Class-Continuous Conditional Generative Neural Radiance Field

Jiwook Kim
tom919@cau.ac.kr
Minhyeok Lee*
mlee@cau.ac.kr

School of Electrical & Electronics Engineering
Chung-Ang University
Seoul 06974, Republic of Korea

Abstract

The focus of 3D-aware image synthesis lies in preserving spatial consistency while generating high-resolution images with fine details. Recently, Neural Radiance Field (NeRF) has emerged as a powerful method for synthesizing novel views with low computational cost and exceptional performance. Although existing generative NeRF approaches have achieved significant results, they are unable to handle conditional and continuous feature manipulation during the generation process. In this work, we present a novel model, called Class-Continuous Conditional Generative NeRF (C^{3}G-NeRF), which synthesizes conditionally manipulated photorealistic 3D-consistent images by projecting conditional features onto the generator and discriminator. We evaluate the proposed C^{3}G-NeRF on three image datasets: AFHQ, CelebA, and Cars. Our model demonstrates robust 3D-consistency, fine details, ability of 360° generation, and smooth interpolation in conditional feature manipulation. For example, C^{3}G-NeRF achieves a Fréchet Inception Distance (FID) of 7.64 in 3D-aware face image synthesis with a 128^2 resolution. Furthermore, we provide FIDs and for generated 3D-aware images of each class within the datasets, showcasing the ability of C^{3}G-NeRF to synthesize class-conditional images.

1 Introduction

There have been many approaches [20, 43] for synthesizing novel images. Generative Adversarial Network (GAN) [11] has shown outstanding results in generating images by learning the distributions of datasets, resulting in the synthesis of photo-realistic instances. Furthermore, many studies have improved the ability to generate high-resolution images [5, 6, 18, 19]. Notwithstanding the advances made by existing studies, GANs still have limitations when synthesizing multiple views of a single object due to most data collections being based on two-dimensional information, which can lead to instability in 3D-consistency.

In order to address this problem, 3D-based GANs [30, 31, 44] have been studied, which make use of volume rendering methods. However, those methods require high computational power and memory in order to train effectively. In recent years, Mildenhall *et al.* [26] proposed the Neural Radiance Field (NeRF) as an alternative to conventional voxel-based volume rendering methods [10, 14, 17]. NeRF greatly reduces the complexity of computations

*Corresponding author.

and memories required compared to other approaches. Accordingly, NeRF has been extended for use in 3D-based GAN studies [4, 9, 32, 39] with excellent results and low complexities.

Nevertheless, existing NeRF-based GANs cannot control image generation with conditional labels continuously, as they do not incorporate the necessary conditional information into the generator of the GAN. Conditioning NeRF-based GANs is crucial, as many industry applications demand precise manipulation during generation, such as selecting avatar details in the emerging metaverse. GIRAFFE [32], a NeRF-based generative model, can generate 3D-aware images without extra information, like camera location and direction, which conventional NeRF requires. Although GIRAFFE achieves impressive results, particularly in 360° generation with real-world datasets compared to other state-of-the-art models like pi-GAN and Efficient Geometry-aware 3D Generative Adversarial Networks (EG3D) [3, 4], it cannot disentangle various features [2, 24, 31] needed to generate images with desired attributes.

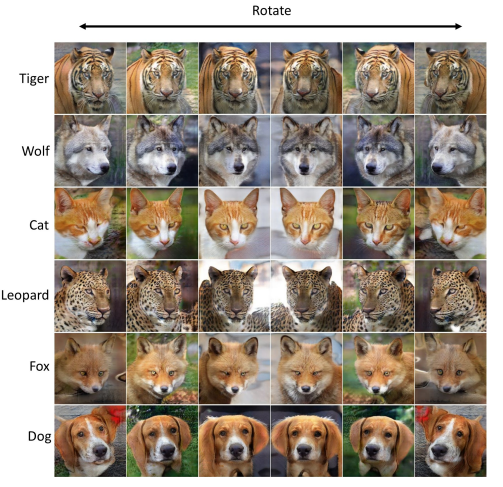


Figure 1: Synthesized images of each class of AFHQ by our model (with a 256^2 resolution). A row displays a single object with different rotation input vectors. Note that the images of different classes are generated by a single model with different conditional input vectors. Our model can generate various views of different objects that conserves strong 3D-consistency.

