CG-NeRF: Conditional Generative Neural Radiance Fields for 3D-aware Image Synthesis

Kyungmin Jo† Gyumin Shim† Sanghun Jung Soyoung Yang
Jaegul Choo
Korea Advanced Institute of Science and Technology (KAIST)
Daejeon, Korea
{bttkm, shimgyumin, shjung13, sy_yang, jchoo}@kaist.ac.kr

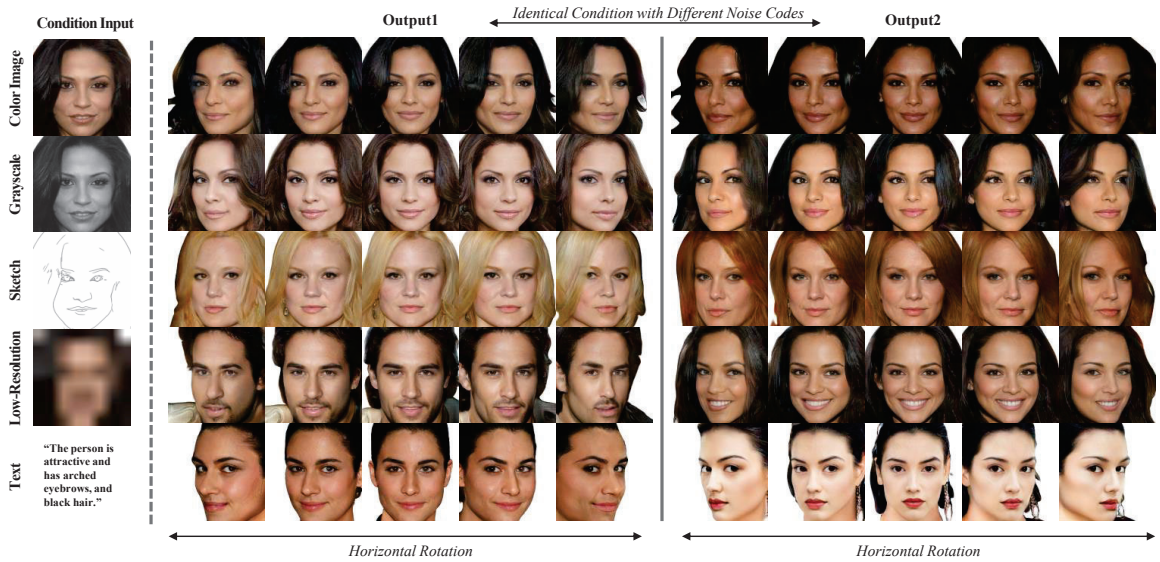


Figure 1: Our method produces diverse 3D-aware output images reflecting various condition inputs (the first column). For each condition input, two different output images generated with different noise codes are shown with horizontal rotation.

Abstract

Recent generative models based on neural radiance fields (NeRF) achieve the generation of diverse 3D-aware images. Despite the success, their applicability can be further expanded by incorporating with various types of user-specified conditions such as text and images. In this paper, we propose a novel approach called the conditional generative neural radiance fields (CG-NeRF), which generates multi-view images that reflect multimodal input conditions such as images or text. However, generating 3D-aware images from multimodal conditions bears several challenges. First, each condition type has different amount of information - e.g., the amount of information in text and color images are significantly different. Furthermore, the pose-consistency is often violated when diversifying the generated images from input conditions. Addressing such challenges, we propose 1) a unified architecture that effectively handles multiple types of conditions, and 2) the pose-consistent diversity loss for generating various images while maintaining the view consistency. Experimental results show that the proposed method maintains consistent image quality on various multimodal condition types and achieves superior fidelity and diversity compared to the existing NeRF-based generative models.

1. Introduction

The neural radiance field (NeRF) [18] successfully addresses unseen view synthesis, a long-lasting problem in computer vision, by learning to construct a 3D scene from

† Both authors contributed equally to this research.

a set of images taken from multiple viewpoints via a differentiable rendering technique. Because NeRF takes the 3D coordinate and the viewpoint of a target scene as inputs, it is capable of synthesizing view-consistent images (*i.e.,* images corresponding to the input view points). Due to the success of NeRF, this approach has been widely extended to various fields, such as view-aware video synthesis [13, 39], pose estimation [34], scene labeling and understanding [48], and 3D object modeling from a collection of single-category images [40].

While these techniques utilize NeRF only for synthesizing an unseen view of an image, recent studies that generate photorealistic multi-view images based on generative adversarial networks(GANs) [30, 23, 2, 25, 1] have emerged. Compared to the existing 2D-based generative models, these studies produce 3D-aware images by generating view-consistent images for given camera poses. However, because the generative models synthesize images without any user-specified condition, these studies require a test-time optimization [2] for generation of images that contain the desired characteristics of the condition, as shown in Fig. 1.

Overcoming such a point and extending the capability of the existing unconditional generative NeRF models, we perform 3D-aware image synthesis that reflects the given multimodal conditions. The proposed task, conditional generative NeRF (CG-NeRF), aims to create view-consistent and *diverse* images by reflecting the characteristics of *conditions*. To the best of our knowledge, our work is the first to tackle this task, extending the existing generative NeRF approaches that does not take user-specified multimodal conditions.

In this paper, we propose a unified method adaptively applicable to various condition types, including color images, grayscale images, sketches, low-resolution images, and text, as shown in the condition inputs in Fig. 1. Since different types of conditions have disparate amounts of information, it is challenging to generate images from various types of conditions with a unified architecture. To tackle this problem, we provide the model with coarse characteristics of input conditions extracted from a semantic multimodal encoder, and additional noise codes to fill the missing fine details in the coarse characteristics. We show that our method consistently generates diverse photo-realistic images regardless of condition types in Sec. 4.

