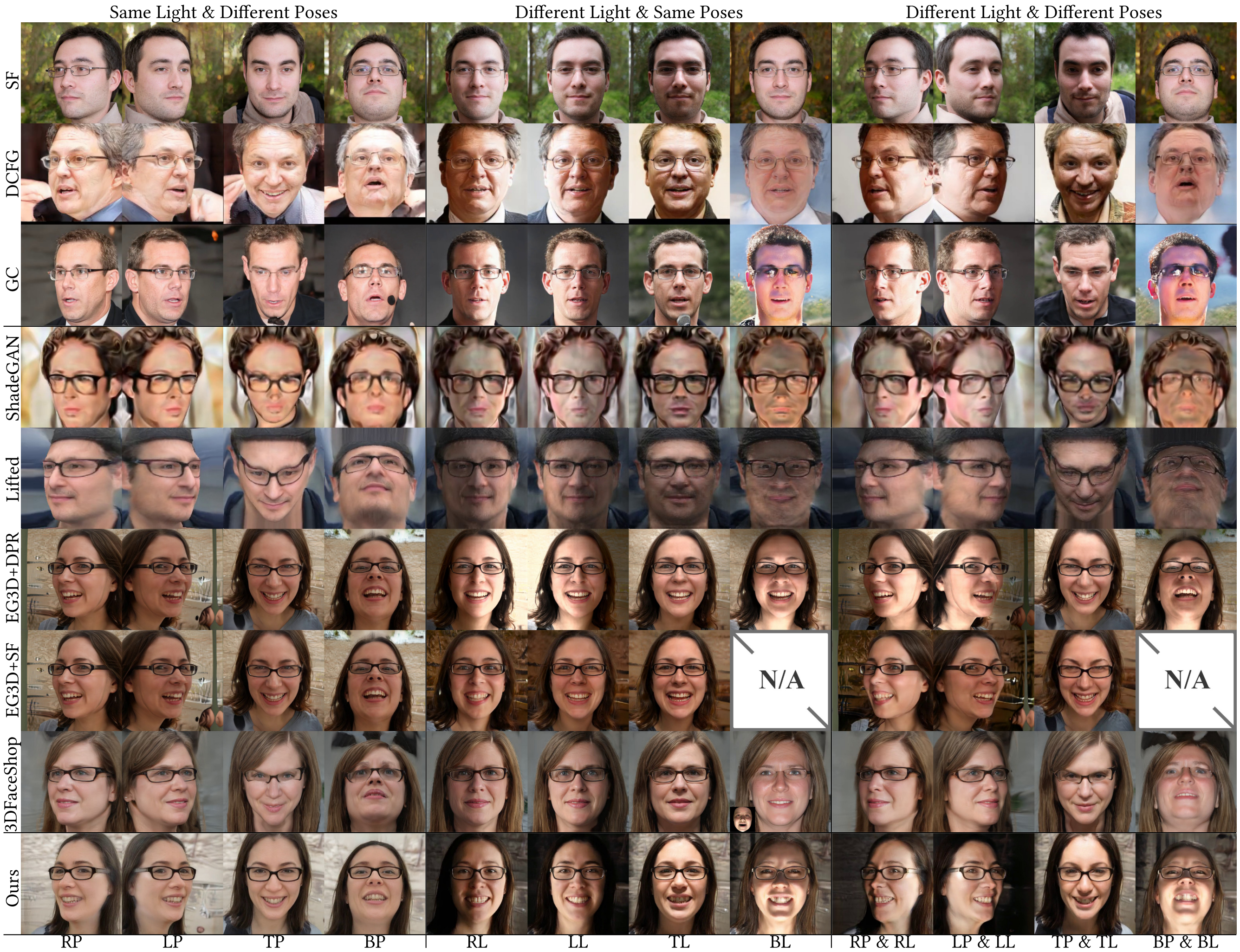

Fig. 6. Qualitative comparisons with StyleFlow (SF), DisCoFaceGAN (DCFG), and GAN-Control (GC) as 2D methods and LiftedStyleGAN (Lifted), EG3D+Deep Portrait Relighting (EG3D+DPR), EG3D+Style Flow (EG3D+SF), and 3DFaceShop as 3D-aware alternative methods. The results are viewed from a right position (RP), left position (LP), top position (TP), and bottom position (BP) and rendered under right light (RL), left light (LL), top light (TL), and bottom light (BL). Note that EG3D+SF fails at the case of bottom light, and we put the input lighting condition rendered by Deep3DRecon [Deng et al. 2019] at the bottom-left corner in the case of bottom light for 3DFaceShop.

setup or off-the-shelf albedo estimator) and whether direct albedo supervision exists. Besides, it also excludes the manners of how different models describe the lighting (e.g., SH, **spherical Gaussian (SG)**). However, this task requires the lighting conditions to be pre-extracted as a condition to the model, thus excluding one baseline method (i.e., VoLux-GAN). Additionally, the lighting estimator we used does not take non-face regions into account, e.g., hair.

4.1.1 Lighting Error.

Based on the lighting transfer task, we sample 1,000 real images from the dataset and generate corresponding fake images with the same lighting condition for each method. We use the off-the-shelf lighting estimator to estimate the lighting conditions for each pair (i.e., a real image and a generated image) and measure the distance between them. After calculating for each pair, we take the average as the final metrics value.

4.1.2 Lighting Stability.

Based on the lighting transfer task, we sample 1,000 real images from the dataset and generate 100 corresponding fake images for each real image with the same lighting condition. For each lighting condition, we use the off-the-shelf lighting estimator to estimate the lighting conditions of every generated fake image and measure the standard deviation of the spherical harmonics coefficients. After calculating for each real image, we take the average as the final metrics value.

We compare our method with alternative methods, including StyleFlow [Abdal et al. 2021], DisCoFaceGAN [Deng et al. 2020], and GAN-Control [Shoshan et al. 2021] as 2D relightable and generative models, and ShadeGAN, EG3D+Deep Portrait Relighting, EG3D+StyleFlow, and 3DFaceShop as 3D-aware relightable and generative models. From Table 3, it is clear that compared to the baselines and 2D methods, except for the DisCoFaceGAN and 3DFaceShop, our method has the lowest lighting error and the

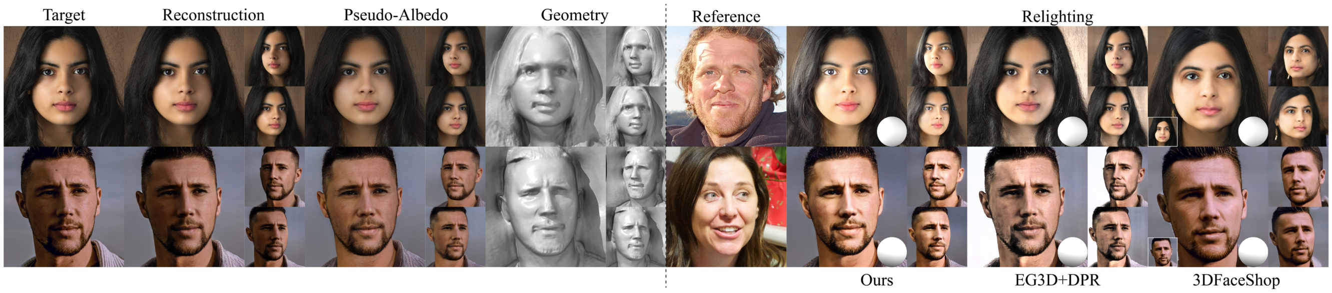

Fig. 7. Demonstration of projecting real portraits (left) and the comparison of 3D-aware portrait relighting with EG3D+DPR [Zhou et al. 2019] and 3DFaceShop [Tang et al. 2022] (right). On the right-hand side, the "Reference" denotes the reference lighting conditions on portraits. For 3DFaceShop, we put relit images with volume blending at the bottom-left corners of the relit images without using volume blending at the original poses. Original images courtesy of Flávio Augusto, Stefano Lubiana, Rene Alsleben, and Darius Dunlap.

