HairNeRF: Geometry-Aware Image Synthesis for Hairstyle Transfer

Seunggyu Chang*
NAVER Cloud

Gihoon Kim†
KAIST

Hayeon Kim†
UNIST

Figure 1: Given portrait images of hairstyle and identity (*thumnails*) that differ in viewing angle, head shape, and size, we seamlessly transfer hairstyle from one image to another utilizing neural rendering. Our method preserves fine-level details of the hairstyle structural geometry and generates diverse range of pose variant transferred images without requiring additional transfer process.

Abstract

We propose a novel hairstyle transferred image synthesis method considering the underlying head geometry of two input images. In traditional GAN-based methods, transferring hairstyle from one image to the other often makes the synthesized result awkward due to differences in pose, shape, and size of heads. To resolve this, we utilize neural rendering by registering two input heads in the volumetric space to make a transferred hairstyle fit on the head of a target image. Because of the geometric nature of neural rendering, our method can render view varying images of synthesized results from a single transfer process without causing distortion from which extant hairstyle transfer methods built upon traditional GAN-based generators suffer. We verify that our method surpasses other baselines in view of preserving the identity and hairstyle of two input images when synthesizing a hairstyle transferred image rendered at any point of view.

*Corresponding author: seunggyu.chang@navercorp.com
†This work was done during internship at NAVER Cloud.

1. Introduction

Selecting an appropriate hairstyle can be difficult as it is hard to imagine how it will look on your face without trying it out for yourself. The reason can be attributed to the inherent variability of individual head shape, resulting in the frequent occurrence of disparate outcomes when attempting to emulate celebrity hairstyles. To address this issue, we propose a novel image synthesis method for hairstyle transfer tailored to fit an individual head shape. We named our method *HairNeRF* as it considers geometric alignment in the volumetric space of neural rendering [33]. Hairstyle transfer synthesizes an image where the hairstyle from one

portrait is transferred to the other and Generative adversarial networks (GANs) [14, 20, 22] have been extensively utilized to address this problem. Notably, following the emergence of StyleGAN [20, 21, 22], which enables the generation of highly realistic images, studies [37, 50, 19, 8] have been made to optimize latent vectors within the well-trained latent space of StyleGAN to generate an optimal synthesis outcome. These methods are successful at transferring hairstyles between images having similar poses, but struggle to generate convincing results for images having vastly different poses. To resolve this, studies [50, 24, 51] have proposed methods aligning pose of transferred hairstyle to a target portrait, simultaneously reducing the perceptual difference between the transferred result and the original hairstyle. However, they inherently encounter the same limitation of the pre-trained latent space of StyleGAN, where pose and style attributes are entangled as StyleGAN is not explicitly trained for hairstyle transfer. Consequently, the alignment process may distort the geometry of an original hairstyle, leading to undesirable alterations such as forehead stretching or shrinking.

We categorize entanglement of pose and style attributes into two folds: entanglement of style with viewpoint, and entanglement of style with head shape. To address the entanglement of style with viewpoint, we leverage the latent space of StyleNeRF [15], a neural radiance field (NeRF) [32, 33] based GAN model as an alternative for StyleGAN. StyleNeRF exhibits an innate pose-invariant latent space as the network implies the volumetric geometry of an image. We basically follow the latent optimization technique following the literature [50, 24, 51] utilizing StyleGAN. Leveraging the pose-invariant nature of the StyleNeRF's latent space, we extract a pose-invariant style vector through GAN inversion [42]. Thereafter, we recombine the volumetric features generated by the inverted style vectors of the hair and the target portraits. However, despite the pose-invariant nature of StyleNeRF's latent space, it still entails an entanglement problem between style and head shape. The optimization of a style vector to fit on the head of a target portrait, while retaining hairstyle of a hair portrait, can potentially result in deformations of the hairstyle. To address the entanglement of style with head shape, we propose a registration method that aligns the geometries of two heads within the volumetric space, while maintaining the inverted style vectors intact. Subsequently, we introduce a novel rendering method that recombines hairstyle features with facial features within the volumetric space using alignment-inducing deformable rays, which are derived from the registration outcome. In this manner, we resolve the distorted hairstyle problem from which StyleGAN-based method suffer, resulting in better alignment of the transferred hairstyle seamlessly fit on the target portrait. Taking the advantage of geometric nature of neural rendering, HairNeRF is capable of generating a diverse set of geometrically consistent novel view hairstyle transferred images from a single synthesis, by merely adjusting camera parameters. This is in contrast to existing StyleGAN-based hairstyle transfer methods which necessitate separate optimizations for each of novel views, which often yield inconsistent outcomes.

Experimental results on diverse datasets demonstrate the superiority of HairNeRF against other baselines at generating seamless hairstyle transfer outcomes, while effectively preserving the underlying geometry of the original hairstyle.

2. Related Work

2.1. Hairstyle Transfer

The proficiency exhibited by generative adversarial networks (GANs) [14, 20, 22] in generating photorealistic images has propelled a surge of research addressing image editing problem. A significant amount of research [3, 7, 19, 16, 12] has effectively addressed attribute editing or style transfer, which modify abstract-level features of an image. However, the task of hairstyle transfer still presents a challenging problem that necessitates the preservation of intricate structure and color details of a hairstyle from a source image. With the emergence of StyleGAN [20, 21, 22], which demonstrates exceptional quality in generating photorealistic facial images, efforts have been made to discover latent vectors capable of capturing transferred images of hairstyles [37, 50]. These methods require the projection of an image into the pretrained StyleGAN's latent space known as inversion, which influences the quality of transfer. Thus, the Barbershop [50] introduces an improved inversion method on the enlarged latent space named *FS*-space to capture finer details of input image. However, since these models lack awareness of the pose of given images, they suffer structural artifacts when the given two images largely differ in poses. Recent studies [24, 51] resolve this issue by aligning pose of the transferred hairstyle to the target image. However, they still suffer style deformation problem arising from the entangled latent space of StyleGAN for pose and style.