For the diversity of the generated images, we design a model capable of creating fine details while reflecting the coarse characteristics of the input conditions. However, unlike the previous unconditional models, an input condition may excessively decrease the diversity of synthesized images. While the diversity sensitive loss helps in generating various images in 2D-based conditional generative models [43, 3], the pose consistency can be violated in 3D-based generative models as shown in Fig. 6. To address such difficulties, we propose a novel pose-consistent diversity (PD) loss that induces the model to generate diverse images but explicitly penalizes view inconsistencies.

In summary, our contributions are as follows:

- We propose a unified architecture called the conditional generative neural radiance fields(CG-NeRF), which generates diverse and photo-realistic images by reflecting the multimodal condition inputs and effectively disentangling the shape and appearance from the input conditions.
- To improve the diversity of the output images, we propose the pose-consistent diversity (PD) loss, which helps in producing various images while maintaining the view consistency.
- We conduct extensive experiments and demonstrate that our unified model generates diverse images, reflecting various types of input conditions.

2. Related Work

Neural Radiance Fields Recent advancements [18, 5, 9, 33, 45, 14] in the area of novel view synthesis have been accomplished by employing the NeRF. The seminal work [18] has proven the effectiveness of volume rendering with NeRF, and later studies [5, 38, 46] proposed further improvements over the original NeRF. While some NeRF studies enhance the original NeRF in terms of both quality and efficiency, our work is more related to generative NeRF methods, which have attracted attention recently.

Generative NeRF Along with the improvements to the NeRF itself, generative NeRF models [30, 23, 2, 21] have also emerged. GRAF [30] proposes a generative model with implicit radiance fields for the novel scene generation. Moreover, GIRAFFE [23] improves GRAF by separating the object instances in a controllable way, which lets users gain more ability to compose new scenes. Another study, pi-GAN [2], which is more closely related to our work, employs the SIREN [32] activation function along with the multilayer perceptron (MLP), which is effective when used for novel scene generation. Furthermore, some approaches have attempted to add conditions or users' constraint while generating. Few-shot novel view synthesis [45, 20] targets to reconstruct images observed from novel views, conditioned on sparse input images. However, they have limitations in generating diverse images, and they require ground truth multi-view images for training. Edit-NeRF [15] proposes editable NeRF, which can edit shapes and textures of output images by varying the latent codes. Some studies [37, 4] suggest optimization based method that can satisfy the user's constraints using real images or text. Dreamfields [8], which is a concurrent work, takes text as input to

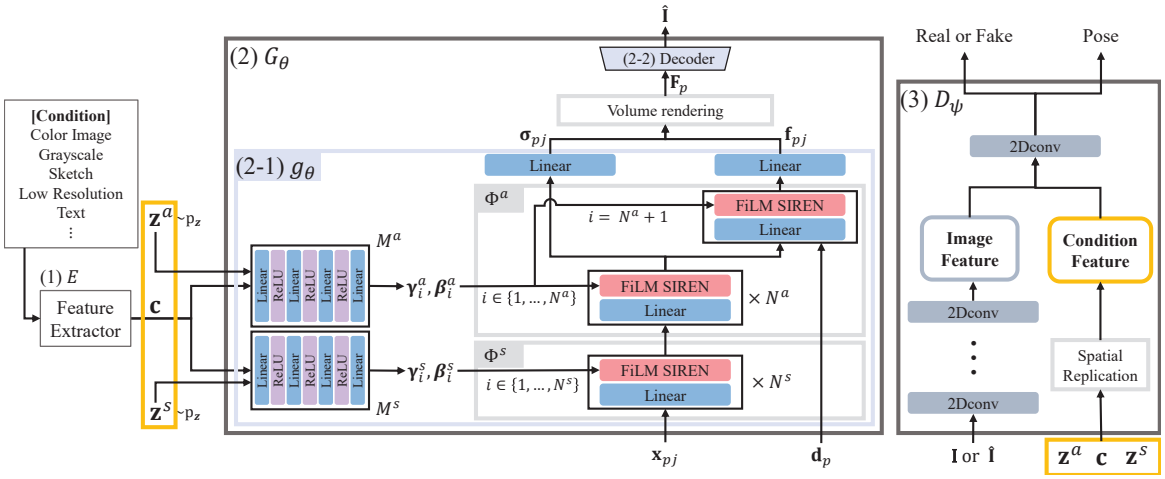


Figure 2: Illustration of our main architecture. Notations are summarized in Table 1.

synthesize images. However, those approaches require test-time optimization or only handle a limited type of condition data. Therefore, in this work, we propose a novel model, CG-NeRF, which can significantly improve the applicability of the NeRF methods and allow users to generate various scenes according to diverse conditions.

CLIP The conditions from which we want to generate images can exist in various forms, typically in the form of images or text. To address both cases at the same time, a model that can take multimodal inputs is required. Among such models [42, 35, 27], CLIP [27] shows an impressive ability to embed text and image information into the same semantic space. We adopt CLIP as our global feature extractor in various conditions, making our model widely applicable for both images and text.

3. Proposed Approach

3.1. Overview

We propose a novel method called conditional generative NeRF (CG-NeRF), which can generate camera-pose-dependent images conditioned on various types of input data. Unlike recent unconditional generative models that learn neural radiance fields from unlabeled 2D images, we extend the generative model to a conditional model utilizing extra information as input, such as text, sketches, grayscale, low-resolution images, or even color images. We design a model that can generate diverse images with different details, sharing the coarse characteristics of condition inputs. As shown in Fig. 2, the global feature vector $\mathbf{c}$ extracted from the input condition is fed to the network along with the noise codes $\mathbf{z}^s$ and $\mathbf{z}^a$ randomly sampled from a standard Gaussian distribution p_z. The noise codes specify fine details that are not contained in the given global features. In the proposed model, the generator G_θ ((2) in Fig. 2) learns radiance field representations and synthesizes images $\hat{\mathbf{I}}$ corresponding to the given global feature vector $\mathbf{c}$ and noise codes $\mathbf{z}^s$ and $\mathbf{z}^a$, *i.e.,*

$$\hat{\mathbf{I}} = G_\theta(\boldsymbol{\xi}, \mathbf{c}, \mathbf{z}^s, \mathbf{z}^a), \tag{1}$$

where $\boldsymbol{\xi}$ is the camera pose for calculating the 3D coordinate $\mathbf{x}$ and the viewing direction $\mathbf{d}$ [30]. Below, we describe the model structure designed for CG-NeRF in detail.