best lighting stability. DisCoFaceGAN and 3DFaceShop use 3DMM as supervision, which enhances their accuracy and stability. However, DisCoFaceGAN suffers from the 3D inconsistency, as shown in the next qualitative comparison, and 3DFaceShop has worsened generation ability, as shown in Table 2. Both of them also disturb the geometric details (e.g., glasses and hair textures) when changing the lighting conditions, as shown in the succeeding qualitative comparison. Besides, as mentioned before, our lighting estimator, which does not consider non-facial regions, may also account for their higher accuracy and stability.

4.2 Qualitative Evaluation

To evaluate the ability of manipulating the lighting conditions in the generative model, we show qualitative comparisons with other relightable 3D-aware or 2D generative methods in Figure 6. Specifically, we choose four simple lighting conditions (i.e., top lights, bottom lights, left lights, and right lights). We compare our method with ShadeGAN [Pan et al. 2021], LiftedStyleGAN [Shi et al. 2021], EG3D+Deep Portrait Relighting [Zhou et al. 2019], EG3D+StyleFlow [Abdal et al. 2021], and 3DFaceShop [Tang et al. 2022] as alternative 3D-aware methods. Our method is compared to StyleFlow [Abdal et al. 2021], DisCoFaceGAN [Deng et al. 2020], and GAN-Control [Shoshan et al. 2021] as alternative 2D methods, which are also capable of changing the camera positions.

For each method, we show one subject randomly sampled from each latent space trained on the FFHQ dataset. In each row (from left to right), we show the reference images under one specific uniform lighting from four camera viewpoints, color images rendered under four different lighting directions from the frontal viewpoint, and color images rendered under four different lighting directions from four camera viewpoints. Specially, we do not adopt the volume blending for keeping the hair and background consistent when comparing with 3DFaceShop, since we want to emphasize on the lighting effects beyond the scope of face regions. For those methods (including ShadeGAN, LiftedStyleGAN, and Ours) that can generate albedo images, the reference images are replaced with albedo images. It is clear that the 2D implicit methods are prone to change the identities (e.g., glasses appear and disappear) when changing the camera parameters but achieve reasonable relighting results except for the bottom light. In contrast, the 3D-aware methods are good at preserving 3D consistency when changing the camera parameters. However, the generated samples of ShadeGAN and LiftedStyleGAN are of lower quality compared to those by ours and 2D methods. Their lighting effects are also unnatural, letting alone high-frequency lighting details such as the shadows cast by the glasses under the top light. The baseline "EG3D+DPR" has effective lighting effects but still cannot handle high-frequency lighting details and causes weird lights near the glasses in the case of "RP & RL." The baseline "EG3D+SF" is less effective at changing the lighting conditions and even fails at the bottom light. 3DFaceShop is able to handle the lighting effects beyond the scope of face regions and high-frequency lighting details. However, it comes at the cost of disturbing the geometry such as the shape of glasses and the mouth. Besides, all the methods except EG3D+DPR cannot handle the lighting condition of bottom lights, which are quite rare in the training dataset. On the contrary, our method not only generates high-quality 3D-consistent samples, thanks to our backbone, EG3D, but also enjoys realistic and detailed lighting effects due to our design. Our method is able to handle high-frequency lighting details (e.g., shadows cast by the glasses in the case of top light ("TL")) and the bottom light.

To evaluate the ability of portrait relighting, we show qualitative comparisons between our method and other representative methods that claim the ability of relighting real images, namely, DPR [Zhou et al. 2019] and 3DFaceShop [Tang et al. 2022]. To lift the 2D method (i.e., DPR) into a 3D-aware method, we first use EG3D to generate original portraits under different poses and then apply these image-based methods. Specially, we do not use volume blending when evaluating 3DFaceShop for the previously stated reason. However, we show relit images with volume blending at the bottom-left corner of the relit images without volume blending at the original pose for reference. It is clear that from Figure 7 (right side), our method can handle the lit area on the hair from top-left lights in the first row and shading on the face in the second row. In comparison, DPR fails to handle lights on the hair in the first row and retains some residual shading from the input target face in the second row. 3DFaceShop disturbs the geometry such as the hair in the first row and the texture on the face in the second row while changing the lighting conditions. It fails to handle the lit area on the hair in the first row and has lower reconstruction quality especially at other poses.

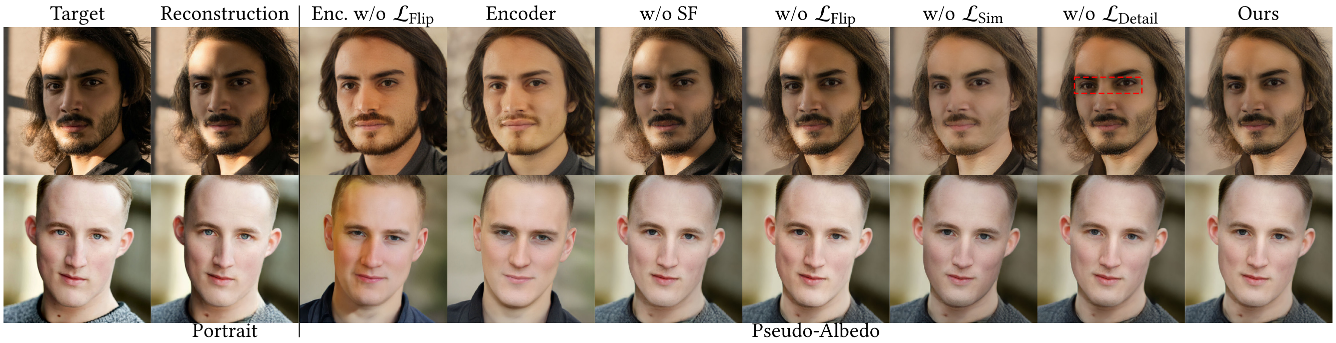
Target
Reconstruction
Enc. w/o $\mathcal{L}_{\text{Flip}}$
Encoder
w/o SF
w/o $\mathcal{L}_{\text{Flip}}$
w/o $\mathcal{L}_{\text{Sim}}$
w/o $\mathcal{L}_{\text{Detail}}$
Ours
Portrait
Pseudo-Albedo

Fig. 8. Ablation study for the design choices used in the projection of real portraits introduced in Section 3.4. "w/o SF" denotes the projection without fine-tuning the shading component. The red rectangle emphasizes on the inaccurate eye details, which lean left compared to the target image. Original images courtesy of Ba Tik and Niall Whitehead.