Jo *et al*. [16] attempted to address this issue by providing condition information such as image types and texts. However, this approach is ineffective in representing condition intensity, as texts are ambiguous compared to numerical values, which offer a more intuitive means of setting intensity. Controlling condition intensity is essential, as most metaverse users want to customize their avatars with specific eye shapes or hair colors, which can be achieved by selecting numerical values that represent the intensity of each condition.

In this paper, we propose a novel method to tackle the task of 3D-aware conditional image generation. This task requires that conditional label values [27, 28, 33] control the features of generated images, as illustrated in Figure 1, and that these values be continuous to enable smooth changes in the corresponding condition intensities. To the best of our knowledge, this paper is the first to address this task.

We introduce our proposed method, Class-Continuous Conditional Generative Neural Radiance Field (C^{3}G-NeRF), which focuses on conditional and continuous feature manipulation in 3D-aware image generation. Our backbone model is GIRAFFE, which enables 360° generation without extra camera parameters, a core task in 3D-aware generation due to its wide range of applications. Although EG3D and pi-GAN outperform GIRAFFE in generation quality, they are not suitable for 360° generation in real-world datasets. EG3D relies on extra camera parameters, making it unsuitable for learning real-world datasets without these parameters. Additionally, we validate the necessity of GIRAFFE by comparing its performance in generating a real-world car dataset [1] with pi-GAN using the Fréchet Inception Distance (FID) and Kernel Inception Distance (KID) metrics.

We observe that conditional GIRAFFE without residual modules struggles to learn data

distributions with fine details, making residual modules essential for training conditional GIRAFFE. To address this issue, we incorporate residual modules [12, 13] into our model architecture to support training and improve image synthesis.

Addressing the challenge of generating multi-view instances in 3D-aware images, C^{3}G-NeRF provides fine 3D-consistency across multiple views. We present results from three datasets, AFHQ [7], CelebA [23], and Cars [1], and demonstrate control over image synthesis through translation, rotation, and the addition of objects within a single image.

The contributions of this paper are as follows:

- We propose the C^{3}G-NeRF model to address a novel task: conditional and continuous feature manipulation in 3D-aware image generation.
- We reduce training time and enhance performance by incorporating residual modules into the NeRF architecture.
- We showcase conditional and continuous feature manipulation in 3D-aware image generation using multiple datasets: AFHQ, CelebA, and Cars.
- Since our model can generate class-conditional 3D-aware images, we provide FID scores for each label in AFHQ and Cars datasets.

2 Related Work

Implicit Neural Representation and Rendering: The use of deep learning techniques [21] to represent three-dimensional space has received considerable attention in recent years. Among the various approaches that have been proposed, implicit neural representations [36, 40, 42] have shown particular promise. NeRFs have been proposed by combining an implicit neural representation with volume rendering [10] to enable the synthesis of novel views that are not explicitly represented in the training data. As a result, NeRF is capable of synthesizing 3D-consistent images with fine details. However, one downside to using NeRFs is that they require highly constrained images for supervision during training. Another concern is that each instance of a NeRF can only represent a single object rather than multiple objects simultaneously. It has been proposed that generative NeRFs may be able to alleviate these problems.

Generative NeRF: There has been some recent progress in NeRF-based methods for generating 3D-aware images from 2D unconstrained image datasets. In these methods, generative models are trained to ensure continuous 3D geometric consistency. For instance, the GRAF [39] and pi-GAN [4] both proposed a generative NeRF, and showed promising results. GIRAFFE is another method that is more closely related to our work and improves on GRAF by separating an object from its background scene. However, none of these methods can control the conditional generation of images, which enables various feature manipulation in generated images. To address this issue, Jo *et al*. [16] proposed CG-NeRF. Specifically, CG-NeRF takes two types of conditional information. First, conditional data forms are used to translate one image into another; while our study aims to generate novel images directly from noise vectors instead. Second, CG-NeRF takes texts as conditions with CLIP (a natural language processing technique) [34]; whereas our model uses numerical conditions, which can be interpolated by varying values, allowing for continuous feature manipulation in image synthesis that represents intensity changes in relation to the given values.