	Notation	Name
Input	$\mathbf{x} \in \mathbb{R}^3$	3D coordinate
	$\mathbf{d} \in \mathbb{R}^2$	Viewing direction
	$\mathbf{c} \in \mathbb{R}^{L_c}$	Global feautre vector
	$\mathbf{z}^s \in \mathbb{R}^{L_s}$	Shape noise code
	$\mathbf{z}^a \in \mathbb{R}^{L_a}$	Appearance noise code
Output	$\boldsymbol{\gamma}_i^s, \boldsymbol{\gamma}_i^a \in \mathbb{R}^{L_\gamma}$	Frequency
	$\boldsymbol{\beta}_i^s, \boldsymbol{\beta}_i^a \in \mathbb{R}^{L_\beta}$	Phase shift
	$\sigma_{pj} \in \mathbb{R}$	Density
	$\mathbf{f}_{pj} \in \mathbb{R}^{L_f}$	Feature vector
	$\mathbf{F}_p \in \mathbb{R}^{L_f}$	Rendered feature
	$\mathbf{I}, \hat{\mathbf{I}} \in \mathbb{R}^{H \times W \times 3}$	Real/Generated image
Function	$g_\theta : \mathbb{R}^{L_c+L_s+L_a+5} \mapsto \mathbb{R}^{2L_f}$	Feature fields generator
	$M^s : \mathbb{R}^{L_c+L_s} \mapsto \mathbb{R}^{N^s \times (L_\gamma+L_\beta)}$	Shape mapping network
	$M^a : \mathbb{R}^{L_c+L_a} \mapsto \mathbb{R}^{(N^a+1) \times (L_\gamma+L_\beta)}$	Appearance mapping network
	$\Phi^s : \mathbb{R}^3 \mapsto \mathbb{R}^{L_f}$	Shape block
	$\Phi^a : \mathbb{R}^{L_f+2} \mapsto \mathbb{R}^{2L_f}$	Appearance block

Table 1: Summarized notations. $p \in \{1, \cdots, H_V W_V\}$, and $j \in \{1, \cdots, J\}$. J indicates the number of sampling points per ray. $H \times W$ and $H_V \times W_V$ are the spatial resoultion of image and features, respectively.

3.2. Model Architecture of CG-NeRF

As illustrated in Fig. 2, the main architecture consists of three components: (1) a feature extractor E that extracts global feature vectors from the given conditions, (2) a generator that creates an image by reflecting the conditions, and

(3) a discriminator that distinguishes real images from fake images based on the condition input and that predicts the camera poses of fake images for the PD loss, which will be described in detail later.

As CG-NeRF aims to synthesize conditional 3D-aware images, the condition input is encoded to a global feature vector through the global feature extractor E ((1) in Fig. 2). To extract global semantic features from the given condition inputs in our case, we adopt CLIP [26], accomodating various types of input conditions such as images and text, as a state-of-the-art multimodal encoder.

We design our generator network by combining two recent promising techniques, which are proven to generate high-quality images for the generative neural radiance field task: a SIREN-based backbone [2] and a feature-level volume rendering method [23]. The SIREN-based [32] network architecture enhances the visual quality of the NeRF-based generative model but requires a large amount of memory for training due to color-level volume rendering at the full image resolution [2]. To address this issue, we leverage feature-level volume rendering, inspired by a recently proposed method [23]. The feature-level volume rendering process substantially mitigates the problem because a volume is rendered at the level of feature vector $\mathbf{f}$, having a smaller scale than the image resolution.

Given a global feature vector $\mathbf{c}$, a noise code of shape $\mathbf{z}^s$ and appearance $\mathbf{z}^a$, the feature fields generator g_θ ((2-1) in Fig. 2) produces the density σ and feature vector $\mathbf{f}$ in the corresponding $\mathbf{x}$ and $\mathbf{d}$ as

$$g_\theta(\mathbf{x}_{pj}, \mathbf{d}_p, \mathbf{c}, \mathbf{z}^s, \mathbf{z}^a) = (\sigma_{pj}, \mathbf{f}_{pj}), \tag{2}$$

where σ_{pj} and $\mathbf{f}_{pj}$ denote the density and the feature vector, respectively, at the corresponding 3D coordinate. Further details are described in the next section.

Once the density σ and the feature vector $\mathbf{f}$ are estimated by the feature fields generator g_θ ((2-1) in Fig. 2) at each 3D coordinate, the final feature $\mathbf{F_p} \in \mathbb{R}^{\mathbf{L_f}}$ is computed through a feature-level volume rendering process as

$$\mathbf{F}_p = \sum_{j=1}^{J} T_{pj} \alpha_{pj} \mathbf{f}_{pj}, \tag{3}$$

where the transmittance $T_{pj} = \prod_{k=1}^{j-1} (1 - \alpha_{pk})$. The alpha value for $\mathbf{x}_{pj}$ is calculated as $\alpha_{pj} = 1 - e^{-\sigma_{pj}\delta_{pj}}$, and δ_{pj} is the distance between neighboring sample points along the ray direction [18]. The 2D feature map $\mathbf{F} \in \mathbb{R}^{H_V \times W_V \times L_f}$ rendered through the volume rendering process is then upsampled to a RGB images at a higher resolution $\hat{\mathbf{I}} \in \mathbb{R}^{H \times W \times 3}$ using the 2D convolutional neural network (CNN) decoder network ((2-2) in Fig. 2). The decoder network consists of CNN layers with leaky ReLU activation functions [41] and nearest neighbor upsampling layers.