Table 4. Quantitative Evaluation with Ablation Models Based on the Lighting Error for Seen Lighting Conditions (Denoted as "LE") and Unseen Lighting Conditions (Denoted as "$\text{LE}_{\text{Unseen}}$") and the Lighting Stability (Denoted as "LS")

Method	LE↓	$\text{LE}_{\text{Unseen}}$↓	LS↓
Ours	**0.6377**	**0.6673**	**0.1455**
-$\mathcal{L}_{\text{Cross}}$	0.6947	0.6989	0.1621
-$\mathcal{L}_{\text{Sim}}$	0.6558	0.6779	0.1517
-Disturbance ($c_\sigma^2 = 0$)	0.6467	0.6818	0.1472

We highlight the best score as boldface, underline the second best, and double-underline the third best.

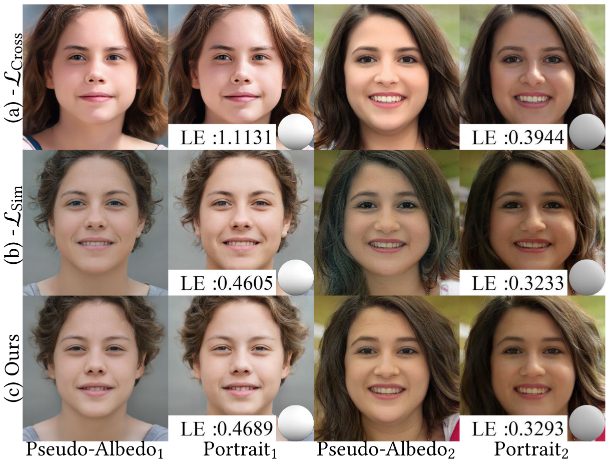

Fig. 9. Demonstration of ablation study for regularization introduced in the Section 3.3. We show two generated samples with random lighting conditions from the training dataset, and the latent codes w are kept the same for each method. "LE" denotes the lighting error (the lower, the better) introduced in Section 4.1.1.

4.3 Ablation Study

To verify the necessity of our design choices, we perform an ablation study on the regularization introduced in Section 3.3 and the loss functions when conducting 3D-aware portrait relighting in Section 3.4.

From Table 4, the ablation model without $\mathcal{L}_{\text{Cross}}$ has the highest lighting error for seen and unseen lighting conditions and the worst lighting stability. The ablation model without $\mathcal{L}_{\text{Sim}}$ and without disturbance have worsened lighting accuracy and stability. In Figure 9, we sample two latent codes and render them under the same lighting condition for each method. In the first row, there remains obvious shading in its pseudo-albedo components generated by the model without $\mathcal{L}_{\text{Cross}}$ in this case, impeding the lighting accuracy. Note that the lighting error of the ablation model without $\mathcal{L}_{\text{Sim}}$ is a bit lower than that of ours for these two cases, but the lighting estimator is not deprived of error and the visual differences are subtle. Besides, it is obvious that in the second row, the color on the hair of the Pseudo-Albedo$_2$ generated by the ablation model without $\mathcal{L}_{\text{Sim}}$ drops, i.e., dark-yellow hairs turn to lightened blue, compared to the corresponding portraits. From Figure 10, the ablation model without disturbance has noticeable black zones in the portraits under unseen lighting conditions or marginal lighting conditions in the seen distribution.

From Figure 8, it is clear that for the encoder, without $\mathcal{L}_{\text{Flip}}$ (denoted as "Enc. w/o $\mathcal{L}_{\text{Flip}}$"), the encoded pseudo-albedo components could retain some residual shading especially on the neck in the second row. Furthermore, the directly encoded pseudo-albedo components (denoted as "Encoder") are not accurate to the target images. Without fine-tuning the generation of shading tri-planes (denoted as "w/o SF"), there remain specular lights on the forehead in the first row. Without $\mathcal{L}_{\text{Flip}}$ (denoted as "w/o $\mathcal{L}_{\text{Flip}}$"), the tuning introduces obvious lighting effects in the pseudo-albedo. Without $\mathcal{L}_{\text{Sim}}$ (denoted as "w/o $\mathcal{L}_{\text{Sim}}$"), the fine-tuned pseudo-albedo is less similar to the input target images compared to ours at the last column (from left to right) (e.g., vague beards on the right face in the first row). Without $\mathcal{L}_{\text{Detail}}$ (denoted as "w/o $\mathcal{L}_{\text{Detail}}$"), the gaze in the left eye of the pseudo-albedo in the first row deviates slightly to the left (emphasized in the red rectangle). In contrast, our complete optimization strategy (denoted as "Ours") achieves the best visual similarity to the target images with slight residual shading on the pseudo-albedo.

4.4 Generalization to Unseen Lighting Conditions

Compared to those methods (e.g., Pan et al. [2021, 2022]; Shi et al. [2021]), which explicitly model the lighting effects through lighting models, our method can be categorized as the implicit

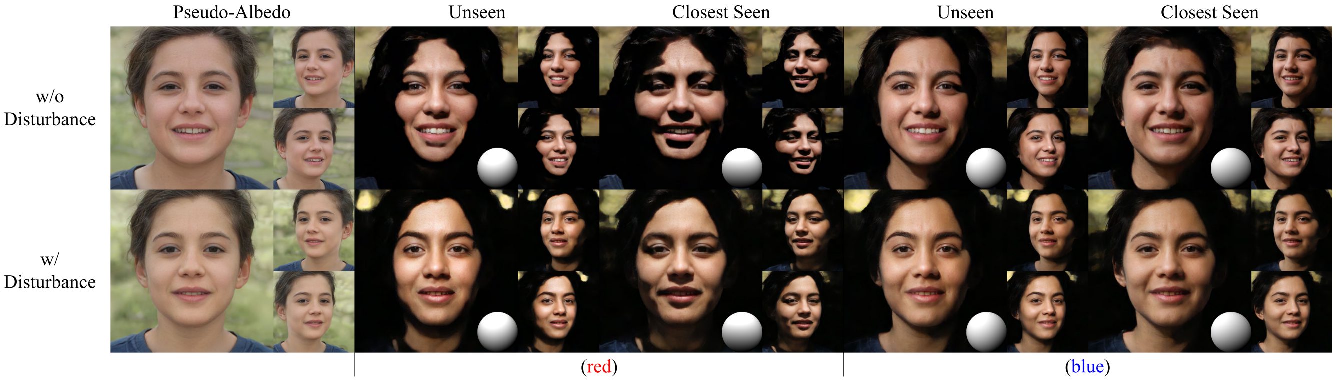

Fig. 10. Illustration of the ability to generalize to unseen lighting conditions w/ or w/o disturbance introduced in Section 3.3. (red) and (blue) correspond to the red and blue points in Figure 11. For each unseen lighting condition, we pick the closest seen lighting condition from the training dataset and demonstrate how these two models behave in the unseen lighting condition and its closest seen match, which is deviated severely from the central region of the distribution. Due to the rareness of such lighting conditions in the training dataset, the implicit representation tends to perform badly compared to those usual cases especially for creating noisy black backgrounds. It is clear that w/o disturbance, the portraits possess unnatural black zones on the faces, while w/ disturbance, the portraits are free of such a problem and look much more natural.