2.2. 3D-Aware Image Synthesis and Editing

The emergence of StyleGAN [20, 21, 22] has facilitated researches on image editing [16, 41, 2, 19, 29, 23, 1], either by navigating the latent style space or leveraging semantic maps. However, StyleGAN-based image editing methods have shown inconsistency among pose changes, which is attributed to the entanglement of style and pose within StyleGAN's latent space. The development of neural radiance fields (NeRF) [32, 33] has enabled the rendering of 3D-consistent images that accommodate varying viewpoints. Consequently, methods for editing a 3D scene

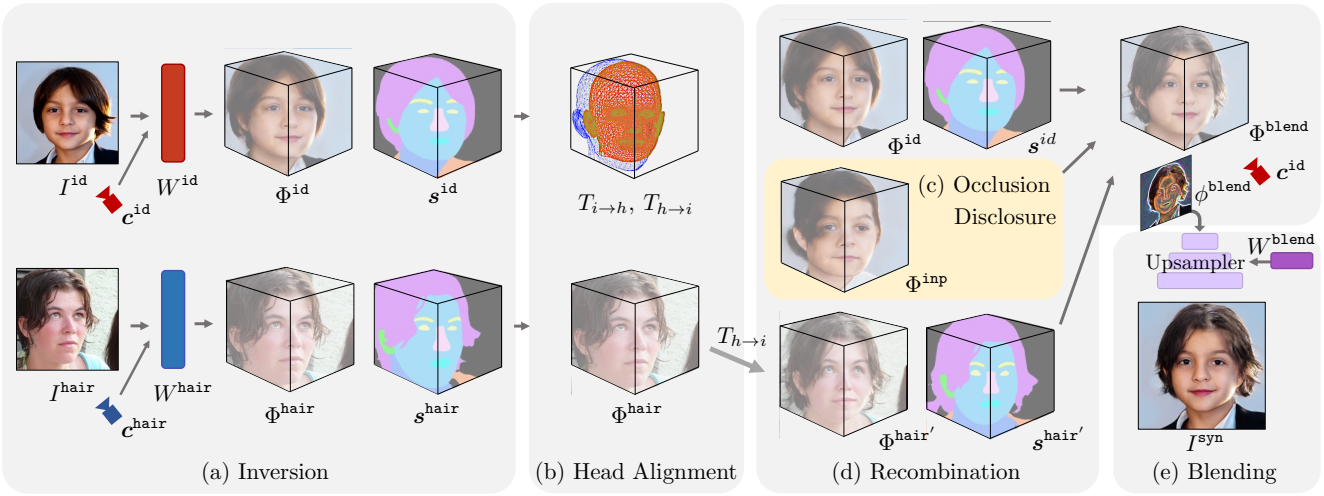


Figure 2: Overview of the hairstyle transfer pipeline of HairNeRF consisting of five modules: (a) inversion, (b) head alignment, (c) occlusion disclosure, (d) recombination, and (e) blending.

[44, 30, 27, 48] such as the addition and subtraction of NeRF models trained on individual objects have been explored. However, these approaches do not support modifications, such as altering the attributes or style of the object, due to the inherent nature of NeRF being trained to solely represent a single entity, thus lacking the necessary information for the desired edits. The advent of generative NeRFs [5, 38, 35, 15, 4, 10, 43] has enhanced the representational capabilities of NeRF models, enabling the expression of an expansive range of objects through the exploration of the latent space within a single model. The enhanced expressive capabilities of generative NeRFs enable the modification of attributes and style of an object [15, 4, 10]. Furthermore, within the generative NeRF modules, methods [30, 18, 40, 39, 6] have been proposed that disentangle the semantic structure and texture. However, these methods are not well-suited for local style modifications, as the latent vector representing the texture exhibits spatial entanglement, encompassing the entire regions of the image.

3. Method

3.1. Overview

We follow the basic synthesis pipeline from the Barbershop [50], a founding work for hairstyle transfer [24, 50]. We encourage readers to refer [50] for the details of the original pipeline. The overall synthesis pipeline of our method consists of four modules: inversion, head alignment, recombination, and blending, as illustrated in Figure 2. Given two images, `id` and `hair`, we transfer hairstyle from the `hair` image into the `id` image. In the inversion module, we first invert the two images to obtain latent vectors and the corresponding features in the volumetric space. Then, in the alignment module, we conduct registration on two geometric heads of the inverted images in the volumetric space. By aligning geometric heads, we can transfer hairstyle of the `hair` image into the `id` image to be more fit. Once the registration is done, we recombine the 3D features in the volumetric space according to semantics to render a 2D feature for synthesis using neural rendering in the recombination module. A simple method is to bring hair semantic features from the `hair` image, and other semantics from the `id` image. However, there occur missing regions which are occluded by the hair in the `id` image but not covered by the hair of the `hair` image. To fill in the missing regions, we inpaint 3D features by optimizing a latent vector, ensuring that the 2D features rendered from the 3D features generated by the optimized latent vector align with the target semantics across all viewpoints. Thereafter, we semantically integrate the 3D features from the `id`, `hair`, and the inpainting output to form a blended feature for the synthesis. The blend feature is rendered to a 2D feature, which is decoded by the upsampler blocks of StyleNeRF [15] conditioned on the blending style vector. In the final blending module, the blending style vector is optimized to make the decoded image preserve the same hairstyle as in the `hair` image while retaining face of the `id` image.

3.2. NeRF Inversion

Semantic Parsing NeRF Head. To semantically recombine features in the 3D volume, semantics of volumetric features are required. It is known that a semantic classifier trained to classify a 2D point feature obtained by (2) can aptly classify semantic of a 3D point feature as well [25]. Based on this observation, we add an additional seman-

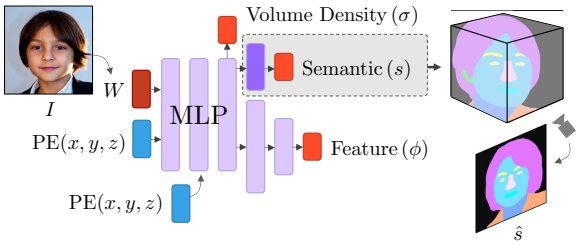


Figure 3: The additional semantic head is added at the end of a MLP block of NeRF module, which outputs a logit of semantics in the volumetric space. The additional semantic head is trained to minimize cross-entropy between the rendered logit, $\hat{s}$, and the ground-truth.

tic branch to the MLP block for neural rendering and train it using outputs of the off-the-shelf semantic segmentation model [46] for generated images as ground-truths as shown in Figure 3. After training, the trained semantic branch outputs a logit of semantics for a 3D point feature.

Semantics-Aware Inversion in 3D. We invert the `hair` and the `id` images into the latent vectors for the synthesis network of the StyleNeRF [15]. The process of obtaining the latent vector generating a desired image is called GAN inversion. We utilze the pivotal tuning inversion (PTI) [36] with a little modification for inversion as the PTI can reconstruct fine details of an input image from the inverted latent vector. As our model is built upon the StyleNeRF, the output image varies as a camera parameter changes. For this reason, the inversion process requires an input camera parameter or should estimate a camera parameter in our case. We choose to provide a fixed camera parameter of an input image estimated by the off-the-shelf head pose estimator [31] for the inversion process. Using the PTI with a fixed camera parameter, we optimize $W \in \mathcal{W}$ for each of the `id` and the `hair` images, and finetune the weights of the pretrained StyleNeRF to reconstruct the input image and the semantics estimated by the off-the-shelf semantic parsing networks [47, 45].