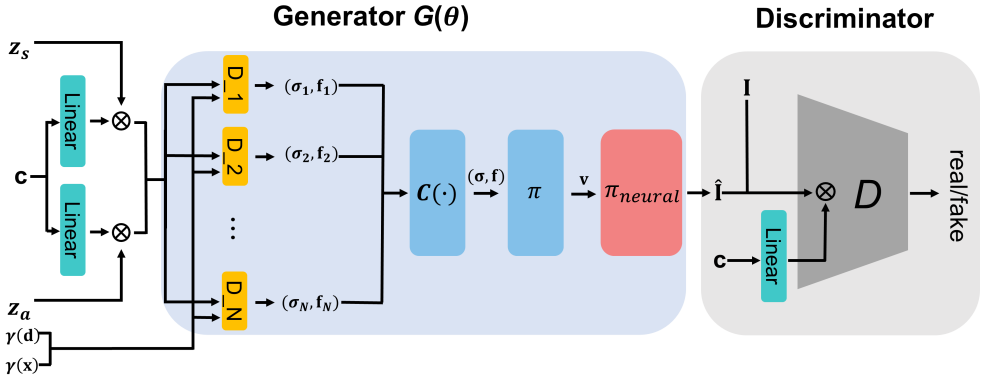


Figure 2: Overview of the proposed C^{3}G-NeRF. Since our model is inspired by the architecture of GIRAFFE, our model generates $N-1$ objects and the background with N decoders and a composition operator. D_i indicates ith decoder and $C(\cdot)$ represents the composition operator. The decoders take a 3D coordinate vectors of positional encoding $\gamma(\mathbf{x})$ and viewing direction $\gamma(\mathbf{d})$, where γ indicates positional encoding functions. In addition, the decoders take conditional vectors $\mathbf{c}$, which are encoded by linear layers, shape codes $\mathbf{z_s}$, and appearance codes $\mathbf{z_a}$. By compositing the outputs of each decoders with the composition operator $C(\cdot)$ and then volume-renders the result. Consequently, a composited feature vector $\mathbf{v}$ is produced. The feature vector $\mathbf{v}$ passes the neural rendering module π_{neural}. In this process, the generator $G(\theta)$ synthesizes a fake image $\hat{\mathbf{I}}$. The discriminator D takes a real image $\mathbf{I}$ or the fake image $\hat{\mathbf{I}}$ projected by the conditional labels $\mathbf{c}$.

3 Methods

Our goal is to make a framework for conditionally controllable 3D-aware image synthesis that guarantees representations of the intensity of conditions with continuous values. Given a labeled real-world 2D image dataset, the 3D-aware image generator G takes conditions and camera pose, which are denoted by $\mathbf{x}$ and $\mathbf{d}$, respectively, and latent vectors for representing shape and appearance, i.e., $\mathbf{z_s}$ and $\mathbf{z_a}$. Then, $\mathbf{G}$ produces an image $\hat{\mathbf{I}}$, corresponding to the input condition. At training time, real images from the dataset and $\hat{\mathbf{I}}$ are directed to the discriminator $\mathbf{D}$. Figure 2 shows an overview of our model.

3.1 Conditional Neural Radiance Fields

A conventional neural radiance field maps a 3D coordinate $\mathbf{x} = (x, y, z)$ and viewing direction $\mathbf{d} = (\theta, \phi)$ to a volume density σ and a view-dependent RGB color value R, G, and B with fully-connected layers. However, several studies [35] observed that deep learning techniques are difficult to represent high-frequency details, especially with low-dimensional inputs. To diversify the inputs, NeRFs introduced positional encoding [41] to the inputs, $\mathbf{x}$ and $\mathbf{d}$, before the fully-connected layers:

$$\gamma(p, L) = (\sin(2^0\pi p), \cos(2^0\pi p), \sin(2^1\pi p), \cos(2^1\pi p), \cdots, \sin(2^{L-1}\pi p), \cos(2^{L-1}\pi p)), \tag{1}$$

where $\gamma(\cdot)$ indicates positional encoding, L is the dimensionality of positional encoding, and p is a scalar value as a component of $\mathbf{x}$ and $\mathbf{d}$. To extend to generative neural radiance fields,

shape and appearance codes, $\mathbf{z_s}$ and $\mathbf{z_a}$, are fed into the MLP as follows:

$$(\gamma(\mathbf{x}), \gamma(\mathbf{d}), \mathbf{z_s}, \mathbf{z_a}) \mapsto (\sigma, \mathrm{R}, \mathrm{G}, \mathrm{B}). \tag{2}$$