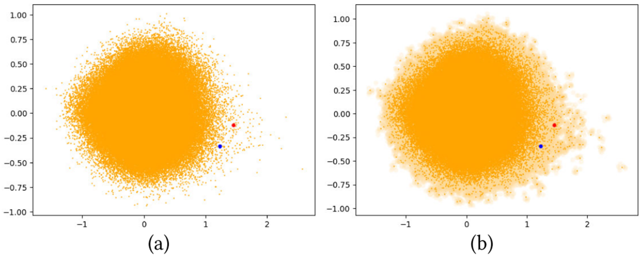

Fig. 11. Illustration of the distribution for lighting conditions in the training dataset. (a) denotes the raw distribution (each orange point stands for a sample) visualized through PCA. (b) denotes the visualized distribution (orange points) with disturbed data (light orange area). Red and blue points denote two unseen lighting conditions used in Figure 10. It can be seen that through disturbance, the model is trained on a more complete distribution (b), enhancing generalization ability.

representation learned from the dataset. Thus, it is necessary to evaluate how our model can generalize to the lighting conditions that are unseen in our training dataset.

We first explore and visualize the raw distribution of the lighting conditions in the dataset in Figure 11(a) through PCA [F.R.S. 1901] by keeping the two most important components. We then add disturbance to each sampled lighting condition and visualize the expanded distribution in Figure 11(b). It is clear that the coverage of the expanded distribution is more continuous and complete. Furthermore, we evaluate the models trained on the raw distribution and the expanded distribution both quantitatively and qualitatively.

To ensure meaningful lighting conditions and intuitive comparisons, we first compute the averaged lighting condition in our training dataset, FFHQ. We then measure the lighting conditions for samples of another widely used facial dataset, CelebA-HQ [Karras et al. 2021b; Liu et al. 2015], with the same off-the-shelf lighting estimator [Zhou et al. 2019] and pick those whose lighting conditions deviate the most from the average lighting conditions of FFHQ. For quantitative comparison, we compute the lighting error for the top 1, 000 lighting conditions from CelebA-HQ, which deviates the most from the training distribution in Table 4. It is clear that with the introduced disturbance, the lighting error for these top unseen lighting conditions is the lowest compared to that without the disturbance. Note that the sense of "unseen" lighting conditions does not always hold for other comparison methods.

For qualitative comparison, we select two unseen lighting conditions from CelebA-HQ, marked as red and blue points in Figure 11, and retrieve their closest seen lighting conditions from the training dataset. For each model (w/ or w/o disturbance), we demonstrate the behaviors of a generated sample on these lighting conditions. As shown in Figure 10, the model trained on the raw distribution ("w/o disturbance") fails to perform well with unnatural black zones on the faces under both the unseen lighting conditions and their closest seen matches, since these matches unavoidably lie in the margin of the training distribution. In contrast, the model trained on the expanded distribution ("w/ disturbance") generates natural portraits under these lighting conditions.

4.5 Distillation of Shading Information

Our method is capable of gradually distilling the shading information from the pseudo-albedo components into the shading components. To demonstrate the effectiveness of the distillation, we show several examples before and after distillation in Figure 12. We keep the latent codes unchanged for each example. It can be seen that our distillation method can reduce the shading significantly (e.g., self-occlusion shadows, left and right lights, strong light, and specular light) in the pseudo-albedo components without dramatically changing the appearance.

Despite our efforts to distill the shading information from the pseudo-albedo components into the shading components, the performance achieved by our method largely depends on how well the sampling strategy of style-mixing covers different lighting conditions. Since we use random fine-level style-mixing to approximate the generation of diverse residual shadings, there is

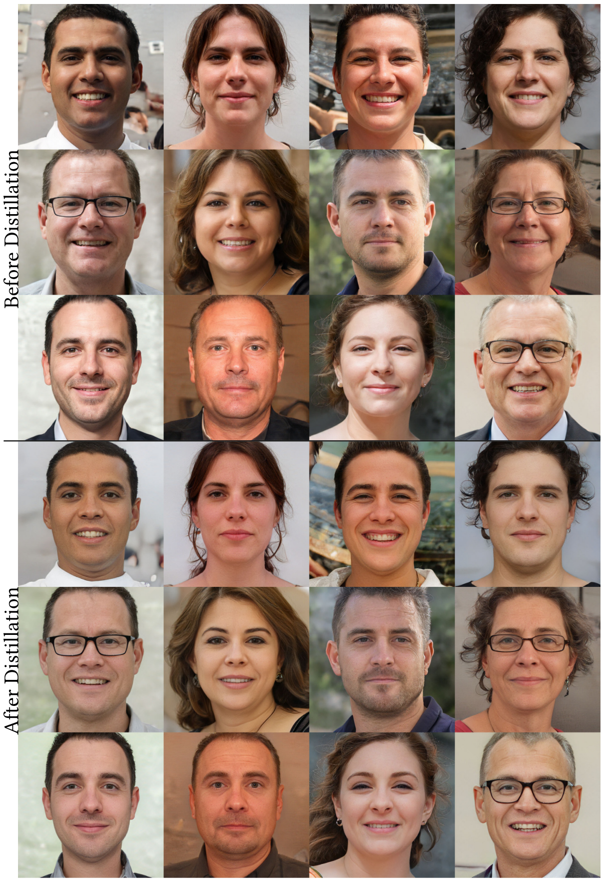

Fig. 12. Illustration of distillation. We keep the latent code unchanged and demonstrate the changes for the pseudo-albedo components before and after the distillation (adversarial training involved so the identity is deviated as well). It is clear that our framework and training strategy can alleviate various challenging lighting phenomena in the pseudo-albedo components (e.g., self-occlusion shadow, left light, right light, strong light, specular light).

no guarantee that the shading can be distilled completely from the pseudo-albedo components, though we still achieve reasonable results especially with the truncation trick, as shown in Figure 13(a). It can be seen from this figure that there is still some shading in the pseudo-albedo components of samples when the truncation ψ is high, but as the truncation ψ goes lower, the residual shading gradually disappears. Moreover, we quantify the effects of truncation on the lighting error proposed in Section 4.1.1, as shown in Figure 14. The lighting error (the lower, the better) improves as the truncation ψ changes from 1 to 0.5. Please see more detailed illustration in the supplementary material. Note that we do not use the truncation trick in the quantitative evaluation.