3.3. Head Alignment

Registration For registration, we render each of inverted images into n_c different viewpoints and extract 2D keypoints of the FLAME [28] head model using the off-the-shelf head pose estimator [31]. Then we estimate 3D keypoints of the FLAME model for each inverted image from the set of 2D keypoints estimated from the eight different camera views by solving a bundle-adjustment-like least square problem:

$$\underset{\boldsymbol{p}}{\text{minimize}} \sum_{i}^{n_c} \|P(\boldsymbol{c}_i, \boldsymbol{p}) - \boldsymbol{x}_i\|_2^2, \tag{1}$$

where $\boldsymbol{x}_i \in \mathbb{R}^2$ denotes the estimated 2D keypoints at i-th viewpoint, $\boldsymbol{c}_i$ denotes the i-th camera parameter, and $P(\boldsymbol{c}_i, \boldsymbol{p}_i)$ denotes the projection function projecting a 3D point $\boldsymbol{p}_i \in \mathbb{R}^3$ into 2D using $\boldsymbol{c}_i$. As we already know the camera parameters used for rendering n_c images, we exclude camera parameters from the optimization variables. We utilize the thin-plate-spline (TPS) transform [11] to find the optimal non-rigid transformation for the registration.

Neural Rendering via Ray Deformation. The two heads of the `id` and the `hair` do not fully overlap due to the differences in characteristics, shape and size of the heads geometry, along with estimation errors in head poses. We refer to this misalignment as *intrinsic* misalignment, as it arises from factors unrelated to external elements such as camera parameters. To alleviate the intrinsic misalignment, the registration process finds an optimal transformation between two point sets, by which one point set is transformed to align with the other. During the neural rendering process, point features are sampled along a ray within the volumetric space, and these features are then aggregated to a single point on the image plane. If the sampling points are collected along the ray, deformed by the transformation determined through registration, we can obtain registered point features. This process leads to a registered 2d image[1] after the aggregation.

Figure 4 illustrates the effective neural rendering process for alignment via ray deformation. Let $T_{i\to h}$ denote a transformation determined through registration, which aligns the point set of the `id` features to the point set of the `hair` features. As depicted in Figure 4a, sampling along rays that radiate straight from the `id`'s camera renders an `id`'s face with an innate head shape, colored in red, as in shown Figure 4c. Likewise, rendering from straight rays radiating through `hair` features $\Phi^{\texttt{hair}}$ using the same camera renders the `hair` image posed as `id`, with the overlaid innate head shape colored in blue, as depicted in Figure 4e. The two heads do not overlap due to differences in shape and size. However, if we render a `hair` image using the deformed ray transformed by $T_{i\to h}$, as depicted in Figure 4b, we obtain a registered `hair` image aligned to the head of `id`, as shown in Figure 4d. With a camera ray $\boldsymbol{r} = \boldsymbol{o} + t\boldsymbol{d}$, the registered neural rendering computes a pixel value of the ray as:

$$\begin{aligned} I_{\omega}^{\text{Registered}}(\boldsymbol{r}) &= \int_0^{\infty} p_{\omega}(t)\boldsymbol{c}_{\omega}(\tilde{\boldsymbol{r}}(t), \tilde{\boldsymbol{d}}) \left|\frac{d\tilde{\boldsymbol{r}}}{dt}\right| dt, \\ \text{where } p_{\omega}(t) &= \exp\left(-\int_0^t \sigma_{\omega}\left(\tilde{\boldsymbol{r}}(s)\right) \left|\frac{d\tilde{\boldsymbol{r}}}{ds}\right| ds\right) \cdot \sigma_{\omega}\left(\tilde{\boldsymbol{r}}(t)\right), \end{aligned} \tag{2}$$

with a registered ray $\tilde{\boldsymbol{r}} = T_{i\to h}(\boldsymbol{r})$, and the adjusted directional vector $\tilde{\boldsymbol{d}} = d\tilde{\boldsymbol{r}}/dt$. Equation (2) has the same form as the original neural rendering, but with the integration computed along the transformed ray. This adjustment

[1] or a registered 2d feature map in the StyleNeRF [15]

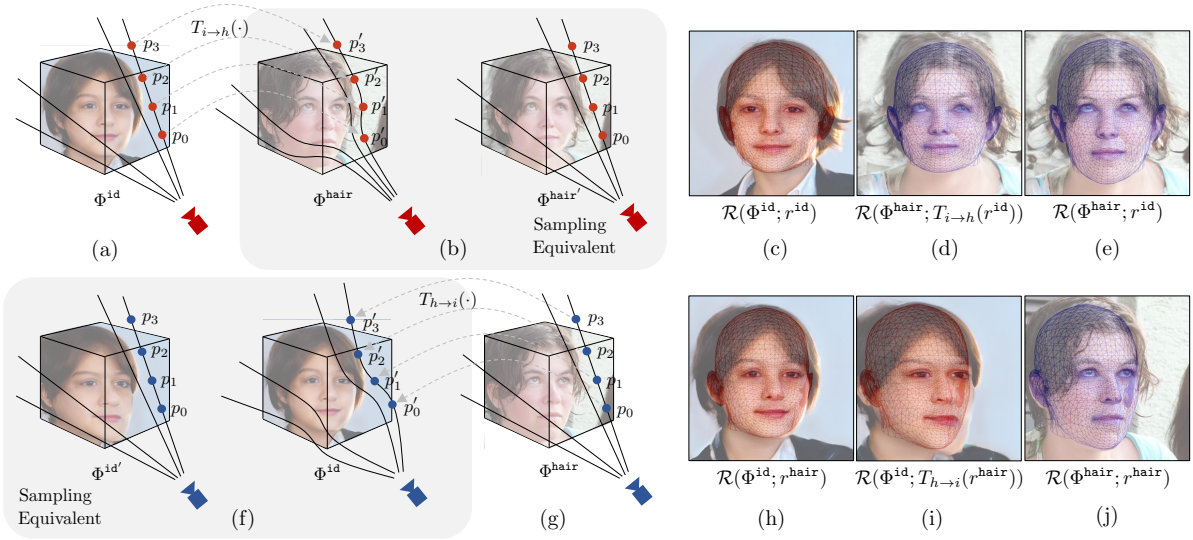


Figure 4: Illustration of alignment-aware neural rendering via ray deformation. Figure (c) – (e), and (h) – (j) show overlaid heads on rendering images. Rendering via transformed ray render a registered image as shown in (d) and (i), which is equivalent to rendering with the straight ray on transformed feature represented as $\Phi^{\mathtt{hair}'}$ and $\Phi^{\mathtt{id}'}$, respectively.

is achieved by substituting the directional vector $\boldsymbol{d}$ with $\tilde{\boldsymbol{d}}$ and including $|d\tilde{\boldsymbol{r}}/dt|$ to adjust the infinitesimal length dt. Since $|d\tilde{\boldsymbol{r}}/dt| \approx 1$ in practice, we omit the determinant term to simplify the computation.