3.3. Condition-based Disentangling Network

We propose a novel approach that aims to disentangle both the shape and appearance contained in a given global feature vector. For a text condition example " round bird with a red body", "round" and "bird" are shapes, and "red" is an attribute indicating the appearance. Two mapping networks M^s and M^a serve to generate the styles of the shape and appearance, respectively, from the global feature vector $\mathbf{c}$ and noise codes $\mathbf{z}^s$ and $\mathbf{z}^a$. The global feature vector $\mathbf{c} \in \mathbb{R}^{L_c}$ contains the prominent attribute of the condition. In constrast, the noise codes $\mathbf{z}^s \in \mathbb{R}^{L_s}$ and $\mathbf{z}^a \in \mathbb{R}^{L_a}$ are responsible for the details that the global feature vector does not include. The mapping network consists of pairs of a linear layer and ReLU and produces frequencies $\boldsymbol{\gamma}$ and phase shifts $\boldsymbol{\beta}$ as

$$\begin{aligned} M^s(\mathbf{c}, \mathbf{z^s}) &= \text{cat}\{(\boldsymbol{\gamma}_i^s, \boldsymbol{\beta}_i^s)\}_{i=1\cdots N^s} \\ M^a(\mathbf{c}, \mathbf{z^a}) &= \text{cat}\{(\boldsymbol{\gamma}_i^a, \boldsymbol{\beta}_i^a)\}_{i=1\ldots N^a+1}, \end{aligned} \tag{4}$$

where N^s and N^a denote the numbers of MLPs in each block. *cat* indicates channel-wise concatenation. The predicted frequencies and phase shifts are fed to the two blocks Φ^s and Φ^a in the feature fields generator. Taking these as inputs along with the 3D coordinate $\mathbf{x}$ and the direction $\mathbf{d}$, two consecutive blocks encode features using pairs of a linear layer and activation function of feature-wise linear modulation (FiLM) SIREN. The sine function of the FiLM SIREN layer modulated by the obtained frequency and phase shift are applied to the outputs of the linear layers as an activation function; *i.e.,*

$$\phi_i(\mathbf{y}_i) = \sin(\boldsymbol{\gamma}_i(\mathbf{W}_i \mathbf{y}_i + \mathbf{b}_i) + \boldsymbol{\beta}_i), \tag{5}$$

where $\phi_i : \mathbb{R}^{M_i} \mapsto \mathbb{R}^{N_i}$ is the i-th MLP of each Φ^s and Φ^a. $\mathbf{W}_i \in \mathbb{R}^{N_i \times M_i}$ and $\mathbf{b}_i \in \mathbb{R}^{N_i}$ are the weight and the bias applied to input $\mathbf{y}_i \in \mathbb{R}^{M_i}$. The two blocks in the feature fields generator have the following formulations:

$$\begin{aligned} &\Phi^s(\mathbf{x}_{pj}) = \phi^s_{N^s}\left(\phi^s_{N^s-1}(\cdots \phi^s_1(\mathbf{x}_{pj}))\right), \\ &\Phi^a(\Phi^s(\mathbf{x}_{pj}), \mathbf{d}_p) = \phi^a_{N^a+1}(\text{cat}(\phi^a_{N^a}(\cdots \phi^a_1(\Phi^s(\mathbf{x}_{pj}))), \mathbf{d}_p)). \end{aligned} \tag{6}$$

Inspired by an existing approach [30], we assign the roles of reflecting the shape to the first block, close to the input, and the appearance to the second block, close to the output. The block for shape utilizes the 3D coordinate as the input to generate shape-encoded features, while the appearance block takes the output of the previous block as input and generates encoded features of the shape and appearance. By utilizing these features and viewing directions as inputs, features reflecting the viewing direction are generated from the last layer of the appearance block.

3.4. Pose-consistent Diversity Loss

As our method generates images conditioned on extra inputs, variations of the output images are restricted, especially when a color image is given as a condition input. To

enable the generator network to produce semantically diverse images based on the condition input, we regularize the generator network with the diversity-sensitive loss [43]. This is defined as

$$\mathcal{L}_{\text{div}}(\theta) = \mathbb{E}_{\mathbf{z}^s,\mathbf{z}^a\sim p_z,\boldsymbol{\xi}\sim p_\xi,\mathbf{c}\sim p_r}[\| \hat{\mathbf{I}}_1 - \hat{\mathbf{I}}_2 \|_1], \quad (7)$$

where $\hat{\mathbf{I}}_1$ is $G_\theta(\boldsymbol{\xi}, \mathbf{c}, \mathbf{z}^{s1}, \mathbf{z}^{a1})$ and $\hat{\mathbf{I}}_2$ is $G_\theta(\boldsymbol{\xi}, \mathbf{c}, \mathbf{z}^{s2}, \mathbf{z}^{a2})$.

However, we empirically discover that simply applying the diversity-sensitive loss causes undesirable effects that attempt to change not only the style but also the pose of the output images (Fig. 6). Because the pose of the output images should be determined only by the input camera pose $\boldsymbol{\xi}$, pose changes in the output images are a significant side effect. We analyze this undesirable phenomenon as follows; from the generator network's point of view, the model maximizes the pixel difference via two different methods: (1) changing the style of the output images as desired or (2) changing the poses between two output images generated with the same camera pose, which is strongly undesired.

To explicitly address such an issue, we propose a pose regularization term applicable to the original diversity-sensitive loss, which explicitly penalizes pose difference between images generated from different noise codes $\mathbf{z}^s$ and $\mathbf{z}^a$ but from the same camera pose. The intuition behind the proposed regularization is that the model generates two images to have only a style difference constrained to have the same pose, which can be additionally learned by an auxiliary network. We propose to add the regularization term $\mathcal{L}_{\text{pose}}$ to the diversity-sensitive loss $\mathcal{L}_{\text{div}}$, which is defined as

$$\mathcal{L}_{\text{pose}}(\theta) = \mathbb{E}_{\mathbf{z}^s,\mathbf{z}^a\sim p_z,\boldsymbol{\xi}\sim p_\xi,\mathbf{c}\sim p_r}[1-\cos(D^\xi_\psi(\hat{\mathbf{I}}_1)-D^\xi_\psi(\hat{\mathbf{I}}_2))], \quad (8)$$

where D^ξ_ψ is the auxiliary pose estimator network we additionally train for the pose penalty loss jointly with the discriminator.