In our model, the generative neural radiance field was substituted to generative neural feature fields by extending the dimensionality of color [32], which was originally three-dimensional, to a feature space having a dimension of M_f as:

$$\mathrm{h}_\theta : \mathbb{R}^{\mathrm{L_x}} \times \mathbb{R}^{L_\mathrm{d}} \times \mathbb{R}^{M_\mathbf{s}} \times \mathbb{R}^{M_\mathbf{a}} \mapsto \mathbb{R} \times \mathbb{R}^{M_f}, \qquad (\gamma(\mathbf{x}), \gamma(\mathbf{d}), \mathbf{z_s}, \mathbf{z_a}) \mapsto (\sigma, \mathbf{f}), \tag{3}$$

where h_θ represents a generative neural feature field, $L_\mathbf{x}$ and $L_\mathbf{d}$ indicate dimensionalities of positional encoding output, and $M_\mathbf{s}$ and $M_\mathbf{a}$ are dimensionalities of latent encodings of the shape and appearance, respectively.

In this work, we project conditions to the latent vectors [28], $\mathbf{z_s}$ and $\mathbf{z_a}$, by element-wise production. Before the projection, conditional vectors $\mathbf{c}$ having a dimension of $M_\mathbf{c}$ are encoded by a fully-connected layer to make the dimension the same as the dimensionalities of $\mathbf{z_s}$ and $\mathbf{z_a}$. The shape and appearance conditional encodings, $\mathbf{c_s}$ and $\mathbf{c_a}$, are constructed as

$$\mathbf{c_s} = L_\mathbf{s}(\mathbf{c}) * \mathbf{z_s}, \quad \mathbf{c_a} = L_\mathbf{a}(\mathbf{c}) * \mathbf{z_a}, \qquad L_\mathbf{s} : \mathbb{R}^{M_\mathbf{c}} \mapsto \mathbb{R}^{M_\mathbf{s}},\ L_\mathbf{a} : \mathbb{R}^{M_\mathbf{c}} \mapsto \mathbb{R}^{M_\mathbf{a}}, \tag{4}$$

where $L_\mathbf{s}$ and $L_\mathbf{a}$ are the encoding layers for the shape and appearance, respectively, and $*$ denotes element-wise multiplication. We employ these conditional projections as a replacement for conventional latent vectors in generative feature fields:

$$h_\theta : \mathbb{R}^{L_\mathbf{x}} \times \mathbb{R}^{L_\mathbf{d}} \times \mathbb{R}^{M_\mathbf{s}} \times \mathbb{R}^{M_\mathbf{a}} \mapsto \mathbb{R} \times \mathbb{R}^{M_f}, \qquad (\gamma(\mathbf{x}), \gamma(\mathbf{d})), \mathbf{c_s}, \mathbf{c_a} \mapsto (\sigma, \mathbf{f}). \tag{5}$$

3.2 Scene Compositions

As our model is motivated by GIRAFFE, it can separate scenes and individual objects with multiple feature vectors [32]. Consequently, in the model, there are N generative feature fields when $N-1$ objects and a background scene exist in an image. To composite N entities, the composition operator $C(\cdot)$ composites all feature fields from the N entities. Each single entity $h^i_{\theta^i}$ outputs a volume density θ_i and a feature vector $\mathbf{f}_i$. Following the method in GIRAFFE, we composite each component of entities with density-weighted mean-based composition. The mathematical expression of the composition operator can be represented as

$$C(\mathbf{x}, \mathbf{d}, \mathbf{c}) = \left(\sigma, \sum_{i=1}^{N} \frac{\sigma_i \mathbf{f}_i}{\sigma}\right), \tag{6}$$

where $\sigma = \sum_{i=1}^{N} \sigma_i$.

