BallGAN: 3D-aware Image Synthesis with a Spherical Background

Minjung Shin[1*] Yunji Seo[1] Jeongmin Bae[1] Young Sun Choi[1]
Hyunsu Kim[2] Hyeran Byun[1] Youngjung Uh[1†]
Yonsei University[1] NAVER AI Lab[2]

(a) Motivation (b) Results on FFHQ/ AFHQv2-Cats (c) Application

Figure 1: (a) 3D space (top) and an image rendered from the white camera (bottom). We are inspired by a 3D graphics technique in which the foreground is represented as a 3D model and the background is approximated as a 2D surface, yet resulting in a realistic appearance on the rendered image. (b) Our method produces high-quality 3D shapes, images, and foreground alpha masks without extra supervision. (c) Realistic novel view rendering on arbitrary backgrounds, even on real image inversion.

Abstract

3D-aware GANs aim to synthesize realistic 3D scenes that can be rendered in arbitrary camera viewpoints, generating high-quality images with well-defined geometry. As 3D content creation becomes more popular, the ability to generate foreground objects separately from the background has become a crucial property. Existing methods have been developed regarding overall image quality, but they can not generate foreground objects only and often show degraded 3D geometry. In this work, we propose to represent the background as a spherical surface for multiple reasons inspired by computer graphics. Our method naturally provides foreground-only 3D synthesis facilitating easier 3D content creation. Furthermore, it improves the foreground geometry of 3D-aware GANs and the training stability on datasets with complex backgrounds. Project page: https://minjung-s.github.io/ballgan/

*Part of the work was done during an internship at NAVER AI Lab.
†Corresponding author

1. Introduction

Traditional generative adversarial networks (GANs) synthesize realistic images. Although they provide some control over the camera poses [36, 37, 15, 38], they lack explicit 3D understanding of the scenes. Recently, 3D-aware GANs [27, 6, 35, 53] reformulate the generative procedure as modeling the potential 3D scenes and rendering them to images. The state-of-the-art 3D-aware GANs [5, 14, 47] rely on neural radiance fields or their variants to represent 3D scenes. Note that they can generate 3D scenes even without 3D supervision or multi-view supervision, rendering realistic images across different viewpoints. Although the quality of images generated by 3D-aware GANs continues to improve, their practical usage has been less explored.

Solely generating foreground objects is an important element for the practical use of generative models, especially for content creation. In this context, the diffusion-based methods have grown popular for 3D object synthesis despite their lack of realism [18, 32, 24, 39, 44]. Some

2D GANs model their output images as a combination of foreground and background, replacing the need for laborious post-processing [1, 4, 54]. On the other hand, few 3D-aware GANs inadequately separate the background and suffer from broken 3D shapes [47] or training instability [14]. Objects generated by EG3D [5] are connected to unrealistic walls as shown in Figure 2.

Learning to synthesize 3D foreground objects using a single-view dataset is challenging because it lacks both depth and separation supervision.

To solve this problem, we are inspired by a popular approach for video games or movies in the graphics community: representing salient objects with detailed 3D models and approximating peripheral scenery with simple surfaces (Figure 1a) to reduce the overall complexity. Despite approximating the 3D space to 2D, the rendered image achieves a realistic appearance. We expect the 3D-aware generators with a similar approach to achieve both separation and physically reasonable foreground geometry.

Accordingly, we propose our novel 3D-aware GAN framework, named BallGAN. It approximates the background as a 2D *opaque surface of a sphere* and employs conventional 3D features as the foreground. It accompanies a modified volume rendering equation for the opaque background. In addition, we introduce regularizers for clear foreground geometry and separation.

We demonstrate the strength of our work as follows. By design, BallGAN provides clear foreground-background separation without extra supervision (Figure 1b). For content creation, it enables inserting generated 3D foregrounds in arbitrary viewpoints without post-processing (Figure 1c). Our background representation as a spherical surface is generally applicable to any generator architectures or foreground representations. BallGAN allows StyleNeRF [14] to be trained on a higher resolution of CompCars[48][1] and achieve a large FID boost, which is notable as the dataset is challenging due to its complex backgrounds. More importantly, BallGAN not only enhances multi-view consistency, pose accuracy, and depth reconstruction compared to EG3D, but it also faithfully captures fine details in 3D space that are easy to represent in 2D images but challenging to model in 3D.