The proposed method simultaneously learns the output images' poses by training the pose estimator network. We modify our discriminator network to contain an auxiliary pose estimator, by adjusting the channel size of the last layer to estimate the camera pose values of the output image. Because we randomly sample camera poses $\boldsymbol{\xi}$ from the prior distribution p_ξ to generate view-consistent images, the sampled camera pose is utilized as the ground truth pose when training the pose estimator. We define the camera pose $\boldsymbol{\xi}$ with radius r_{cam}, rotation angle $\kappa_r \in [-\pi, \pi]$, and elevation angle $\kappa_e \in [0, \pi]$. Given that we use a fixed value for r_{cam}=1, the pose estimator predicts the rotation angle and elevation angle, applying the Sigmoid function to the output value multiplied by 2π and π respectively. The camera pose reconstruction loss is defined as

$$\mathcal{L}_{\text{pose}}(\psi) = \mathbb{E}_{\mathbf{z}^s,\mathbf{z}^a\sim p_z,\boldsymbol{\xi}\sim p_\xi,\mathbf{c}\sim p_r}[1 - \cos(D^\xi_\psi(\hat{\mathbf{I}}) - \boldsymbol{\xi}_{\text{gt}})], \quad (9)$$

where $D^\xi_\psi(\hat{\mathbf{I}}) = \boldsymbol{\xi}_{\text{pred}} = (\hat{\kappa_r}, \hat{\kappa_e})$. D^ξ_ψ denotes the auxiliary pose estimator and $\boldsymbol{\xi}_{\text{gt}}$ is a randomly sampled camera pose value to generate $\hat{\mathbf{I}}$. Because the angle can be represented by a periodic function, we design the pose reconstruction loss with the cosine function to penalize the angle difference, addressing its discontinuity at 2π.

3.5. Training Objective

To synthesize conditional outputs, we adopt a conditional GAN [7] by training a discriminator that learns to match images and condition feature vectors. As shown in Fig. 2, the discriminator extracts the image feature through a series of 2D convolution layers, and the image feature is then concatenated with matching condition $\mathbf{e}$ to predict the condition-image semantic consistency. The matching condition $\mathbf{e} \in \mathbb{R}^{L_c+L_s+L_a}$ is the global feature vector $\mathbf{c}$ concatenated with detail codes $\mathbf{z^s}$ and $\mathbf{z^a}$. The number of feature extracting layers is determined by the resolution of the training images. The discriminator network learns whether the given image is real or fake and matches its condition feature vector simultaneously.

At training time, we use the non-saturating GAN loss with a matching-aware gradient penalty [17, 35]. Instead of the R_1 gradient penalty [17], we adopt the matching-aware gradient penalty loss, which is known to promote the generator to synthesize more realistic and semantic-consistent images to condition-image pairs. We define three different types of data items: synthetic images with the matching condition, real images with a matching condition, and real images with a mismatching condition. The target data point on which the gradient penalty is applied can be defined by real images with the matching condition feature vector. The entire formulation of conditional GAN loss, *i.e.,*

$$\begin{aligned}
&\mathcal{L}_{\text{adv}}(\psi) = \mathbb{E}_{\mathbf{I}\sim p_r}\left[f\left(D_\psi(\mathbf{I}, \mathbf{e})\right)\right] \\
&+ (1/2)\mathbb{E}_{\mathbf{I}\sim p_{mis}}\left[f\left(-D_\psi(\mathbf{I}, \mathbf{e})\right)\right] \\
&+ (1/2)\mathbb{E}_{\boldsymbol{\xi}\sim p_\xi,\mathbf{e}\sim p_r,p_z}\left[f\left(-D_\psi\left(G_\theta(\boldsymbol{\xi}, \mathbf{c}, \mathbf{z}^s, \mathbf{z}^a), \mathbf{e}\right)\right)\right] \\
&+ k\mathbb{E}_{\mathbf{I}\sim p_r}\left[(\|\nabla_{\mathbf{I}} D_\psi(\mathbf{I}, \mathbf{e})\| + \|\nabla_{\mathbf{e}} D_\psi(\mathbf{I}, \mathbf{e})\|)^p\right], \\
&\mathcal{L}_{\text{adv}}(\theta) = \mathbb{E}_{\boldsymbol{\xi}\sim p_\xi,\mathbf{e}\sim p_r,p_z}\left[f\left(D_\psi\left(G_\theta(\boldsymbol{\xi}, \mathbf{c}, \mathbf{z}^s, \mathbf{z}^a), \mathbf{e}\right)\right)\right]
\end{aligned} \quad (10)$$

where $f(u) = -\log(1 + \exp(-u))$. p_r and p_{mis} denote the real data distribution and mismatching data distribution, respectively. k and p are two hyper-parameters that balance the gradient penalty effects.

Our full training objective functions for the generator network G_θ are summarized as

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{adv}} - \lambda_{\text{div}}\mathcal{L}_{\text{div}} + \lambda_{\text{pose}}\mathcal{L}_{\text{pose}}, \quad (11)$$

where λ_{div} and λ_{pose} are weights for each loss term.