3.3 Volume Rendering and Neural Rendering

For scene rendering, our model volume-renders a camera ray $r(t) = \mathbf{o} + t\mathbf{d}$ to feature vectors [32], and subsequently, neural-renders the feature vectors to generate synthetic images. Our approach basically follows a discretized form of volume rendering methods used in NeRF [26]; nonetheless, detailed methods follow those of GIRAFFE to render features to images, which can be represented as

$$\mathbf{v} = \sum_{j=1}^{N_\mathbf{s}} T_j (1 - e^{-\sigma_j \delta_j}) f_j, \qquad \text{where} \quad T_j = \prod_{k=1}^{j-1} e^{-\sigma_j \theta_j}, \tag{7}$$

where $\mathbf{v}$ represents a final feature vector, T denotes an accumulated transmittance along the cast ray, and $N_\mathbf{N}$ is the number of sample points along a cast ray for an arbitrary camera pose ξ. By sampling points along the camera ray, we utilize a feature vector f_j and a density σ_j corresponding to each point. Furthermore, the feature images have a 16^2 resolution, i.e., $H_V \times W_V$ for cost-effectiveness. Existing studies, including GRAF and NeRF [26, 39], commonly adopted the volume rendering approach for 3D-to-2D projections. However, in our model, an additional 2D neural rendering network is required since the volume rendering composes feature images, not colored high-resolution images. The additional neural rendering network,

$$\pi_{neural} : \mathbb{R}^{H_V \times W_V \times M_f} \mapsto \mathbb{R}^{H \times W \times 3}, \tag{8}$$

maps outputs of the volume rendering to a synthetic image, $\hat{\mathbf{I}} \in \mathbb{R}^{H \times W \times 3}$, by upsampling the feature images. The architecture of neural rendering network is similar to a neural rendering operator in existing studies [32]; however, we hire residual modules instead of conventional convolutional networks to enhance training speed and performance.

3.4 Training Details

At training time, we randomly sample latent vectors $\mathbf{z_s}$ and $\mathbf{s_a}$, as well as camera pose ξ from prior distributions $p_\mathbf{s}$, $p_\mathbf{a}$, and p_ξ. Prior distributions $p_\mathbf{s}$ and $p_\mathbf{a}$ are defined as Gaussian distributions, and camera pose distribution ξ are set to a uniform distribution. In the generator G, we project conditional labels to latent vectors [28] $\mathbf{z_s}$ and $\mathbf{z_a}$. Similarly, conditional labels are projected to real images $\mathbf{I}$ or synthesized images $\hat{\mathbf{I}}$, in the discriminator D. Real images $\mathbf{I}$ are randomly sampled from the training dataset, which follows distribution $p_\mathbf{I}$. We use a GAN loss with R1 gradient penalty [25] as follows:

$$\begin{aligned}\mathcal{L}(G,D) =& \mathbb{E}_{\mathbf{z_s}\sim p_\mathbf{s}, \mathbf{z_a}\sim p_\mathbf{a}, \xi\sim p_\xi}\left[-\log(D(G(\mathbf{z_s},\mathbf{z_a},\xi,\mathbf{c})))\right] \\ &+ \mathbb{E}_{\mathbf{I}\sim p_\mathbf{I}}\left[-\log(1-D(\mathbf{I})) + \lambda\|\nabla D(\mathbf{I})\|^2\right].\end{aligned} \tag{9}$$

We train the generator and discriminator of the proposed model by competing for a zero-sum game with the above loss function. We use the RMSprop optimizer [38] with a learning rate of 3×10^{-4} and 1×10^{-4} for the generator and the discriminator, respectively. The model is trained on one A6000 GPU with 48GB memory with a batch size of 32. We set $M_f = 128$ for a 64^2 resolution and a 128^2 resolution, while we set $M_f = 256$ for a 256^2 resolution. We utilize ReLU activations [29] as activation functions used in our model, except for the final layer in the neural renderer, which uses a sigmoid function [37].

3.5 Accelerating Training with Residual Modules

We observe elaborate conditional generation with GIRAFFE is not feasible without residual modules [12, 25]. In the experiments of this study, we demonstrate that conditional generation with plain networks shows lagging performance. We argue that this result is because conditionally projected latent vectors are too far from the discriminator, which causes gradient vanishing. Therefore, residual modules support conveying the information to conditionally projected latent vectors from the discriminator. Moreover, the range of varying latent vectors expands since we projected conditional labels to latent vectors, which is challenging to learn with plain networks. We apply residual modules to our model in the decoder, neural rendering network, and discriminator. We validate the efficiency of this contribution in the Section 4.