3.4. Volumetric Recombination

Occlusion Disclosure. To inpaint the missing regions occluded by the hair in $\Phi^{\mathtt{hair}'}$ and produce the recombined 3D features $\Phi^{\mathtt{blend}}$ as depicted in Figure 2c, we optimize the latent vector $W^{\text{inp}} \in \mathcal{W}+$ within the StyleNeRF [15] framework to generate Φ^{inp}. Analogous to [50], we produce a volumetric target semantic mask, as depicted in Figure 5. For each iteration, we optimize W^{inp} to generate Φ^{inp} that aligns with the target semantic mask $\boldsymbol{s}^{\text{inp}}$ from a randomly selected viewpoint. Let $\boldsymbol{m}_{hair}$ denote a volumetric binary mask representing hair, $\boldsymbol{c}$ denote a camera parameter, and $\phi\big|_{\boldsymbol{c}}$ represents 2D features rendered by the camera parameter $\boldsymbol{c}$. The target semantics $\boldsymbol{s}^{\text{inp}}$ and the objectives for the optimization are constructed as follows:

$$\begin{aligned}\boldsymbol{s}^{\text{inp}} = \boldsymbol{m}_{hair}^{\mathtt{hair}'} \circ \boldsymbol{s}^{\mathtt{hair}'} + \Big(1 - \boldsymbol{m}_{hair}^{\mathtt{hair}'}\Big) \\ \circ \Big(\big(1 - \boldsymbol{m}_{hair}^{\mathtt{id}}\big) \circ \boldsymbol{s}^{\mathtt{id}} + \boldsymbol{m}_{hair}^{\mathtt{id}} \circ \boldsymbol{s}^{\mathtt{hair}'}\Big),\end{aligned} \tag{3}$$

$$\mathcal{L}^{\text{inp-sem}} = \mathbb{E}_{\boldsymbol{c}}\, \text{CE}\big(\mathcal{R}_{\boldsymbol{c}}(\boldsymbol{s}^{\text{inp}}),\ \mathcal{R}_{\boldsymbol{c}}(f^{\text{sem}}_{W^{\text{inp}}})\big), \tag{4}$$

$$\begin{aligned}\mathcal{L}^{\text{inp-reg}} = \mathbb{E}_{\boldsymbol{c}} \Big\| G\Big(\big(1 - \mathcal{R}_{\boldsymbol{c}}(\boldsymbol{m}_{hair}^{\text{inp}})\big) \circ U_W\big(\phi^{\text{inp}}\big|_{\boldsymbol{c}}\big)\Big) \\ - G\Big(\big(1 - \mathcal{R}_{\boldsymbol{c}^{\mathtt{id}}}(\boldsymbol{m}_{hair}^{\mathtt{id}})\big) \circ I^{\mathtt{id}}\Big)\Big\|_2^2,\end{aligned}$$

where $\mathcal{R}_{\boldsymbol{c}}$ represents the aggregation function of neural rendering according to the camera $\boldsymbol{c}$, f^{sem} denotes the MLP block with semantic head, U_W refers to the upsampler block of StyleNeRF whose output is a final image, and G denotes the Gram matrix for style loss [13]. Symbols marked with an apostrophe indicate that they are registered to their counterparts, as denoted by the associated superscripts. Unlike [50], our target semantic mask is defined as a continuous function within the volumetric space, rather than as a tensor defined at discrete positions. As a result, $\mathcal{L}^{\text{inp-sem}}$ entails integration over this volumetric space. Hence, for each optimization iteration in practice, we compute $\mathcal{L}^{\text{inp-sem}}$ using a random set of rays radiating from a randomly sampled camera $\boldsymbol{c}$ positioned on half of a hemisphere. Combining all elements, we obtain the inpainting feature Φ^{inp} as

$$\begin{aligned}\Phi^{\text{inp}} &= \text{MLP}(W^{\text{inp}*}), \\ \text{where} \quad W^{\text{inp}*} &= \arg\min_{W} \mathcal{L}^{\text{inp-sem}} + \lambda\, \mathcal{L}^{\text{inp-reg}}.\end{aligned} \tag{5}$$

Recombination. We make blend features in the volumetric space by summing up `id` and `hair` features, weighting them using binarized semantic maps representing the hair. Subsequently, the blend features are expected to be aggregated to 2D features using (2). However, the blend features contain missing regions that are occluded by the hair in the `id` image but are not fully covered by the hair in the `hair` image, leading to missing holes in the final rendered image. To mitigate this, we fill the missing regions within the blend features Φ^{blend} by the inpainting features Φ^{inp}, as obtained by (5). This blending concept is motivated by the Barbershop [50], but ours differs in that the features to be blended

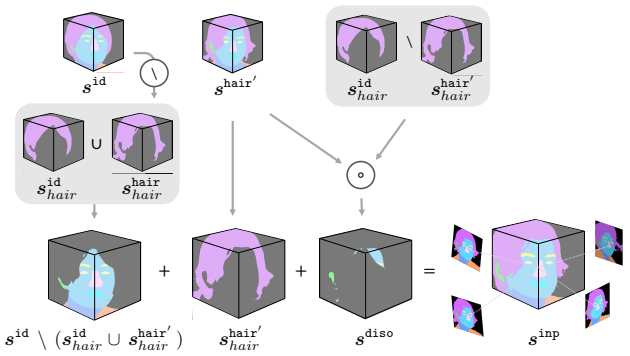


Figure 5: Target semantics generation for occlusion disclosure. The target semantics can be assumed to an ideal semantics for the hairstyle transferred result. The inpainting latent $W^{\texttt{inp}}$ is optimized to minimize to match the 2D semantics rendered from $\boldsymbol{s}^{\texttt{inp}}$ at a random viewpoint at every iteration.

lie in the volumetric space.