Dataset	CelebA				Cats			
Method	Image Resolution	FID↓	Precision↑	Recall↑	Image Resolution	FID↓	Precision↑	Recall↑
GRAF	128	66.37	0.71	0.00	64	**13.73**	0.86	0.20
GIRAFFE	64	24.11	**0.88**	0.08	64	16.05	0.74	0.37
pi-GAN	128	21.38	0.72	0.45	128	22.57	0.61	0.25
Ours	64	**7.81**	0.87	**0.50**	128	13.86	**0.91**	**0.52**
	128	9.32	0.86	0.47				

Dataset	FFHQ				CUB-200			
Method	Image Resolution	FID↓	Precision↑	Recall↑	Image Resolution	FID↓	Precision↑	Recall↑
GRAF	-	-	-	-	64	41.65	0.80	0.09
StyleNeRF	256	22.054	0.501	0.470	-	-	-	-
Ours	256	**10.020**	**0.866**	**0.498**	128	**26.53**	**0.82**	**0.22**

Table 2: Quantitative comparison in terms of FID, precision, and recall. A low FID score means that the distribution of the generated image is close to that of the real image in terms of the mean and standard deviation. A high precision score implies that the generated image is realistic, and a high recall score indicates that the generated images capture greater variation of the real images.

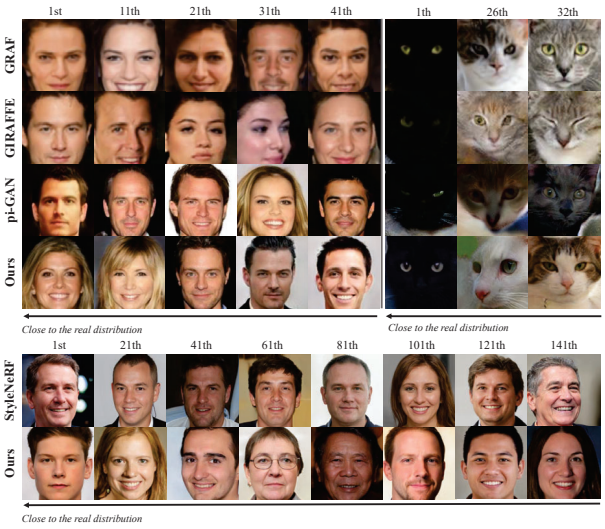


Figure 3: Comparison of qualitative results to previous studies on the CelebA, Cats, and FFHQ. For each dataset, the distance between the generated and the real image increases from left to right. To show diverse images, we sample results with different rank intervals depending on the datasets.

4. Experiments

Dataset setups We evaluate our CG-NeRF on various datasets, in this case CelebA [16], CelebA-HQ [10], FFHQ [11], CUB-200 [36], and Cats [47]. For the condition inputs, we select five different data forms to consider the different properties of input conditions in terms of the shape and appearance, *e.g.,* color images, grayscale, sketches, low-resolution images, and text. To generate 3D-aware images from sketch conditions, first we apply a Sobel filter to extract pseudo sketch information from the image [28] after which we apply a sketch simplification method [31]. For low-resolution image conditions, we apply bilinear downsampling to images with a ratio of 1/16. Training images are resized to a resolution of 128×128. To extract the global feature only of the object, we remove the background for CelebA-HQ and CUB-200 datasets.

4.1. Experimental results

To the best of our knowledge, there exists a no comparable previous work performing conditional generative NeRF task has been published. Hence, we perform quantitative and qualitative comparison of our model with existing NeRF-based generative models [30, 23, 2, 4] to demonstrate the competitive performance of the proposed method.

4.1.1 Quantitative comparison

To evaluate our approach quantitatively, we measure three metrics: the Frechet Inception Distance (FID) [6], precision, and recall using publicly available libraries[1][2] [24, 19]. FID is the most popular metric for evaluating the quality of GANs as it reveals a discrepancy between distributions of real and fake images. On the other hand, precision and recall measure the quality of GANs in terms of fidelity and diversity, respectively.

CelebA-HQ	FID↓	IS↑
Color Image	7.01	2.14
Grayscale	7.23	2.12
Sketch	7.01	2.16
Low-Resolution	7.91	2.05
Text	7.31	2.13

Cats	FID↓	IS↑
Color Image	13.86	2.06
Grayscale	12.51	2.02
Low-Resolution	19.40	2.13
CUB-200	FID↓	IS↑
Text	26.53	3.52

Table 3: Quantitative comparisons (FID / IS) on the CelebA-HQ, Cats, and CUB-200 datasets with different condition types in terms of the image quality.

As reported in Table 2, to guarantee the most reliable performance of the previous methods, we evaluate the comparison results using a publicly available pre-trained model and its corresponding experiment setting. Based on the performances we measured, the proposed method shows better scores in terms of FID, precision, and recall compared to the existing methods for the most part. For the CelebA dataset, our method still produces competitive performance on precision as well as the best performance on FID and recall.

4.1.2 Qualitative comparison

Fig. 3 shows comparisons of our method with other NeRF-based generative models in terms of the visual quality. For a fair comparison, according to the definition of precision [12], we select images in the order of the closest distance to the real image among fake images existing in the manifold of the real image. The distance is measured utilizing features of the real and fake images in the Euclidean space due to the high dimensionality of the image and lack of seman-

[1]https://github.com/toshas/torch-fidelity
[2]https://github.com/clovaai/generative-evaluation-prdc

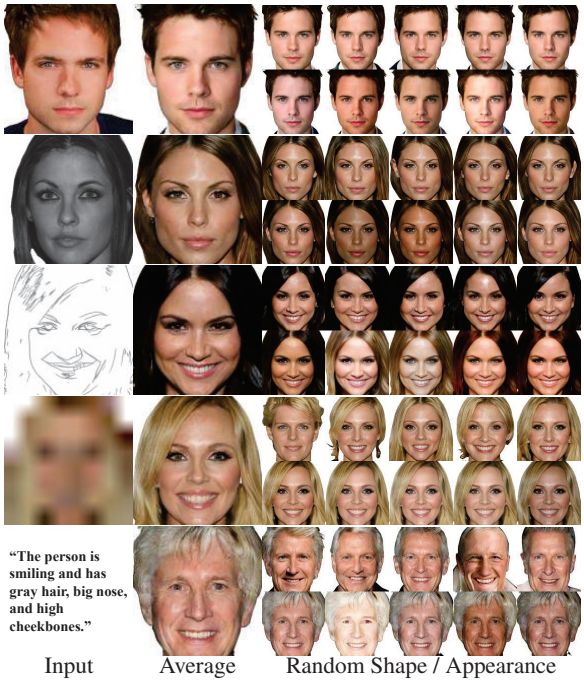


Figure 4: Qualitative results with various condition input. For each condition type, the average output image generated with zero-value noise codes and output images generated from five different shape noise codes (in row 1) and appearance noise codes (in row 2) are visualized.

tics in the RGB space. To display diverse images across all the methods, images are sampled with the distanced rank interval. Our method shows competitive visual quality regardless of datasets (Fig. 3).