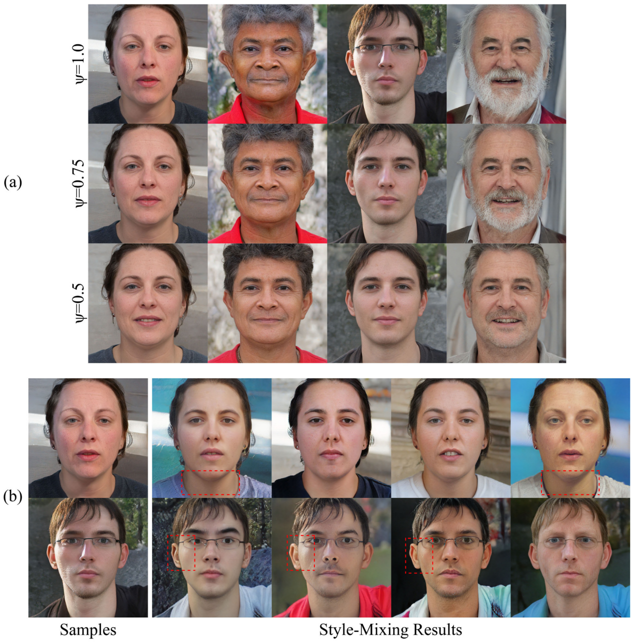

Fig. 13. Illustration of residual shading in the pseudo-albedo components. We show the pseudo-albedo components for four samples and the effect of truncation (denoted as ψ) in (a). For the first and third (from left to right) samples, which contain obvious residual shadings in the pseudo-albedo components, we further show four style-mixed pseudo-albedo images for each case in (b), and the residual shading on these style-mixed pseudo-albedo images is emphasized by red rectangles.

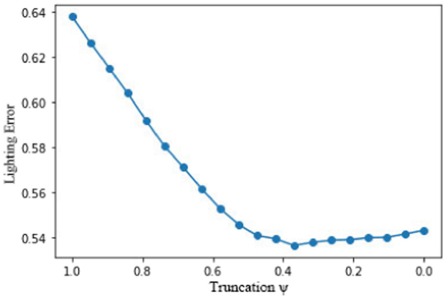

Fig. 14. Illustration of the effect on the lighting error for different truncation values of ψ. It is as expected that as the truncation ψ decreases, the generation quality improves and the pseudo-albedo components are less likely to contain the residual shading, causing the reduced lighting error.

We speculate that some lighting information is baked into the shallow layers of the original EG3D for some samples, making these samples contain residual shading effects. From Figure 13(b), the style-mixed pseudo-albedo components in the first row tend to contain the same right-to-left light effects (most visible for the shadow on the neck) and those in the second row tend to contain the same left lights (most visible for the specular lights on the left face). In these cases, the coverage of residual shadings through random fine-level style-mixing does not enjoy the averaged state as being deprived of shading. To alleviate this problem, a more intelligent sampling strategy for changing the lighting conditions while keeping the geometry unchanged of generated samples might be helpful.

4.6 Leaky Appearance Information

Because of the lack of ground-truth for the pseudo-albedo and shading components during training, it is unavoidable that some appearance information is still leaked into the shading component

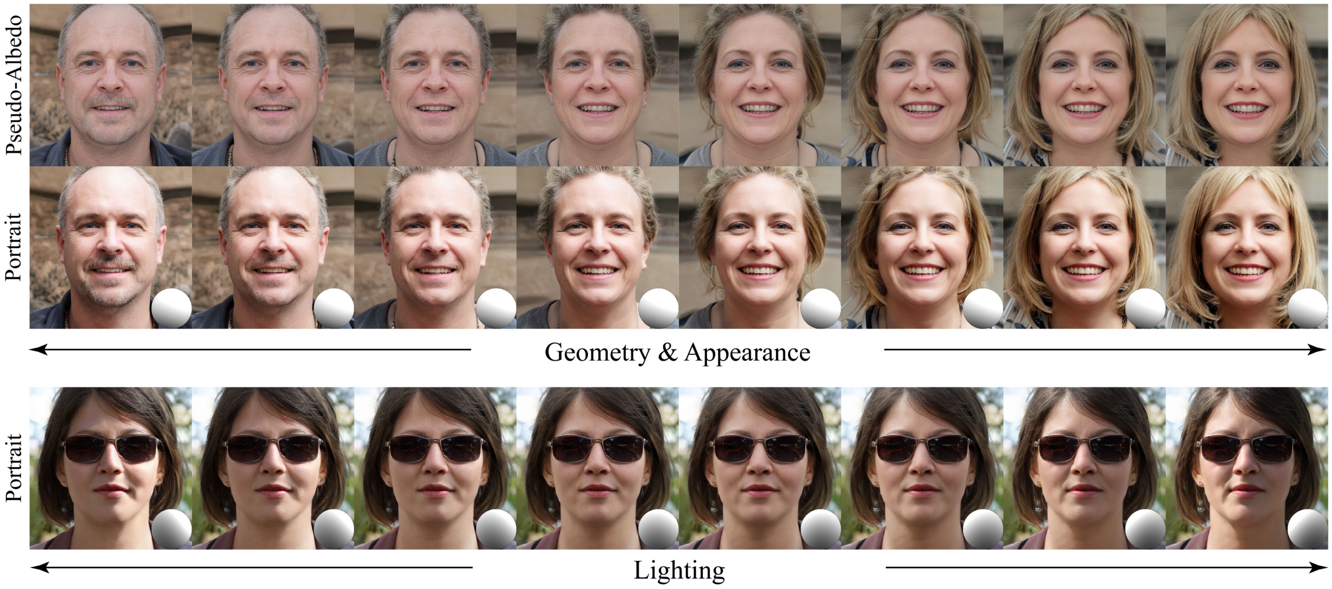

Fig. 15. Demonstration for interpolation of geometry and appearance latent codes in the first and second rows and interpolation of lighting latent codes in the third row. The camera is fixed at frontal for easier comparison.

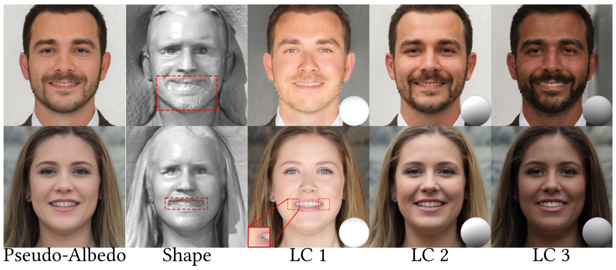

Fig. 16. Illustration of connection between geometry and lighting. “LC 1,” “LC 2,” and “LC 3” denote different lighting conditions. Red rectangles on the shape emphasize the flaws on the shape (e.g., incomplete beards or deep creases near the corners of mouths). Red rectangles on the portrait under “LC 1” in the second row emphasize disconnected lips and lights.