(a) (b)

Figure 3: Class conditional synthetic object rotation generated by C^{3}G-NeRF trained with CelebA and Cars. In (a), each row represents a single object of CelebA with the same latent vectors. Each column indicates rotation angles. In the left figure of (a), we fixed the input conditions as a bald man, whereas we fixed the conditions as a blonde smiling woman in the right figure of (a). In (b), by controlling the horizontal and depth translation, the disentanglement of the objects and background are shown in Horizontal translation and Depth translation. After training with unstructured 2D images with a single object, we can generate $N-1$ objects in one scene by replicating N decoders as in Add objects. All images have a resolution of 128^2. Using C^{3}G-NeRF, 3D-consistent image generation is successful under the given conditions.

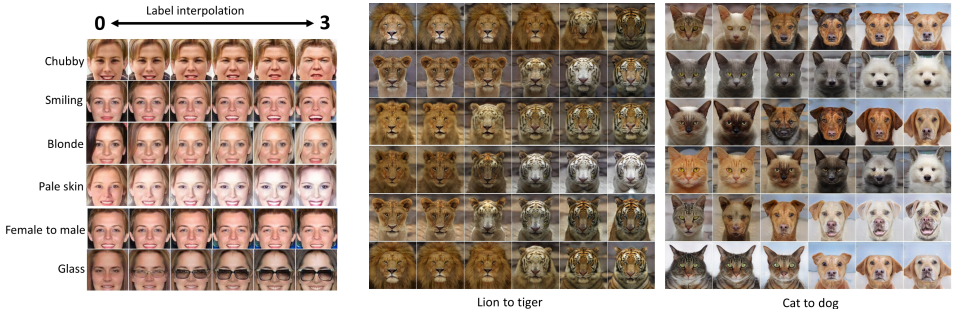
4 Experiments

In this section, we evaluate our C^{3}G-NeRF on three real-world datasets: CelebA [23], AFHQ [7], and Cars [1]. We first evaluate the conditionally controlling 3D-consistent image generations. We then evaluate the quality of generation by FIDs [15]. Finally, we include an evaluation of residual modules to validate the efficiency of residual modules adopted in our model.

4.1 Controllable Features in 3D Object Generation

The model evaluation involves performing object rotations, horizontal translations, depth translations, and object additions. The results of this evaluation are presented in Figure 1 and 3. These figures demonstrate that C^{3}G-NeRF effectively learns 3D-consistency for AFHQ, CelebA, and Cars datasets, respectively, by preserving spatial consistency despite the introduced transformations. Additionally, the model is observed to successfully capture the conditional input features and generate images accordingly.

Furthermore, in Figure 3, we assess C^{3}G-NeRF using out-of-distribution images with translating depth and horizontal at test time. This means C^{3}G-NeRF can generate beyond the distribution of training images by using extended rotation and transition values over the training sets. Moreover, we can finely control each object and scene in the generated images by C^{3}G-NeRF; for instance, while adding the objects in a scene, each object can be controlled by translating and rotating in 3D space.



(a) (b)

Figure 4: Interpolation and extrapolation on conditional input values with CelebA and AFHQ. Each row and column represent the same latent vectors (identical object) of AFHQ and the same class-conditional values, respectively. In (a), we present the conditional results according to conditional input values with the range of zero to three. Note that, in training time, the features are trained only with the two values of zero and one, which indicate the existence of the corresponding feature. The features in the face images with interpolated and extrapolated input values smoothly change, which means the continuous conditional learning is adequately progressed. By interpolating the values of each class, features of each category coexist at the intermediate state of class-conditional values.

4.2 Continuously Controllable Features in 3D Object Generation

We demonstrate our model's ability to manipulate individual conditions by interpolating 40 conditional binary values in the CelebA dataset, such as chubby, smiling, blonde, and pale skin. Although the training procedure has a range of zero to one due to binary encoding in the CelebA dataset, we expand the test procedure range to zero to three, as illustrated in Figure 4. This experiment assesses whether each facial feature maps to the generator's input label. With various conditional label values, C^{3}G-NeRF exhibits superior performance in interpolation and extrapolation for each condition. For instance, chubby and smiling conditions [23] change smoothly according to conditional values, regardless of the extrapolation range. This result highlights our model's capacity to generate out-of-distribution images, such as exaggerated features with high input label values exceeding one.

We also examine non-characteristic (implicit) labels corresponding to AFHQ classes, which are more challenging to learn due to less obvious image features. We generate inter-class images with interpolated class-conditional input values, as depicted in Figure 4. We observe that features of each class coexist at intermediate class-conditional values, indicating that feature manipulation can be applied to implicit class labels.