Let $\phi^{\texttt{id}}$ and $\phi^{\texttt{hair}}$ denote rendered 2D features of `id` and `hair`, respectively, rendered using neural rendering without ray deformation. Meanwhile, $\phi^{\texttt{id}'}$ and $\phi^{\texttt{hair}'}$ denote the registered 2D features of `id` and `hair`, aligned to their respective counterparts. Then we recombine features for blending as:

$$\begin{aligned}\phi^{\text{blend}}_{\texttt{id}} &= \boldsymbol{m}^{\texttt{hair}'}_{hair} \circ \phi^{\texttt{hair}'} + \left(1 - \boldsymbol{m}^{\texttt{hair}'}_{hair}\right) \\ &\quad \circ \left(\left(1 - \boldsymbol{m}^{\texttt{id}}_{hair}\right) \circ \phi^{\texttt{id}} + \boldsymbol{m}^{\texttt{id}}_{hair} \circ \phi^{\text{inp}} \right), \\ \phi^{blend}_{\texttt{hair}} &= \boldsymbol{m}^{\texttt{hair}}_{hair} \circ \phi^{\texttt{hair}} + \left(1 - \boldsymbol{m}^{\texttt{hair}}_{hair}\right) \\ &\quad \circ \left(\left(1 - \boldsymbol{m}^{\texttt{id}'}_{hair}\right) \circ \phi^{\texttt{id}'} + \boldsymbol{m}^{\texttt{id}'}_{hair} \circ \phi^{\text{inp}'} \right),\end{aligned} \tag{6}$$

where $\boldsymbol{m}_{hair}$ denotes a 2D binary mask for the hair semantics of ϕ, associated with a corresponding superscript.

3.5. Blending

Alignment-Aware Blending Loss. In the blending step, the upsampler blocks in the StyleNeRF [15] decode the blend features obtained by (6) into an output image in high resolution, conditioned on a latent vector W^{blend}. We optimize the latent vector W^{blend} to ensure that the decoded image retains the facial identity from the `id` image while seamlessly transferring hairstyle of the `hair` image. To this end, we design the objective function for the optimization to include face-preserving and hair-preserving terms.

$$\begin{aligned}\mathcal{L}^{\text{lpips}}_{face} &= \mathcal{L}^{\text{lpips}}\big((1 - \boldsymbol{m}^{\texttt{id}}_{hair}) \circ U_{W^{\text{blend}}}(\phi^{\text{blend}}_{\texttt{id}}), \\ &\qquad (1 - \boldsymbol{m}^{\texttt{id}}_{hair}) \circ I^{\texttt{id}}\big), \\ \mathcal{L}^{\text{lpips}}_{hair} &= \mathcal{L}^{\text{lpips}}\big(\boldsymbol{m}^{\texttt{hair}}_{hair} \circ U_{W^{\text{blend}}}(\phi^{\text{blend}}_{\texttt{hair}}), \\ &\qquad \boldsymbol{m}^{\texttt{hair}}_{hair} \circ I^{\texttt{hair}}\big), \\ \mathcal{L}^{\text{style}}_{hair} &= \Big\| G\big(\boldsymbol{m}^{\texttt{hair}}_{hair} \circ U_{W^{\text{blend}}}(\phi^{\text{blend}}_{\texttt{hair}})\big) \\ &\qquad - G\big(\boldsymbol{m}^{\texttt{hair}}_{hair} \circ I^{\texttt{hair}}\big) \Big\|^2_2,\end{aligned} \tag{7}$$

where $\boldsymbol{m}^{\texttt{id}}_{face}$ denotes the binary mask for face region of the face image, $\boldsymbol{m}^{\texttt{hair}}_{hair}$ denotes the binary mask for hair region of the hair image, U_W denotes the upsampler conditioned on W, $\mathcal{L}^{\text{lpips}}$ denotes the perceptual loss [49], and G denotes the Gram matrix for the style loss [13]. Combining all elements, the optimal blend vector $W^{\text{blend}*}$ is obtained as

$$W^{\text{blend}*} = \arg\min_{W} \lambda_{face}\mathcal{L}^{\text{lpips}}_{face} + \lambda_{hair}\mathcal{L}^{\text{lpips}}_{hair} + \mathcal{L}^{\text{style}}_{hair}. \tag{8}$$