2. Related work

Representations for 3D-aware GANs Generators in 3D-aware GANs involve representing 3D scenes somehow and rendering them to 2D images so that the generator is aware of the 3D scene given only a collection of unstructured 2D images. HoloGAN [27] represents a scene with a 3D grid of voxels containing feature vectors, *i.e.*, 4D tensor. However, as the 3D grid of voxels is limited by computational complexity, its maximum resolution is 128^2.

[1] StyleNeRF diverges on CompCars while growing from 128^2 to 256^2.

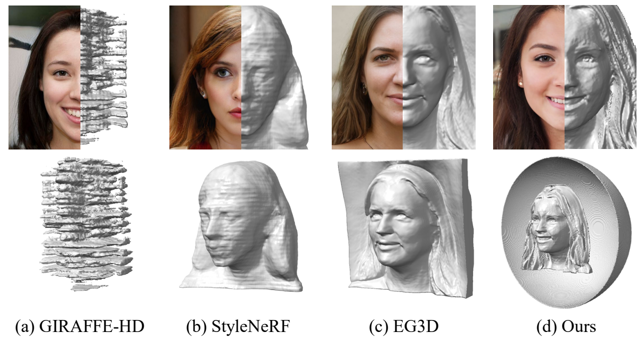
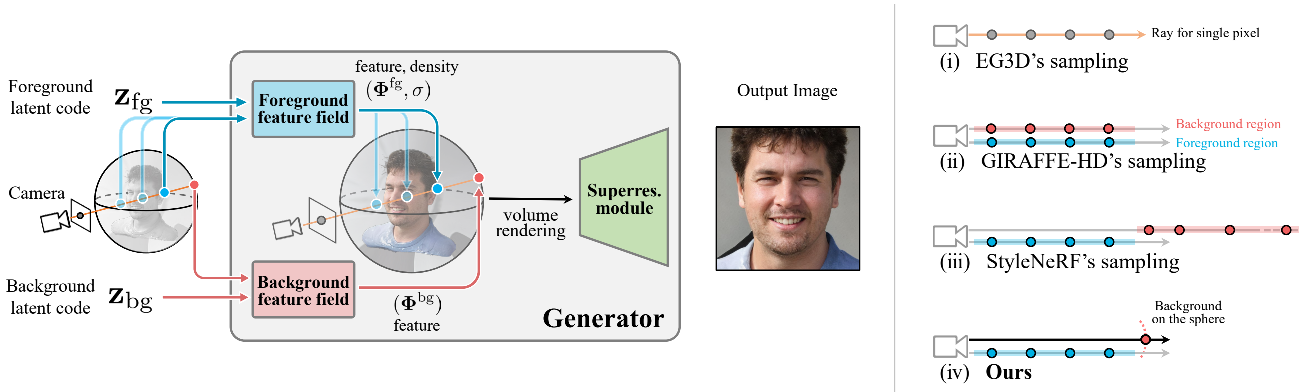


Figure 2: **Comparison of the 3D geometry extracted by marching cubes.** (a) GIRAFFE-HD exhibits broken 3D shapes, (b) StyleNeRF has jaggy surfaces, and (c) EG3D has hair sticking to the wall. Unlike other models, (d) our model produces high-quality foreground geometry that is separated from the background.

Recent 3D-aware GANs integrate neural radiance fields (NeRFs) [26]. NeRF represents a 3D scene using a coordinate-based function that produces RGB color and density at that coordinates. This 3D scene can be projected onto a 2D image from arbitrary camera poses via volume rendering integral. GRAF [35] introduces a patch-based discriminator, which dramatically reduces memory usage in high-resolution 3D-aware image synthesis. Its successors improve image quality and 3D awareness by 1) enhancing the function for NeRF [6, 14], 2) volume rendering feature field followed by neural rendering with upsampling blocks [14, 29, 45, 5, 47], or 3) designing voxel-based [43, 12, 16, 28, 45]or hybrid [5] representations. Going further, our method introduces a separate NeRF for modeling spherical background, which encloses the foreground of EG3D [5] or StyleNeRF [14].

Scene decomposition Some methods decompose the 3D scenes into multiple components. GIRAFFE and its variant [29, 47] separate scenes into objects and the background, enabling them to control objects independently with the background fixed. However, their background representation lives in the same ray points with the foregrounds, and the 3D geometry does not benefit from the separation. StyleNeRF [14] and EpiGRAF [40] separate the background outside a sphere following NeRF++ [50] where the background region goes through the same volume rendering with multiple ray points at variable depth. On the contrary, we remove the depth ambiguity of the background by modeling it with an opaque representation on a 2D spherical surface enclosing the foreground.