4.3 Quantitative Evaluation

We evaluate image quality using a conventional method with KIDs and FIDs, comparing 20,000 randomly sampled real and generated images. Assessments demonstrate generation quality for each dataset and label, evaluating images generated with random conditional inputs across different resolutions and datasets.

Table 1 shows that C^{3}G-NeRF achieves impressive FIDs and KIDs in conditional 3D-consistent generation, outperforming conditional GIRAFFE regardless of resolution and indicating robustness in high resolutions. The conditional GIRAFFE exhibits high FIDs, suggesting training issues. For instance, in CelebA face image generation at 128^2 resolution,

Model	AFHQ (FID)			CelebA(FID)		Cars(FID)		
	64^2	128^2	256^2	64^2	128^2	64^2	128^2	256^2
Baseline	212.74	226.51	239.01	55.90	83.89	228.66	244.55	266.51
Ours	**26.72**	**25.79**	**28.58**	**5.60**	**7.64**	**44.79**	**43.29**	**31.63**
Model	AFHQ (KID)			CelebA(KID)		Cars(KID)		
	64^2	128^2	256^2	64^2	128^2	64^2	128^2	256^2
Baseline	0.200	0.352	0.323	0.119	0.296	0.096	0.166	0.303
Ours	**0.077**	**0.059**	**0.051**	**0.046**	**0.023**	**0.057**	**0.039**	**0.043**

Table 1: Quantitative comparison with FIDs (↓) and KIDs (↓) with three datasets. The baseline is set to the conditional GIRAFFE with plain networks. The 64^2, 128^2, and 256^2 are the image resolutions of the generated images and real images.

Category	CARS			Category	AFHQ		
	64^2	128^2	256^2		64^2	128^2	256^2
Peykan	73.54	67.47	68.57	cat	10.38	13.65	15.48
Quik	58.73	62.71	49.64	cat	10.38	13.65	15.48
Samand	54.63	50.53	61.01	dog	31.64	43.37	51.73
Peugeot-Pars	65.34	66.27	46.45	leopard	17.20	14.98	13.42
Peugeot-207i	71.17	71.06	52.99	fox	25.97	28.01	22.50
Pride-111	67.91	62.28	68.84	lion	8.76	12.20	7.61
Pride-131	68.70	57.70	65.43	tiger	12.78	8.96	5.93
Tiba2	60.80	61.79	55.57	wolf	28.39	30.82	14.85
Renault-L90	67.79	65.51	76.84				
Nissan-Zamiad	78.17	88.83	133.77				
Peugeot-206	63.53	61.55	128.10				
Peugeot-405	71.79	66.15	71.45				
Mazda-2000	71.75	77.23	104.53				

Table 2: Quantitative comparison with FIDs (↓) with each class of AFHQ and Cars. The 64^2, 128^2, and 256^2 are the image resolutions of the generated images and real images.

C^{3}G-NeRF attains an FID of 7.64, reducing the value by 90.9% compared to the baseline.

Table 2 presents FIDs for each label, reflecting the conditional disentanglement achieved by C^{3}G-NeRF. FIDs for individual labels are similar to or better than those for all labels combined, implying that C^{3}G-NeRF effectively learns features for each condition across datasets. Moreover, even with few examples (e.g., foxes, lions, tigers in AFHQ), comparable FIDs indicate that C^{3}G-NeRF excels at learning from data, irrespective of its unposed distribution.

Model	Cars (KID)		Cars (FID)	
	64^2	128^2	64^2	128^2
pi-GAN	0.105	0.083	137.34	104.51
Ours	**0.057**	**0.039**	**44.79**	**43.29**

Table 3: Quantitative comparison with FIDs (↓) and KIDs (↓) on Cars dataset for 360° generation at resolutions 64^2 and 128^2, with comparison to pi-GAN.

Table 3 shows that our model outperforms pi-GAN on Cars dataset, effectively disentangling class conditions. Generally, conditional models underperform unconditional ones, rendering pi-GAN unsuitable for 360° generation. Additionally, EG3D struggles with Cars dataset as it needs camera positions corresponding to samples while EG3D can obtain camera positions using an independent algorithm for facial datasts [8, 22]. Thus, we adopt GIRAFFE for the generality in 360° generation with real-world datasets.