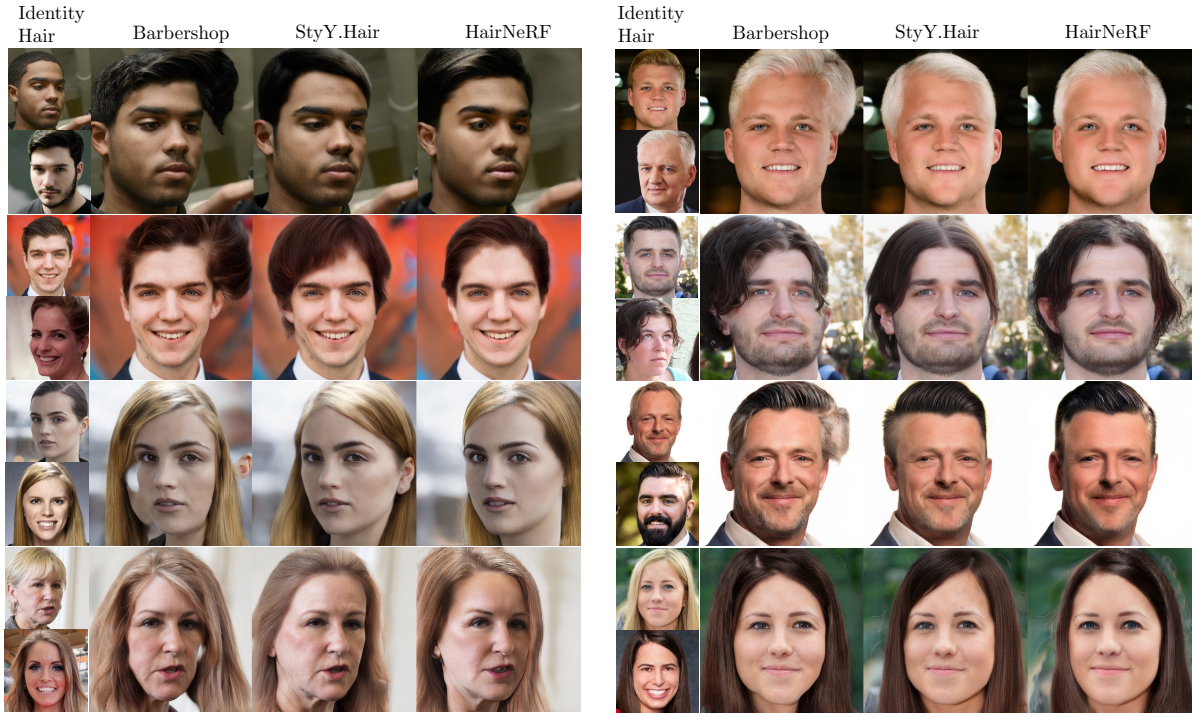
4. Experimental Results

We evaluate our method on two distinct tasks following [24]: hairstyle transfer and reconstruction. In the hairstyle transfer task, each input pair consists of different identities, and the goal is to transfer the hairstyle from the `hair` image to the face of the `id` image. In the reconstruction task, the input pair shares the same identity but with different poses. Since the `id` and `hair` images share the same identity in the reconstruction task, the `id` serves as the ground-truth for the evaluation. We use Flicker-Faces-HQ (FFHQ) dataset [21] and faces images crawled from Unsplash (UF), as used in [50, 51], for the hairstyle transfer task. For the reconstruction task, we employ the VoxCeleb2 [9] dataset. To evaluate the overall synthesis quality of the generated images, we measure the Frèchet Inception distance (FID) [17], the naturalness image quality estimator (NIQE) [34], and the precision and recall [26]. In the hairstyle transfer task, we synthesize 1,000 images using the FFHQ dataset and 380 images using the Unsplash dataset. For the reconstruction task, we synthesize 500 images. To demonstrate the effectiveness of HairNeRF at preserving geometric structure, we provide both qualitative and quantitative comparisons of our method against other baselines.

4.1. Comparison Study

We compare our method to LOHO [37], the Barbershop [50], and StyleYourHair [24] on the FFHQ and VoxCeleb2 datasets. For the hairstyle transfer task, we include a comparison with HairNet [51] using only the UF dataset.

Hairstyle Transfer. Quantitative results on FFHQ and UF are summarized in Table 1. Notice that LOHO, the Barbershop and StyleYourHair is built upon StyleGAN2, while HairNeRF utilizes StyleNeRF. Both StyleGAN2 and



(a) (b)

Figure 6: Comparison of results for FFHQ datasets. Both StyleYourHair and our method align the hair even if the poses between the input images are different, but our method preserves the direction of the hair better (first and the last rows).

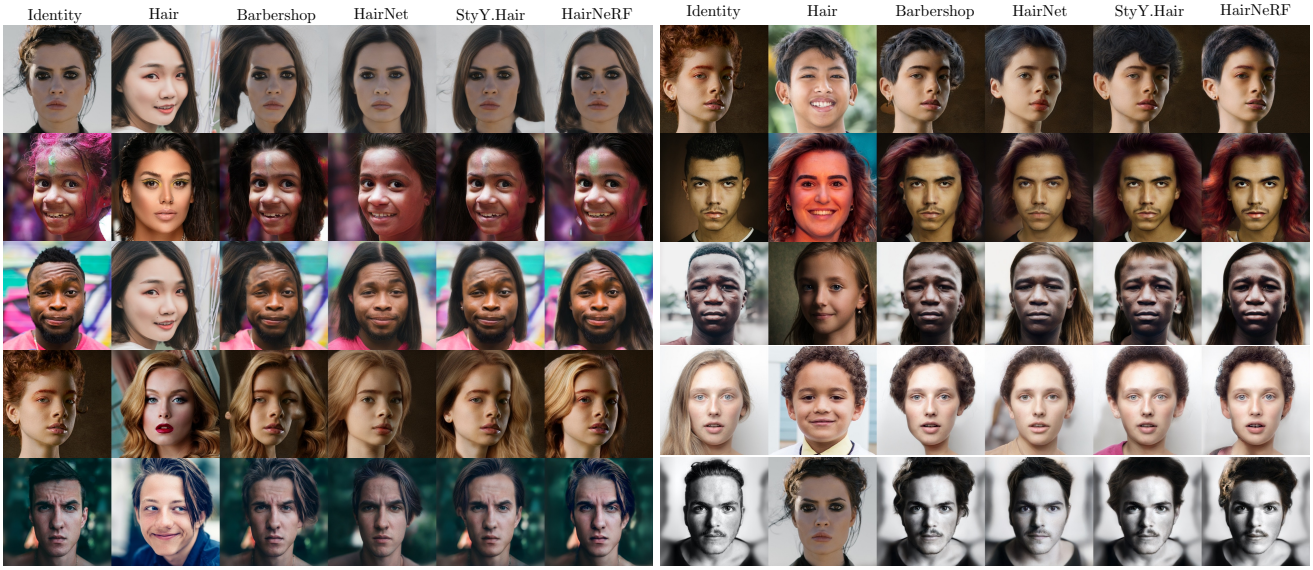


Figure 7: Comparison of results for Unsplash datasets. Our method preserves identity better (second row on the left), and hair transfer is performed to fit the structure of identity (third row on the left).

StyleNeRF are effective in generating high-quality face images; however, it has been reported that StyleGANs slightly outperforms NeRF-based model in terms of commonly used metrics, including FID. Consequently, the overall metrics for HairNeRF are slightly lower than the other baselines, with the exception of NIQE. Nevertheless, this difference is not substantial, and the qualitative results of HairNeRF reveal perceptually better quality in comparison to the other

	FFHQ					
Method	NIQE↓	ArcFace↑	Precision↑	Recall↑	Realism↑	FID↓
LOHO	40.65	**0.83**	0.85	0.55	1.34	91.42
Barbershop	12.24	0.77	0.93	**0.83**	1.06	**48.00**
StyleYourHair	12.00	0.55	0.93	0.77	1.11	51.36
HairNet	11.63	0.76	**0.97**	0.59	1.14	56.45
HairNeRF	**9.57**	0.70	0.89	0.67	**1.40**	57.26

	UF					
Method	NIQE↓	ArcFace↑	Precision↑	Recall↑	Realism↑	FID↓
LOHO	42.37	**0.80**	0.91	0.45	1.07	86.16
Barbershop	12.13	0.74	**0.94**	**0.84**	1.91	**41.06**
StyleYourHair	12.07	0.75	**0.94**	0.80	**2.17**	42.62
HairNeRF	**10.09**	0.70	0.89	0.57	1.71	49.74

Table 1: Hairstyle transfer results on FFHQ and UF datasets. HairNeRF shows comparable results to StyleGAN2-based hairstyle transfer models.

Figure 8: Comparison examples for reconstruction task.

baselines. Figures 6 and 7 show synthesized examples using the FFHQ and UF datasets, respectively. As observed in the fifth row of Figure 6b and Figure 7, the Barbershop generates fine-quality images when the poses of the `id` and `hair` images are similar. However, the output quality substantially declines when the difference in pose between the two input images increases. As shown in the second and the third row of Figure 6a, substantial structural artifacts, such as bumps and missing holes, appear in the transferred hairstyle images synthesized by the Barbershop. In contrast, the transferred hairstyle of HairNeRF seamlessly fits the head of the `id` image. StyleYourHair and HairNet also generate satisfactory results for input pairs with significant pose differences. However, as seen in the fifth row of Figure 6b, the detailed structure of the hairstyle often changes in the synthesized results. Furthermore, they often elongate the length of the forehead, as shown in the fourth row of Figure 7a.