Reducing dimensions has been a viable option for reducing space and time complexity. TensoRF [7] uses a sum of vector-matrix outer products to represent a 3D feature field.



(a) BallGAN Generator Overview (b) Comparison of ray points

Figure 3: **Overview of the BallGAN generator and definition of ray points.** We bound the 3D space with an opaque background on a spherical surface. (i) EG3D does not separate the background. (ii) GIRAFFE-HD samples the background points within the same range of the foreground. (iii) StyleNeRF samples multiple background points outside the boundary. (iv) We sample a single background point on the sphere. It drastically reduces the depth ambiguity in the background.

EG3D [5] represents a 3D feature field with three 2D planes to adopt StyleGAN architecture. K-Planes [11] represents a d-dimensional scene using $\binom{b}{2}$ planes. While these methods decompose 3D feature fields into low-dimensional feature representations to reduce the memory usage of NeRFs, BallGAN squeezes the background space into a surface to provide an easier task for 3D-aware GANs.

3. BallGAN

In this section, we provide an overview of our framework and describe its key components and intuitions.

Overview We suppose that generating unbounded 3D scenes is too complex to learn relying on a limited guide for producing realistic 2D images. To resolve this challenge, BallGAN bounds the scene in a ball and approximates the background as an opaque spherical surface. We expect it to alleviate the burden of producing correct shapes of the backgrounds because the shape is fixed on a ball.

As shown in Figure 3, our generator consists of two backbone networks for foreground and background (§3.1). Representations from these networks are rendered by our modified volume rendering equation to synthesize images (§3.2) and trained with GAN objectives and auxiliary regularizations (§3.3).

3.1. Bounding the 3D space

While traditional 2D GANs learn to produce arrays of RGB pixels in fixed dimensions, 3D-aware GANs aim to produce realistic images by synthesizing 3D scenes and rendering them into 2D images. In contrast to training NeRFs with multi-view observations of a single scene, the only objective for the 3D-aware GANs is producing realistic 2D images. In other words, the datasets and the objective functions do not provide any clues for the 3D geometry. To reformulate 3D-aware generation as an easier constrained problem, we approximate the backgrounds on an opaque spherical surface.

Background model We model the background as a neural feature field defined on a sphere with a fixed radius. Given a ray $\mathbf{r} = \mathbf{o} + t\mathbf{d}$ (t is the distance from the camera center $\mathbf{o}$), we find the 3D background point on the sphere with radius R_{bg} by simply computing the ray's intersection on the sphere surface:

$$\mathbf{x}^{\text{bg}} = \mathbf{o} + \frac{-2[\mathbf{d}\cdot\mathbf{o}] + \sqrt{(2[\mathbf{d}\cdot\mathbf{o}])^2 - 4\|\mathbf{d}\|^2(\|\mathbf{o}\|^2 - R_{\text{bg}}^2)}}{2\|\mathbf{d}\|^2}\mathbf{d} \tag{1}$$

Since the background points are on a sphere surface of fixed radius R_{bg}, we further reparameterize the 3D coordinates $\mathbf{x}$ as 2D spherical coordinates $\mathbf{s} = (\theta, \phi)$ to further reduce the complexity.

Then we represent the feature field F_{bg} using a StyleGAN2-like architecture :

$$F_{\text{bg}}(\mathbf{s}, \mathbf{z}_{\text{bg}}) = \mathbf{g}^n_{\mathbf{w}} \circ ...\mathbf{g}^1_{\mathbf{w}} \circ \zeta(\mathbf{s}), \tag{2}$$

where $\mathbf{w} = \mathbf{f}(\mathbf{z}_{\text{bg}})$ is the style vector produced by a mapping network $\mathbf{f}$ given a noise vector $\mathbf{z}_{\text{bg}}$, and ζ is the positional encoding [42] of $\mathbf{s}$, and $\mathbf{g}_{\mathbf{w}}$ denotes 1×1 convolutions whose weights are modulated by $\mathbf{w}$. Note that there is no mapping for density from the background feature field because our background is an opaque surface.

Our background representation drastically reduces the number of points to be fed to the model, *i.e.*, only one intersection of our sphere background and the ray $\mathbf{r}$. Therefore, we do not use hierarchical sampling for the background.