4.1.3 Effects of various condition types

In this section, we perform experiments to analyze the training behavior of our method depending on the input condition type. We compare the results with five different types of condition input to validate that our method yields consistent generation performance. As shown in Fig. 4, as the color image has the largest amount of condition information among the five different condition types, it restricts the range of style variation of output images generated with random noise codes. In contrast, weak conditions such as text or low-resolution images show dynamic changes in their results with random shapes or appearances. To evaluate our approach in terms of condition types quantitatively, we measure the FID [6] and Inception Score (IS) [29] as shown in Table 3. For each dataset, our method consistently maintains high visual quality across all types of input conditions.

(a) Trained without PD loss (b) Trained with PD loss

Figure 5: Qualitative analysis of the PD loss. Along with condition inputs which are visualized with red rectangles (grayscale in row 1, sketch in row 2), Eleven output images generated with different noise codes are visualized.

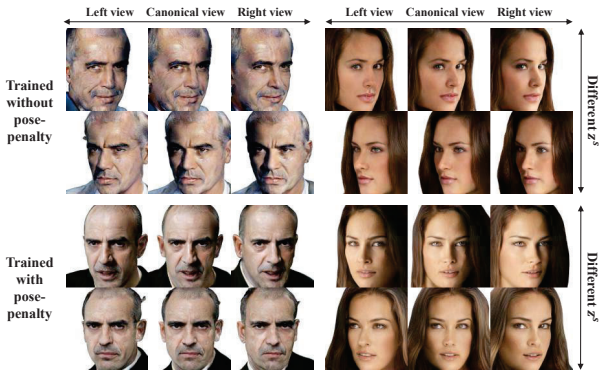
	without PDloss		with PDloss	
Condition Types	Precision↑	Recall↑	Precision↑	Recall↑
Color Image	0.899	0.520	0.900	0.550
Grayscale	0.897	0.532	0.900	0.536
Sketch	0.904	0.547	0.892	0.567
Low-Resolution	0.910	0.497	0.896	0.514
Text	0.898	0.489	0.891	0.510
Average	0.902	0.517	0.895	0.535

Table 4: Effect of the PD loss on precision and recall for measuring the fidelity and diversity, respectively, on the CelebA-HQ Dataset.

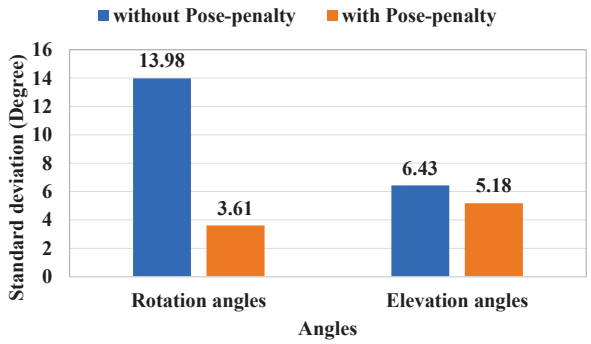
4.2. Analysis of Experiments

4.2.1 Enhanced Diversity

Because the PD loss proposed in this paper can improve the diversity of the generated images, we analyze the effect of the PD loss by taking recall and precision measurements. As shown in Table 4, as a result of applying the PD loss, the recall value is improved by about 3.5%, and the precision shows a decrease of about 0.77% on average, showing minimul degradation of visual quality. In addition, the recall is improved in all conditions; in particular, for the color and grayscale condition settings, both precision and recall are improved. From this result, applying the PD loss can increase the diversity while maintaining similar fidelity outcomes. Fig. 5 visualizes the result for a qualitative comparison of cases with and without the PD loss. The PD loss encourages the model to generate more diverse images compared to those without this loss, not only on the hair and skin color but also on the illumination.



(a) Trained without and with a pose-penalty



(b) Standard deviations of head poses.

Figure 6: Effects of the pose-penalty when attaching the diversity-sensitive loss when training. As shown in (a), for the result trained without a pose-penalty, the canonical view varies as different shape noise codes are sampled. In contrast, the result trained with a pose-penalty maintains the canonical view with different shape noise codes. (b) shows the standard deviation of head poses of randomly generated canonical view images.

4.2.2 Pose Penalty

To validate the importance of the pose-penalty in relation to the diversity-sensitive loss [43] for our method, we conduct an ablation study to confirm the effect of the pose-penalty when attaching the diversity-sensitive loss when training. As shown in Fig. 6 (a), the diversity-sensitive loss alone prevents the network from properly learning the canonical views of objects. This implies that the model maximizes the pixel-level difference causing the pose difference of the output image, which is an undesirable effect. With the PD loss, the network properly learns to maximize the style difference while maintaining the pose. For a quantitative validation, we measure the head poses of randomly generated canonical view images using the pre-trained head pose estimator [44]. As shown in Fig. 6 (b), view-consistency is maintained with a pose-penalty by a large margin compared to the result without a pose-penalty, by showing the lower standard deviation of angles of identical view images. Note that the difference in the standard deviation of the rotation angle is larger than that in the elevation angle, as the prior camera pose distribution has a broader range of the rotation angle.

Figure 7: Multi-view output images(the second row) in CUB-200 dataset. The corresponding input text is in the first row.