Reconstruction. Following [24], we evaluate reconstruction quality on the VoxCeleb2 dataset, consisting of pairs of images with same identity but different poses. Figure 8 shows an example of reconstruction task. Ideally, the synthesis result should be identical to the `id` image, since the `id` and `hair` images represent the same person. Figure 8 shows the reconstruction results of HairNeRF in comparison to other baselines. While the reconstructed image preserves the detailed geometry of the hair, both the Barbershop and StyleYourHair show slight alterations. The quantitative resutls, summarized in Table 2, further validate the superiority of HairNeRF in preserving the geometry of hairstyles.

Method	MSE↓	SSIM↑	LPIPS↓	PSNR↑
Barbershop [50]	1.165	0.801	0.127	31.44
StyleYourHair [24]	1.218	0.795	0.155	31.34
HairNeRF	**0.957**	**0.821**	**0.106**	**31.84**

Table 2: Reconstruction result. HairNeRF surpasses other baselines in view of MSE, SSIM, LPIPS, and PSNR. The result verifies the effectiveness of HairNeRF at preserving hairstyle geometry compared to the other baselines.

Figure 9: Examples of novel view synthesis.

Novel View Synthesis. Owing to the geometric nature of neural rendering, HairNeRF can generate diverse pose varying images from the same latent vector once optimized for synthesis. Figure 9 shows examples of novel view images obtained by varying the camera parameters. For a qualitative comparison, we generate a diverse set of novel view images of the `id` image using the inversion result of HairNeRF. We then synthesize each novel view image using the Barbershop and StyleYourHair from the given `hair` image. As shown in Figure 9, the transferred hairstyle varies by views in the case of baselines, whereas HairNeRF preserves a consistent hair structure that aligns with the original `hair` image.

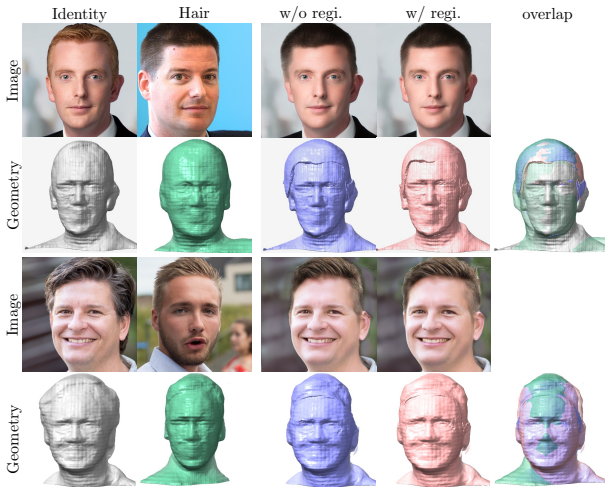


Figure 10: Effectiveness of registration shown with the underlying geometry. Without registration the transferred hairstyle does not perfectly fit to the face (first row) or erode the head (third row) of the `id` image.

Method	MSE↓	SSIM↑	LPIPS↓	PSNR↑
w/o regi.	0.980	0.818	0.108	31.77
w/ regi.	**0.957**	**0.821**	**0.106**	**31.84**

Table 3: Reconstruction quality with and without head alignment module. With registration, the reconstruction quality increases in view of MSE, SSIM, LIPIS, and PSNR.

4.2. Ablation Study

Effectiveness of Registration. To validate the effectiveness of the head alignment module, we evaluate the reconstruction quality using the VoxCeleb2 dataset. Table 3 summarizes the results. As mentioned in Section 3.3, StyleNeRF suffers from an intrinsic misalignment problem that results in subtle differences, even if the pair of input images shares the same identity. However, the registration process within the head alignment module reduces the existing geometric differences to make transferred hairstyle more fit to the `id` image. Table 3 shows that the reconstruction quality deteriorates when the registration module is missing. Figure 10 shows the effectiveness of registration with the underlying geometry. The first example shows that the shape of the original hair is slightly larger than the head of `id` image. Without registration, the transferred hair sticks out to the right side of the head of the `id`. However, with registration, the transferred hair fits more closely to the head of the `id`. The difference is clearly shown in the overlap image. In the second example, it is clearly shown that the transferred hair is stretched to fit on the head of `id` by registration.

5. Conclusion

We introduce *HairNeRF*, a novel geometry-aware hairstyle transfer method designed to address the entanglement issue of pose and style attributes within the latent space of StyleGAN. By leveraging NeRF-based generative models, where the pose of the rendered image is determined by external camera parameters, we achieve a complete disentanglement of pose from the style latent space. Additionally, we propose an efficient method for recombining features within the volumetric space using semantics. This is achieved by aligning two distinct innate head structures via ray deformation. This method unveils the potential to seamlessly integrate multiple NeRF-represented objects of varying shapes and sizes into one unified object, without requiring additional modification to NeRF weights. Using the intrinsic geometric properties of neural rendering, our method can produce view-varied images from a single synthesis process without inducing geometric deformation. The efficacy of HairNeRF in preserving the geometric structure of the original hairstyle, compared to other baselines, is substantiated both qualitatively and quantitatively through comprehensive experiments on diverse datasets.

References

[1] Rameen Abdal, Peihao Zhu, John Femiani, Niloy J. Mitra, and Peter Wonka. Clip2stylegan: Unsupervised extraction of stylegan edit directions. In *SIGGRAPH*, 2022. 2

[2] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *Int. Conf. Comput. Vis.*, 2021. 2

[3] Guha Balakrishnan, Raghudeep Gadde, Aleix Martinez, and Pietro Perona. Rayleigh eigendirections (reds): Nonlinear GAN latent space traversals for multidimensional features. In *Eur. Conf. Comput. Vis.*, 2022. 2

[4] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay andgu2021stylenerf Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3d generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 3

[5] Eric R. Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. Pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 3

[6] Anpei Chen, Ruiyang Liu, Ling Xie, Zhang Chen, Hao Su, and Jingyi Yu. Sofgan: A portrait image generator with dynamic styling. *ACM Trans. Graph.*, 41(1):1:1–1:26, 2022. 3

[7] Zikun Chen, Ruowei Jiang, Brendan Duke, Han Zhao, and Parham Aarabi. Exploring gradient-based multi-directional controls in gans. In *Eur. Conf. Comput. Vis.*, 2022. 2