Figure 3b visualizes the difference in space for each method with ray points. GIRAFFE-HD does not separate the background coordinate space from the foreground, StyleNeRF keeps multiple point candidates for the unbounded continuous depth. On the other hand, our method separates the foreground and background and bounds the background to lie on the surface. This effectively constrains the solution space and improves training stability and output quality.

Design choice for background One may wonder why we chose the sphere among many alternatives. First, the background should enclose the scene entirely to cover all viewing directions. Thus, an open plane is not available in wide-angle scenes. Second, the background should be identical when observed from all directions to make it easier for the generator to perform consistently well. Therefore, the spherical surface is the only reasonable choice. Appendix A provides empirical comparison.

Foreground model We adopt StyleNeRF [14] or EG3D [5] for foreground modeling, where a random foreground code $\mathbf{z}_{\text{fg}}$ is fed to StyleGAN2 [22] network to produce implicit or hybrid representation, respectively. Formally:

$$(\mathbf{\Phi}^{\text{fg}}, \sigma) = F_{\text{fg}}(\mathbf{x}, \mathbf{z}_{\text{fg}}). \tag{3}$$

Note that our simple and effective background modeling is applicable to arbitrary 3D scene representations other than StyleNeRF and EG3D.

3.2. Volume rendering

Volume rendering aggregates the neural feature field along the rays through individual pixels to produce feature maps for a given camera pose. The conventional volume rendering computes the contribution of all points $\{\mathbf{x}_i\}$ sampled on a ray using the same equation $T(\mathbf{x}_i)(1 - \exp(-\sigma(\mathbf{x}_i)\delta(\mathbf{x}_i)))$, where T denotes transmittance, σ denotes density.

We modify the volume rendering equation to reflect our background design, a single point with full density:

$$\phi(\mathbf{r}) = \sum_{i=1}^{\mathbf{N}_{\text{fg}}} T_i(1 - \exp(-\sigma_i\delta_i))\mathbf{\Phi}_i^{\text{fg}} + T^{\text{bg}}\mathbf{\Phi}^{\text{bg}}, \tag{4}$$

where $\phi(\mathbf{r})$ is an aggregated pixel feature along the ray $\mathbf{r}$, $T_i = \exp(\sum_{j=1}^{i-1} -\sigma_j\delta_j))$ denotes accumulated transmittance at i-th point $\mathbf{x}_i$, $\mathbf{\Phi}_i$ and σ_i are the feature and the density at $\mathbf{x}_i$, and $\delta_i = t_{i+1} - t_i$ denotes the distance between adjacent points. Since the background point is considered opaque and proceeded by all foreground points, we define its contribution using only the transmittance $T^{\text{bg}} = \exp(\sum_{j=1}^{\mathbf{N}_{\text{fg}}} -\sigma_j\delta_j))$. It is equivalent to placing an opaque background behind the scene in computer graphics techniques.

To synthesize high-resolution images, we employ a 2D-CNN-based super-resolution module to upsample and refine the feature maps to an RGB image as commonly done in recent methods [29, 47, 14, 5].

3.3. Training objectives

We use the non-saturating GAN loss $\mathcal{L}_{\text{adv}}$ [13] and R1 regularization $\mathcal{L}_{\text{R}_1}$ [25]. Additionally, we use two regularizations.

Background transmittance loss To ensure clear separation between foreground and background, we introduce new regularization on T^{bg}. The ray through the foreground region in the image should have a high foreground density that makes T^{bg} close to 0, and thus the background feature should not affect the aggregated pixel. In contrast, foreground density should be small enough to make T^{bg} close to 1 when the ray corresponds to the background, so the aggregated pixel feature should be the same as the background feature. Therefore, we induce the transmittance of the background to be binarized:

$$\mathcal{L}_{\text{bg}} = \sum \min(T^{\text{bg}}, 1 - T^{\text{bg}}). \tag{5}$$

Foreground density loss To encourage clear shape, we use foreground regularization to prevent foreground density from diffusing. Similar to Mip-NeRF 360[3], our foreground loss penalizes the entropy of the aggregation weights on the ray to locate foreground points in the area where the actual geometry is located:

$$\mathcal{L}_{\text{fg}} = \sum_r \left(\sum_{i,j} \mathbf{w}_i^r \mathbf{w}_j^r |t_i^r - t_j^r| + \frac{1}{3}\sum_i \mathbf{w}_i^{r\,2}\delta_i^r \right), \tag{6}$$

where i and j are the indices of the weight, r is the index of the ray, $\delta_i = t_{i+1} - t_i$ is the distance between adjacent points and $\mathbf{w}$ is the aggregation weights after sigmoid function. This regularization is the integral of the weighted distance between all pairs of points on each ray.