4.2.3 Results of CUB-200

Fig. 7 shows qualitative results on the CUB-200 dataset for text input condition. Our proposed model successfully utilizes contextual information in the given text input to generate conditional multi-view images. However, for most existing NeRF-based generative models, we empirically find that the visual quality is degraded for CUB-200 dataset in certain range of viewpoints. We suppose the performance degradation comes from large discrepancy between the prior camera pose distribution and the real one, as described in [22]. We plan to address this issue for future work.

5. Conclusion

In this paper, we propose a novel conditional generative model called CG-NeRF, which takes the existing generative NeRF to the next level. CG-NeRF creates photorealistic view-consistent images reflecting the multimodal condition inputs, such as sketches or text. Our framework also effectively extracts both the shape and appearance from the condition and generates diverse images by adding details through noise codes. In addition, we propose the PD loss to enhance the variety of generated images while maintaining view consistency. Experimental results demonstrate that our method achieves state-of-the-art performance qualitatively and quantitatively based on the quality metrics of FID, precision, and recall. In addition, the proposed method generates various images reflecting the properties of the condition types in terms of the shape and appearance.

Acknowledgement This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-00075, Artificial Intelligence Graduate School Program(KAIST)), National Research Foundation of Korea (No. 2022R1A2B5B0200191311), and the KAIST-NAVER hypercreative AI center.

References

[1] 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. Efficient geometry-aware 3D generative adversarial networks. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, 2022.

[2] 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 *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 5799–5809, 2021.

[3] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, 2020.

[4] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*, 2021.

[5] Peter Hedman, Pratul P. Srinivasan, Ben Mildenhall, Jonathan T. Barron, and Paul Debevec. Baking neural radiance fields for real-time view synthesis, 2021.

[6] 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. *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[7] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1125–1134, 2017.

[8] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. *arXiv preprint arXiv:2112.01455*, 2021.

[9] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis, 2021.

[10] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[11] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4401–4410, 2019.

[12] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *arXiv preprint arXiv:1904.06991*, 2019.

[13] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 6498–6508, 2021.

[14] David B Lindell, Julien NP Martel, and Gordon Wetzstein. Autoint: Automatic integration for fast neural volume rendering. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 14556–14565, 2021.

[15] Steven Liu, Xiuming Zhang, Zhoutong Zhang, Richard Zhang, Jun-Yan Zhu, and Bryan Russell. Editing conditional radiance fields. In *Proc. of IEEE international conference on computer vision (ICCV)*, pages 5773–5783, 2021.

[16] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proc. of IEEE international conference on computer vision (ICCV)*, December 2015.

[17] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *Proc. of the International Conference on Machine Learning (ICML)*, pages 3481–3490. PMLR, 2018.

[18] 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 *Proc. of the European Conference on Computer Vision (ECCV)*, pages 405–421. Springer, 2020.

[19] Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. 2020.

[20] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 5480–5490, 2022.

[21] Michael Niemeyer and Andreas Geiger. Campari: Camera-aware decomposed generative neural radiance fields, 2021.

[22] Michael Niemeyer and Andreas Geiger. Campari: Camera-aware decomposed generative neural radiance fields. *arXiv preprint arXiv:2103.17269*, 2021.

[23] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 11453–11464, 2021.

[24] Anton Obukhov, Maximilian Seitzer, Po-Wei Wu, Semen Zhydenko, Jonathan Kyl, and Elvis Yu-Jing Lin. High-fidelity performance metrics for generative models in pytorch, 2020. Version: 0.3.0, DOI: 10.5281/zenodo.4957738.

[25] Roy Or-El, Xuan Luo, Mengyi Shan, Eli Shechtman, Jeong Joon Park, and Ira Kemelmacher-Shlizerman. Stylesdf: High-resolution 3d-consistent image and geometry generation. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 13503–13513, 2022.

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[28] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2287–2296, 2021.

[29] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, 29:2234–2242, 2016.

[30] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *arXiv preprint arXiv:2007.02442*, 2020.

[31] Edgar Simo-Serra, Satoshi Iizuka, Kazuma Sasaki, and Hiroshi Ishikawa. Learning to simplify: fully convolutional networks for rough sketch cleanup. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016.

[32] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Proc. of the Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020.

[33] Pratul P. Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T. Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, 2021.

[34] Shih-Yang Su, Frank Yu, Michael Zollhoefer, and Helge Rhodin. A-nerf: Surface-free human 3d pose refinement via neural rendering. *arXiv preprint arXiv:2102.06199*, 2021.

[35] Ming Tao, Hao Tang, Songsong Wu, Nicu Sebe, Xiao-Yuan Jing, Fei Wu, and Bingkun Bao. Df-gan: Deep fusion generative adversarial networks for text-to-image synthesis. *arXiv preprint arXiv:2008.05865*, 2020.

[36] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[37] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. *arXiv preprint arXiv:2112.05139*, 2021.

[38] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021.

[39] Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 9421–9431, 2021.

[40] Christopher Xie, Keunhong Park, Ricardo Martin-Brualla, and Matthew Brown. Fig-nerf: Figure-ground neural radiance fields for 3d object category modelling. *arXiv preprint arXiv:2104.08418*, 2021.

[41] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.

[42] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1316–1324, 2018.

[43] Dingdong Yang, Seunghoon Hong, Yunseok Jang, Tianchen Zhao, and Honglak Lee. Diversity-sensitive conditional generative adversarial networks. *arXiv preprint arXiv:1901.09024*, 2019.

[44] Tsun-Yi Yang, Yi-Ting Chen, Yen-Yu Lin, and Yung-Yu Chuang. Fsa-net: Learning fine-grained structure aggregation for head pose estimation from a single image. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1087–1096, 2019.

[45] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proc. of IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4578–4587, 2021.

[46] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.

[47] Weiwei Zhang, Jian Sun, and Xiaoou Tang. Cat head detection-how to effectively exploit shape and texture features. In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 802–816. Springer, 2008.

[48] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. *arXiv preprint arXiv:2103.15875*, 2021.