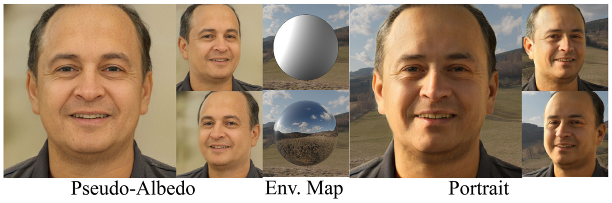

Fig. 17. Demonstration of lighting control by an environment map.

despite our regularization described in Section 3.3. Specifically, we find that the leaked appearance is in close relationship with the flaws of underlying genuine shape, as shown in Figure 16. In the first row, the high-frequency geometric details, i.e., beards, are not modelled well in the underlying genuine shape, resulting in growth or disappearance under certain lighting conditions (“LC 1” and “LC 3”). In these cases, the beards as the appearance are partly affected by the shading. Besides, in the second row, the generated portraits are affected by the creases near the corners of the mouth in the underlying genuine shape, resulting in weird lights and disconnected lip under a certain lighting condition (“LC 1”). In this case, the shading changes the appearance near the mouth. However, these problems are not visible under another lighting condition (“LC 2”). An even more obvious problem is with the background. Since the textures on the background are too diverse, the generator fails to capture its underlying shape. It is much more harder to distinguish, for example, a piece of gray region belonging to a curtain or results from the shadow, causing noisy backgrounds under different lighting conditions (e.g., the relatively clean background under “LC 1” and “LC 2” compared to the discontinuous texture on the background under the “LC 3” in the first row). We believe that a more accurate modelling of shape especially for beards will improve upon these limitations.

4.7 Additional Results

4.7.1 Interpolation of Latent Codes. In the first and second rows of Figure 15, we show the results of interpolating geometry and appearance latent codes while keeping the lighting conditions unchanged. As the hair gradually “grows” on the forehead, our method correspondingly changes the lighting effects in a smooth and natural way. In the third row of Figure 15, we show the results of interpolating lighting latent codes while keeping the geometry and appearance unchanged. Our method achieves smooth changes of the shadow cast by the sunglasses.

4.7.2 Lighting Control from Environment Map. In Figure 17, we show the result of lighting control by an environment map. We first extract SH from the environment map by Driscoll and Healy [1994]; Holmes and Featherstone [2002] and then plug these coefficients into our pipeline to generate corresponding portraits. The background is replaced by matting.

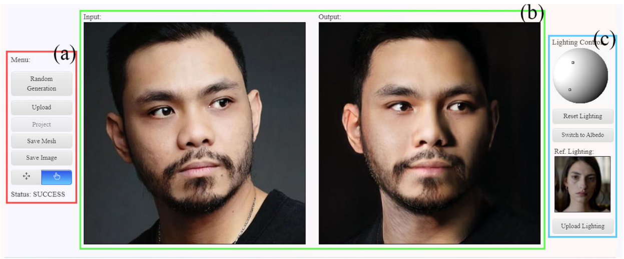

Fig. 18. A screenshot of our interface for users to manipulate the lighting conditions on a generated or real face in a 3D-aware manner. (a) and (c) are the utility control panel and lighting control panel, respectively. (b) includes an input real image (denoted as "Input") and projected synthesized image (denoted as "Output"), whose camera parameters and lighting conditions have been manipulated in this case. Original images courtesy of Andi Hamzah Lazuardy and cottonbro studio.

4.7.3 User Interface. We design a real-time face lighting-control user interface shown in Figure 18, allowing users to interactively manipulate the lighting conditions on a generated or real face in a 3D-aware manner. The menu (Figure 18(a)) consists of utility buttons such as "Random Generation" for users to test on generated samples, "Upload" for users to upload their own facial images, and so on, and necessary tools to manipulate the orientation of the face, i.e., pitch, yaw, roll. The lighting control panel (Figure 18(c)) offers two types of intuitive lighting controls. The lighting condition is first visualized as a sphere. Users can click on the sphere to add any number of light sources (visualized as little squares with black borders) and manipulate their locations by dragging or intensities by sliding the wheel. Another option is to upload a reference portrait for the system to extract its lighting condition. These two types are compatible, namely, the user can first upload a reference portrait and then manually add several light sources, which is exactly the case shown in Figure 18. Our system generates synthesized results in real-time, in terms of the camera manipulation and lighting condition manipulation.

5 CONCLUSION, LIMITATIONS, AND FUTURE WORK

Disentangled lighting control is tricky in the context of NeRF due to its innate ambiguity of decomposition as the albedo and the shading, which are necessary for faithful and disentangled relighting. A common solution may be to provide strong supervision from capture of real people, which is quite onerous for as many as tens of thousands of subjects, or by synthesis, which inevitably suffers from the gap with real images. In this work, leveraging generative prior, we explore how to decouple the lighting without direct external supervision and how to handle complex lighting phenomena (e.g., specular lights, shadows, occlusions) in an implicit but efficient and realistic way. Relighting is usually a data-hungry task, and we believe that by leveraging the generative prior of 3D GAN, our method is a step forward in the direction of alleviating such a harsh requirement.

One limitation of this work is that the performance of lighting control is in close relationship with the underlying shape, which is not error-free, causing problems such as inaccurate shading on beards or shadows from hats, as shown in Figure 16 and Figure 19(a). Besides, since we use an off-the-shelf lighting estimator to label the lighting condition of real portraits in the dataset, our method inevitably inherits its bias, as shown in Figure 19(b), where the darkened colors are correlated with very dim lights, causing it much harder to generate plausible pseudo-albedo and relit portraits. For reconstructing and relighting real portraits, the speed of projection is relatively slow, as shown in Section 4.1 due to the fine-tuning of the generator's weights. As future work, it would be interesting to explore how to speed up the projection and explore how to efficiently incorporate the explicit supervision (e.g., [Blanz and Vetter 1999; Li et al. 2017; Paysan et al. 2009]) used in previous methods into our implicit representation and regularization to enhance the performance of our model.

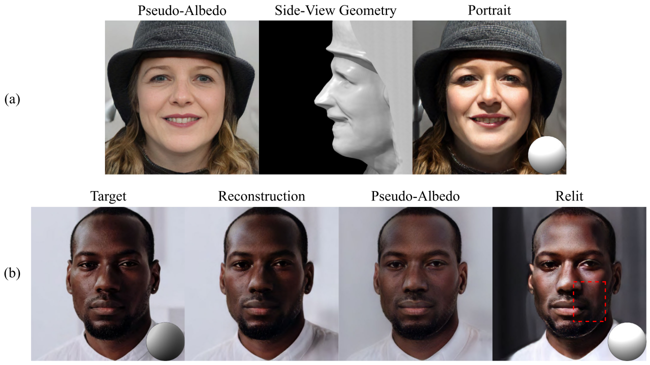

Fig. 19. Less successful cases for relighting a generated sample and a real portrait. (a) An example of how our model deals with other occlusions (e.g., hat) for a generated sample. Even though our model produces some shadow cast by the hat, the shadow only expands to the region above the eyebrows instead of around the nose, since the brim of the hat is too short, as shown in the side-view geometry. (b) For darkened colors, the lighting estimator predicts very dim lights. Note that even though the relighting condition in (b) is the same as that in (a), the portrait in (b) is less accurate compared to that in (a). For example, the face area in the red rectangle should be lighter. Original image courtesy of cottonbro studio.

ACKNOWLEDGMENTS

Thanks to Ms. Xiaohong Wang for her valuable comments and suggestions on the use of OneITLab. We also thank Prof. Lingjie Liu for her valuable suggestions and discussions of this research work.

REFERENCES

Rameen Abdal, Peihao Zhu, Niloy J. Mitra, and Peter Wonka. 2021. StyleFlow: Attribute-conditioned exploration of StyleGAN-generated images using conditional continuous normalizing flows. *ACM Trans. Graph.* 40, 3 (May 2021).