The total loss function is then

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{adv}} + \lambda_{\text{R}_1}\mathcal{L}_{\text{R}_1} + \lambda_{\text{fg}}\mathcal{L}_{\text{fg}} + \lambda_{\text{bg}}\mathcal{L}_{\text{bg}}, \tag{7}$$

where λ_{R_1}, λ_{fg} and λ_{bg} are hyperparameters.

4. Experiments

In this section, we evaluate the effectiveness of BallGAN compared to the baselines regarding the faithfulness of foreground-background separation in §4.1, effectiveness on complex backgrounds in §4.2, the faithfulness of underlying 3D geometry in §4.3, and image quality in §4.4. Implementation details are in Appendix D.

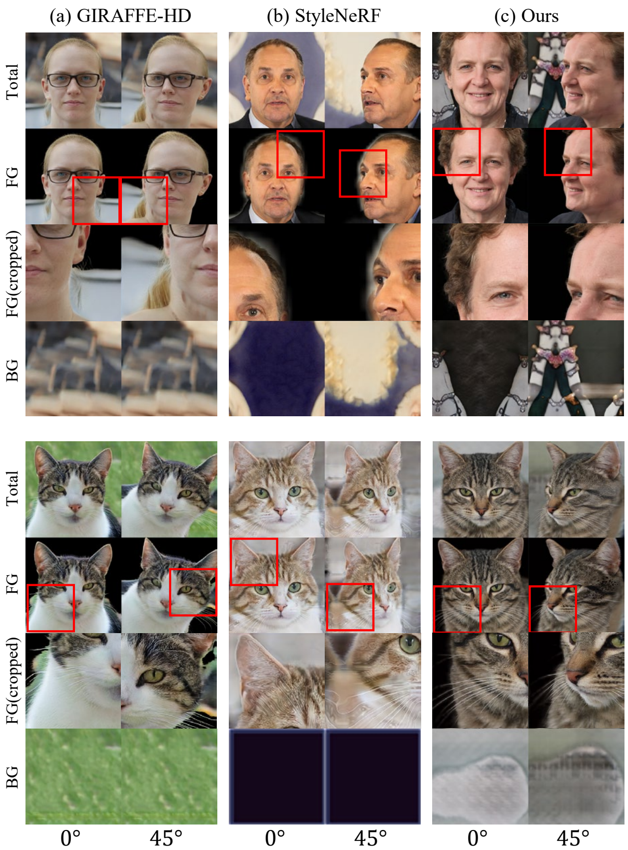


Figure 4: **Separate renderings of the foreground and background.** For easy comparison, we also show cropped foreground images.

Datasets We validate our method on two front-facing datasets, FFHQ [21] and AFHQv2-Cats [8, 20], and one 360° dataset, CompCars [48]. FFHQ has 70K images of real human faces, and AFHQv2-Cats contains 5,558 images of cat faces. We resize the resolutions of these datasets to 512^2. CompCars contains 136K images of cars with various resolutions and aspect ratios. In CompCars, we use a center cropping for each image and resize it to 256^2.

Competitors For our main comparisons we use EG3D [5], StyleNeRF [14] and GIRAFFE-HD [47]. We include EpiGRAFF [40][1], MVCGAN [52], VolumeGAN [46] and StyleSDF [30] for quantitative comparisons.

[1]By incorporating NeRF++'s inverse sphere parameterization, EpiGRAF can separate foreground and background, same as StyleNeRF. However, the reported performance in the paper is based on a setting without the utilization of background representation. The official repository indicates a performance drop of approximately 10% to 15% when background representation is employed. Therefore, we employ the official version of EpiGRAF that doesn't use the background representation as a competitor. Refer to the Appendix G for a detailed ablation study using EpiGRAF, which adopts NeRF++ as the background representation.