4.4 Evaluation of Residual Modules

We assess image quality generated by C^{3}G-NeRFs with residual modules, comparing it to conditional GIRAFFE with a plain network. This experiment emphasizes the impact of incorporating residual modules [12, 25] in the C^{3}G-NeRF architecture. FID and KID comparisons across three datasets (Table 1) show that C^{3}G-NeRF significantly outperforms conditional GIRAFFE with plain networks, supporting our hypothesis that residual modules enhance generation quality and are crucial for conditional 3D-aware generation. Residual modules aid gradient flow from the discriminator's output to the conditional inputs of the generator, facilitating conditional information learning. Qualitative comparisons are shown with the Figure 12, which is included in the Supplementary Materials.

5 Conclusions

We presented a novel model, C^{3}G-NeRF, for conditional and continuous feature manipulation in 3D-aware image generation. Our approach projects conditional labels with encoding layers onto the generator's latent vectors and an intermediate discriminator layer to disentangle dataset features. C^{3}G-NeRF leverages residual modules to optimize 3D-aware conditional training. Interpolating and extrapolating conditional input values, we achieved precise 3D-consistent image generation and feature manipulation.

6 Acknowledgement

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2023-00251528).

References

[1] Yousef Ashrafi. Iran cars, Aug 2022. URL https://www.kaggle.com/datasets/usefashrfi/iran-used-cars-dataset.

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[3] 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 *arXiv*, 2021.

[4] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5799–5809, 2021.

[5] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative

adversarial nets. *Advances in Neural Information Processing Systems*, 29:2172–2180, 2016.

[6] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer vision and Pattern Recognition*, pages 8789–8797, 2018.

[7] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

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

[9] Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. Gram: Generative radiance manifolds for 3d-aware image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10673–10683, 2022.

[10] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM Siggraph Computer Graphics*, 22(4):65–74, 1988.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27:2672–2680, 2014.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer, 2016.

[14] Philipp Henzler, Niloy J Mitra, and Tobias Ritschel. Escaping plato's cave: 3d shape from adversarial rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9984–9993, 2019.

[15] 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. *Advances in Neural Information Processing Systems*, 30:6626–6637, 2017.

[16] Kyungmin Jo, Gyumin Shim, Sanghun Jung, Soyoung Yang, and Jaegul Choo. Cg-nerf: Conditional generative neural radiance fields. *arXiv preprint arXiv:2112.03517*, 2021.

[17] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH Computer Graphics*, 18(3):165–174, 1984.

[18] 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.

[19] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and pattern Recognition*, pages 8110–8119, 2020.

[20] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[21] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553): 436–444, 2015.

[22] Taehee Brad Lee. Cat_hipsterizer, Oct 2018. URL https://github.com/kairess/cat_hipsterizer.

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

[24] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pages 4114–4124. PMLR, 2019.

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

[26] 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, 2021.

[27] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[28] Takeru Miyato and Masanori Koyama. cgans with projection discriminator. *arXiv preprint arXiv:1802.05637*, 2018.

[29] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *International Conference on Machine Learning*, pages 807–814, 2010.

[30] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7588–7597, 2019.

[31] Thu H Nguyen-Phuoc, Christian Richardt, Long Mai, Yongliang Yang, and Niloy Mitra. Blockgan: Learning 3d object-aware scene representations from unlabelled images. *Advances in Neural Information Processing Systems*, 33:6767–6778, 2020.

[32] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11453–11464, 2021.

[33] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International Conference on Machine Learning*, pages 2642–2651. PMLR, 2017.

[34] 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. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[35] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019.

[36] Sai Rajeswar, Fahim Mannan, Florian Golemo, Jérôme Parent-Lévesque, David Vazquez, Derek Nowrouzezahrai, and Aaron Courville. Pix2shape: Towards unsupervised learning of 3d scenes from images using a view-based representation. *International Journal of Computer Vision*, 128(10):2478–2493, 2020.

[37] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[38] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[39] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems*, 33:20154–20166, 2020.

[40] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32:1121–1132, 2019.

[41] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020.

[42] Shubham Tulsiani, Nilesh Kulkarni, and Abhinav Gupta. Implicit mesh reconstruction from unannotated image collections. *arXiv preprint arXiv:2007.08504*, 2020.

[43] Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International Conference on Machine Learning*, pages 1747–1756. PMLR, 2016.

[44] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in Neural Information Processing Systems*, 29:82–90, 2016.