Volker Blanz and Thomas Vetter. 1999. A morphable model for the synthesis of 3D faces. In *26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'99)*. ACM Press/Addison-Wesley Publishing Co., 187–194.

Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P. A. Lensch. 2021. NeRD: Neural reflectance decomposition from image collections. In *IEEE/CVF International Conference on Computer Vision*. 12664–12674.

Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini de Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. 2022. Efficient geometry-aware 3D generative adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16102–16112.

Eric R. Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. 2021. pi-GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5795–5805.

Jiankang Deng, Jia Guo, Jing Yang, Niannan Xue, Irene Kotsia, and Stefanos Zafeiriou. 2022. ArcFace: Additive angular margin loss for deep face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 10 (2022), 5962–5979.

Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. 2020. Disentangled and controllable face image generation via 3D imitative-contrastive learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 5153–5162.

Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. 2022. GRAM: Generative radiance manifolds for 3D-aware image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 10663–10673.

Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. 2019. Accurate 3D face reconstruction with weakly-supervised learning: From single image to image set. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.*

J. R. Driscoll and D. M. Healy. 1994. Computing fourier transforms and convolutions on the 2-sphere. *Adv. Appl. Math.* 15, 2 (1994), 202–250.

Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. 2021. Learning an animatable detailed 3D face model from in-the-wild images. *ACM Trans. Graph.* 40, 4 (July 2021).

Karl Pearson F. R. S. 1901. LIII. On lines and planes of closest fit to systems of points in space. *Lond., Edinb., Dubl. Philos. Mag. J. Sci.* 2, 11 (1901), 559–572.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (Eds.), Vol. 27. Curran Associates, Inc.

Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. 2022. StyleNeRF: A style-based 3D aware generator for high-resolution image synthesis. In *10th International Conference on Learning Representations (ICLR'22).* OpenReview.net.

Yingqing He, Yazhou Xing, Tianjia Zhang, and Qifeng Chen. 2021. Unsupervised portrait shadow removal via generative priors. In *29th ACM International Conference on Multimedia (MM'21).* Association for Computing Machinery, New York, NY, 236–244.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc.

S. Holmes and Will Featherstone. 2002. A unified approach to the Clenshaw summation and the recursive computation of very high degree and order normalised associated Legendre functions. *J. Geod.* 76 (05 2002), 279–299.

Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. 2022. HeadNeRF: A realtime NeRF-based parametric head model. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 20342–20352.

Andrew Hou, Michel Sarkis, Ning Bi, Yiying Tong, and Xiaoming Liu. 2022. Face relighting with geometrically consistent shadows. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 4207–4216.

Andrew Hou, Ze Zhang, Michel Sarkis, Ning Bi, Yiying Tong, and Xiaoming Liu. 2021. Towards high fidelity face relighting with realistic shadows. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 14714–14723.

Shi-Min Hu, Dun Liang, Guo-Ye Yang, Guo-Wei Yang, and Wen-Yang Zhou. 2020. Jittor: A novel deep learning framework with meta-operators and unified graph execution. *Sci. China Inf. Sci.* 63, 222103 (2020), 1–21.

Kaiwen Jiang, Shu-Yu Chen, Feng-Lin Liu, Hongbo Fu, and Lin Gao. 2022. NeRFFaceEditing: Disentangled face editing in neural radiance fields. In *SIGGRAPH Asia 2022 Conference Papers (SA'22).* Association for Computing Machinery, New York, NY.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. 2020a. Training generative adversarial networks with limited data. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 12104–12114.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2020b. Analyzing and improving the image quality of StyleGAN. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 8107–8116.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2021a. Alias-free generative adversarial networks. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 852–863.

Tero Karras, Samuli Laine, and Timo Aila. 2021b. A style-based generator architecture for generative adversarial networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 12 (2021), 4217–4228.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations ICLR.*

Jeong-gi Kwak, Yuanming Li, Dongsik Yoon, Donghyeon Kim, David Han, and Hanseok Ko. 2022. Injecting 3D perception of controllable NeRF-GAN into StyleGAN for editable portrait image synthesis. In *European Conference on Computer Vision.* 236–253.

Tuomas Kynkäänniemi, Tero Karras, Miika Aittala, Timo Aila, and Jaakko Lehtinen. 2022. The role of ImageNet classes in Fréchet inception distance. *arXiv preprint arXiv:2203.06026* (2022).

Gengyan Li, Abhimitra Meka, Franziska Mueller, Marcel C. Buehler, Otmar Hilliges, and Thabo Beeler. 2022. EyeNeRF: A hybrid representation for photorealistic synthesis, animation and relighting of human eyes. *ACM Trans. Graph.* 41, 4 (July 2022).

Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. 2017. Learning a model of facial shape and expression from 4D scans. *ACM Trans. Graph.* 36, 6 (Nov. 2017).

Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. 2021. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 6494–6504.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision.* 3730–3738.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. 2018. Which training methods for GANs do actually converge? In *International Conference on Machine Learning.* 3478–3487.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2021. NeRF: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (Dec. 2021), 99–106.

Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.* 41, 4 (July 2022).

Thomas Nestmeyer, Jean-François Lalonde, Iain Matthews, and Andreas Lehrmann. 2020. Learning physics-guided face relighting under directional light. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 5123–5132.

Michael Niemeyer and Andreas Geiger. 2021. GIRAFFE: Representing scenes as compositional generative neural feature fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 11448–11459.

Roy Or-El, Xuan Luo, Mengyi Shan, Eli Shechtman, Jeong Joon Park, and Ira Kemelmacher-Shlizerman. 2022. StyleSDF: High-resolution 3D-consistent image and geometry generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 13493–13503.

Xingang Pan, Ayush Tewari, Lingjie Liu, and Christian Theobalt. 2022. GAN2X: Non-Lambertian inverse rendering of image GANs. In *International Conference on 3D Vision.* 711–721.

Xingang Pan, Xudong Xu, Chen Change Loy, Christian Theobalt, and Bo Dai. 2021. A shading-guided generative implicit model for shape-accurate 3D-aware image synthesis. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 20002–20013.

Rohit Pandey, Sergio Orts Escolano, Chloe Legendre, Christian Häne, Sofien Bouaziz, Christoph Rhemann, Paul Debevec, and Sean Fanello. 2021. Total relighting: Learning to relight portraits for background replacement. *ACM Trans. Graph.* 40, 4 (July 2021).

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B. Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. 2021. HyperNeRF: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Trans. Graph.* 40, 6 (Dec. 2021).

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS.* 8024–8035.

Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. 2009. A 3D face model for pose and illumination invariant face recognition. In *6th IEEE International Conference on Advanced Video and Signal Based Surveillance.* 296–301.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning.* PMLR, 8748–8763.

Ravi Ramamoorthi and Pat Hanrahan. 2001. An efficient representation for irradiance environment maps. In *28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'01).* Association for Computing Machinery, New York, NY, 497–500.

Daniel Rebain, Mark Matthews, Kwang Moo Yi, Dmitry Lagun, and Andrea Tagliasacchi. 2022. LOLNeRF: Learn from one look. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 1548–1557.