[8] Min Jin Chong, Wen-Sheng Chu, Abhishek Kumar, and David A. Forsyth. Retrieve in style: Unsupervised facial feature transfer and retrieval. In *Int. Conf. Comput. Vis.*, 2021. 2

[9] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. In B. Yegnanarayana, editor, *Conf. of the Int. Speech Comm. Assoc.*, 2018. 6

[10] Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. GRAM: generative radiance manifolds for 3d-aware image generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 3

[11] Jean Duchon. Splines minimizing rotation-invariant seminorms in sobolev spaces. In Walter Schempp and Karl Zeller, editors, *Constructive Theory of Functions of Several Variables: Proceedings of a Conference Held at Oberwolfach, Germany, April 25 - May 1, 1976*, volume 571 of *Lecture Notes in Mathematics*, pages 85–100, 1976. 4

[12] Rinon Gal, Or Patashnik, Haggai Maron, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Trans. Graph.*, 41(4):141:1–141:13, 2022. 2

[13] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 5, 6

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[15] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. Stylenerf: A style-based 3d aware generator for high-resolution image synthesis. In *Int. Conf. Learn. Represent.*, 2022. 2, 3, 4, 5, 6

[16] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable GAN controls. In *Adv. Neural Inform. Process. Syst.*, 2020. 2

[17] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Adv. Neural Inform. Process. Syst.*, 2017. 6

[18] Kaiwen Jiang, Shu-Yu Chen, Feng-Lin Liu, Hongbo Fu, and Lin Gao. Nerffaceediting: Disentangled face editing in neural radiance fields. In *SIGGRAPH Asia*, 2022. 3

[19] Omer Kafri, Or Patashnik, Yuval Alaluf, and Daniel Cohen-Or. Stylefusion: A generative model for disentangling spatial segments. *ACM Trans. Graph.*, 41(5):197:1–197:15, 2022. 2

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(12):4217–4228, 2021. 2, 6

[22] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[23] Hyunsu Kim, Yunjey Choi, Junho Kim, Sungjoo Yoo, and Youngjung Uh. Exploiting spatial dimensions of latent in GAN for real-time image editing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 852–861, 2021. 2

[24] Taewoo Kim, Chaeyeon Chung, Yoonseo Kim, Sunghyun Park, Kangyeol Kim, and Jaegul Choo. Style your hair: Latent optimization for pose-invariant hairstyle transfer via local-style-aware hair alignment. In *Eur. Conf. Comput. Vis.*, 2022. 2, 3, 6, 8

[25] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing nerf for editing via feature field distillation. In *Adv. Neural Inform. Process. Syst.*, 2022. 3

[26] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *Adv. Neural Inform. Process. Syst.*, 2019. 6

[27] Verica Lazova, Vladimir Guzov, Kyle Olszewski, Sergey Tulyakov, and Gerard Pons-Moll. Control-nerf: Editable feature volumes for scene rendering and manipulation. In *Winter Conf. on Appl. of Comput. Vis.*, 2023. 3

[28] Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Trans. Graph.*, 36(6):194:1–194:17, 2017. 4

[29] Huan Ling, Karsten Kreis, Daiqing Li, Seung Wook Kim, Antonio Torralba, and Sanja Fidler. Editgan: High-precision semantic image editing. pages 16331–16345, 2021. 2

[30] Steven Liu, Xiuming Zhang, Zhoutong Zhang, Richard Zhang, Jun-Yan Zhu, and Bryan Russell. Editing conditional radiance fields. In *Int. Conf. Comput. Vis.*, 2021. 3

[31] Tetiana Martyniuk, Orest Kupyn, Yana Kurlyak, Igor Krashenyi, Jiri Matas, and Viktoriia Sharmanska. Dad-3dheads: A large-scale dense, accurate and diverse dataset for 3d head alignment from a single image. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 4

[32] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Eur. Conf. Comput. Vis.*, 2020. 2

[33] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2022. 1, 2

[34] Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Sign. Process. Letters*, 20(3):209–212, 2013. 6

[35] Michael Niemeyer and Andreas Geiger. GIRAFFE: representing scenes as compositional generative neural feature fields. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 3

[36] Daniel Roich, Ron Mokady, Amit H Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Trans. Graph.*, 42(1):1–13, 2022. 4

[37] Rohit Saha, Brendan Duke, Florian Shkurti, Graham W Taylor, and Parham Aarabi. Loho: Latent optimization of hairstyles via orthogonalization. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2, 6

[38] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. GRAF: generative radiance fields for 3d-aware image synthesis. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Adv. Neural Inform. Process. Syst.*, 2020. 3

[39] Jingxiang Sun, Xuan Wang, Yichun Shi, Lizhen Wang, Jue Wang, and Yebin Liu. IDE-3D: interactive disentangled editing for high-resolution 3d-aware portrait synthesis. *ACM Trans. Graph.*, 41(6):270:1–270:10, 2022. 3

[40] Jingxiang Sun, Xuan Wang, Yong Zhang, Xiaoyu Li, Qi Zhang, Yebin Liu, and Jue Wang. Fenerf: Face editing in neural radiance fields. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 3

[41] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Trans. Graph.*, 40(4):133:1–133:14, 2021. 2

[42] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. GAN inversion: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(3):3121–3138, 2023. 2

[43] Jianfeng Xiang, Jiaolong Yang, Yu Deng, and Xin Tong. GRAM-HD: 3d-consistent image generation at high resolution with generative radiance manifolds. *arXiv*, 2022. 3

[44] Bangbang Yang, Yinda Zhang, Yinghao Xu, Yijin Li, Han Zhou, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Learning object-compositional neural radiance field for editable scene rendering. In *Int. Conf. Comput. Vis.*, 2021. 3

[45] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet V2: bilateral network with guided aggregation for real-time semantic segmentation. *Int. J. Comput. Vis.*, 129(11):3051–3068, 2021. 4

[46] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Eur. Conf. Comput. Vis.*, 2018. 4

[47] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Eur. Conf. Comput. Vis.*, 2018. 4

[48] Yu-Jie Yuan, Yang-Tian Sun, Yu-Kun Lai, Yuewen Ma, Rongfei Jia, and Lin Gao. Nerf-editing: Geometry editing of neural radiance fields. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 3

[49] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 6

[50] Peihao Zhu, Rameen Abdal, John Femiani, and Peter Wonka. Barbershop: Gan-based image compositing using segmentation masks. *ACM Trans. Graph.*, 40(6):215:1–215:13, 2021. 2, 3, 5, 6, 8

[51] Peihao Zhu, Rameen Abdal, John Femiani, and Peter Wonka. Hairnet: Hairstyle transfer with pose changes. In *Eur. Conf. Comput. Vis.*, 2022. 2, 6