Daniel Roich, Ron Mokady, Amit H. Bermano, and Daniel Cohen-Or. 2022. Pivotal tuning for latent-based editing of real images. *ACM Trans. Graph.* 42, 1 (Aug. 2022).

Viktor Rudnev, Mohamed Elgharib, William Smith, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. 2022. NeRF for outdoor scene relighting. In *European Conference on Computer Vision.* 615–631.

Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. 2020. GRAF: Generative radiance fields for 3D-aware image synthesis. In *Advances in Neural*

Information Processing Systems, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 20154–20166.

Katja Schwarz, Axel Sauer, Michael Niemeyer, Yiyi Liao, and Andreas Geiger. 2022. VoxGRAF: Fast 3D-aware image synthesis with sparse voxel grids. *arXiv preprint arXiv:2206.07695* (2022).

Soumyadip Sengupta, Angjoo Kanazawa, Carlos D. Castillo, and David W. Jacobs. 2018. SfSNet: Learning shape, reflectance and illuminance of faces "in the wild." In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6296–6305.

A. Shashua and T. Riklin-Raviv. 2001. The quotient image: Class-based re-rendering and recognition with varying illuminations. *IEEE Trans. Pattern Anal. Mach. Intell.* 23, 2 (2001), 129–139.

Yichun Shi, Divyansh Aggarwal, and Anil K. Jain. 2021. Lifting 2D StyleGAN for 3D-aware face generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6254–6262.

Yichun Shi, Xiao Yang, Yangyue Wan, and Xiaohui Shen. 2022. SemanticStyleGAN: Learning compositional generative priors for controllable image synthesis and editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11244–11254.

Alon Shoshan, Nadav Bhonker, Igor Kviatkovsky, and Gérard Medioni. 2021. GAN-control: Explicitly controllable GANs. In *IEEE/CVF International Conference on Computer Vision*. 14063–14073.

Peter-Pike Sloan, Jesse Hall, John Hart, and John Snyder. 2003. Clustered principal components for precomputed radiance transfer. *ACM Trans. Graph.* 22, 3 (July 2003), 382–391.

Pratul P. Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T. Barron. 2021. NeRV: Neural reflectance and visibility fields for relighting and view synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7491–7500.

Jingxiang Sun, Xuan Wang, Yichun Shi, Lizhen Wang, Jue Wang, and Yebin Liu. 2022a. IDE-3D: Interactive disentangled editing for high-resolution 3D-aware portrait synthesis. *ACM Trans. Graph.* 41, 6 (Nov. 2022).

Jingxiang Sun, Xuan Wang, Yong Zhang, Xiaoyu Li, Qi Zhang, Yebin Liu, and Jue Wang. 2022b. FENeRF: Face editing in neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7662–7672.

Tiancheng Sun, Jonathan T. Barron, Yun-Ta Tsai, Zexiang Xu, Xueming Yu, Graham Fyffe, Christoph Rhemann, Jay Busch, Paul Debevec, and Ravi Ramamoorthi. 2019. Single image portrait relighting. *ACM Trans. Graph.* 38, 4 (July 2019).

Tiancheng Sun, Kai-En Lin, Sai Bi, Zexiang Xu, and Ravi Ramamoorthi. 2021. NeLF: Neural light-transport field for portrait view synthesis and relighting. In *Eurographics Symposium on Rendering*. 155–166.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2818–2826.

Feitong Tan, Sean Fanello, Abhimitra Meka, Sergio Orts-Escolano, Danhang Tang, Rohit Pandey, Jonathan Taylor, Ping Tan, and Yinda Zhang. 2022. 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.

Junshu Tang, Bo Zhang, Binxin Yang, Ting Zhang, Dong Chen, Lizhuang Ma, and Fang Wen. 2022. Explicitly controllable 3D-aware portrait generation. *arXiv preprint arXiv:2209.05434* (2022).

Ayush Tewari, Mohamed Elgharib, Gaurav Bharaj, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhöfer, and Christian Theobalt. 2020. StyleRig: Rigging StyleGAN for 3D control over portrait images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6141–6150.

Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. 2021. Designing an encoder for StyleGAN image manipulation. *ACM Trans. Graph.* 40, 4 (July 2021).

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 27171–27183.

Sheng-Yu Wang, David Bau, and Jun-Yan Zhu. 2022a. Rewriting geometric rules of a GAN. *ACM Trans. Graph.* 41, 4 (July 2022).

Yifan Wang, Aleksander Holynski, Xiuming Zhang, and Xuaner Cecilia Zhang. 2022b. SunStage: Portrait reconstruction and relighting using the sun as a light stage. *arXiv preprint arXiv:2204.03648* (2022).

Jianfeng Xiang, Jiaolong Yang, Yu Deng, and Xin Tong. 2022. GRAM-HD: 3D-consistent image generation at high resolution with generative radiance manifolds. *arXiv preprint arXiv:2206.07255* (2022).

Yinghao Xu, Sida Peng, Ceyuan Yang, Yujun Shen, and Bolei Zhou. 2022. 3D-aware image synthesis via learning structural and textural representations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18409–18418.

Yu-Ying Yeh, Koki Nagano, Sameh Khamis, Jan Kautz, Ming-Yu Liu, and Ting-Chun Wang. 2022. Learning to relight portrait images via a virtual light stage and synthetic-to-real adaptation. *ACM Trans. Graph.* 41, 6 (Nov. 2022).

Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. 2018. BiSeNet: Bilateral segmentation network for real-time semantic segmentation. In *European Conference on Computer Vision*. 325–341.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 586–595.

Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul Debevec, William T. Freeman, and Jonathan T. Barron. 2021. NeRFactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Trans. Graph.* 40, 6 (Dec. 2021).

Xuaner (Cecilia) Zhang, Jonathan T. Barron, Yun-Ta Tsai, Rohit Pandey, Xiuming Zhang, Ren Ng, and David E. Jacobs. 2020. Portrait shadow manipulation. *ACM Trans. Graph.* 39, 4 (Aug. 2020).

Boming Zhao, Bangbang Yang, Zhenyang Li, Zuoyue Li, Guofeng Zhang, Jiashu Zhao, Dawei Yin, Zhaopeng Cui, and Hujun Bao. 2022b. Factorized and controllable neural re-rendering of outdoor scene for photo extrapolation. In *30th ACM International Conference on Multimedia (MM'22)*. Association for Computing Machinery, New York, NY, 1455–1464.

Xiaoming Zhao, Fangchang Ma, David Güera, Zhile Ren, Alexander G. Schwing, and Alex Colburn. 2022a. Generative multiplane images: Making a 2D GAN 3D-aware. In *European Conference on Computer Vision*. 18–35.

Hao Zhou, Sunil Hadap, Kalyan Sunkavalli, and David Jacobs. 2019. Deep single-image portrait relighting. In *IEEE/CVF International Conference on Computer Vision*. 7193–7201.

Peng Zhou, Lingxi Xie, Bingbing Ni, and Qi Tian. 2021. CIPS-3D: A 3D-aware generator of GANs based on conditionally-independent pixel synthesis. *arXiv preprint arXiv:2110.09788* (2021).

Received 8 November 2022; revised 8 February 2023; accepted 22 April 2023