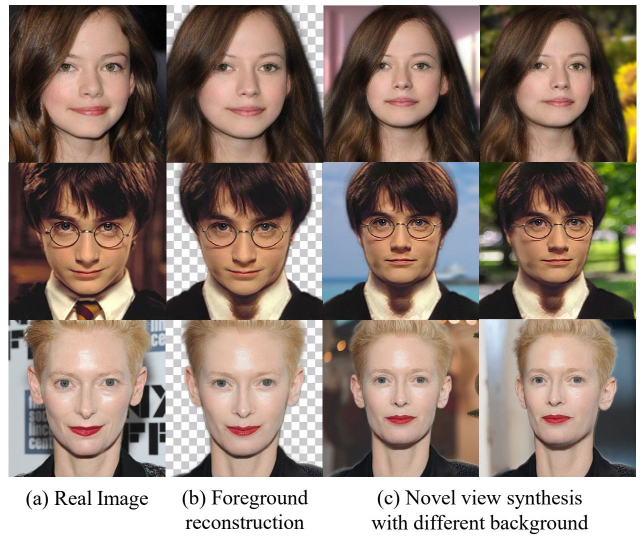


Figure 5: **Compositing foreground in different viewpoints on arbitrary backgrounds.** (a) is a target image, and (b) is a reconstructed foreground of ours using PTI) [33]. (c) is a result of novel views on arbitrary backgrounds. By changing the camera pose and FOV, we show that our model can generate attributes of unobserved regions well.

4.1. Foreground separation

To achieve reasonable 3D perception and applicability, accurately separating foreground and background is an important evaluation factor. As the background on a spherical surface is one of the key components of our method, we evaluate the separability and geometry of foregrounds against GIRAFFE-HD and StyleNeRF. EG3D is excluded because it does not provide separation.

Comparison Figure 4 shows rendered images of foreground and background, respectively. GIRAFFE-HD uses an alpha mask for detailed foreground separation, but it relies on 2D feature maps instead of understanding the 3D scene. Therefore, the foreground partly includes the background. StyleNeRF shows some ability to separate the foreground on FFHQ, but fails to do so for all cases of AFHQ-cats, which contain a significant amount of fine-grained details. By contrast, our results demonstrate fine-grained foreground separation, including intricate details like cat whiskers. Please refer to Appendix E for quantitative evaluation (User study).

Content creation Figure 5 demonstrates the content creation capabilities achievable with BallGAN. Given a real image, its inversion on BallGAN provides 3D foreground that can be rendered in novel views and combined with different backgrounds. The alpha channel for the background is computed from the background transmittance in the volume rendering step, i.e., the last term in (4). Even the facial

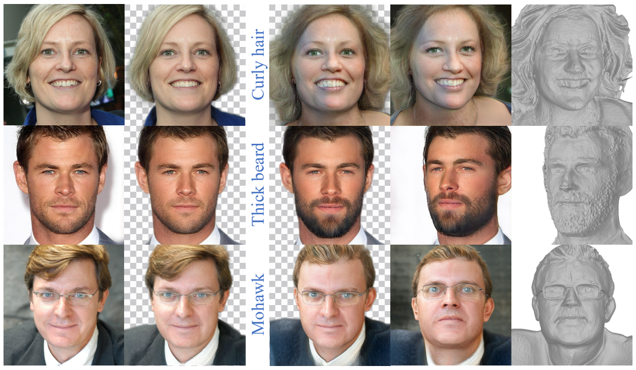


Figure 6: **CLIP guided editing results.** Given text prompt is blue.

regions that are not seen in the original images are realistic in the rendered images, such as parts of hair or chin. Note that Figure 5 has a wider field-of-view than the standard to produce more diverse results.

Figure 6 demonstrates the potential of BallGAN to 3D content creation. We can synthesize novel views of the edited foregrounds by inverting images to the latent space and using text-guided latent editing [31]. Note that the 3D shapes are properly changed by the editing, e.g., hair. Therefore, BallGAN is useful for 3D content creation thanks to its foreground-background separation.

4.2. Effectiveness on complex backgrounds

Here, we demonstrate the effectiveness of our idea on complex backgrounds and wide camera angles, *i.e.* CompCars dataset. To use CompCars dataset where EG3D is not applicable due to the absence of a camera pose estimator, we apply a sphere background to StyleNeRF, namely ***BallGAN-S***.

Training stability Figure 7 compares image quality of BallGAN-S and StyleNeRF using Fréchet Inception Distance (FID) [17] over iterations. While StyleNeRF diverges as the image resolution grows from 128^2 to 256^2[2], BallGAN-S smoothly converges below the reported FID of StyleNeRF. It implies that our method is generally beneficial to different foreground backbones and greatly improves training stability.

Comparisons In Figure 8, we present qualitative results of BallGAN-S, which showcase the robustness of our design on CompCars. Figure 8a shows that both GIRAFFE-HD and StyleNeRF exhibit a deficiency in fidelity in their modeled 3D compared to the quality of the generated images. On the other hand, ours maintains a high level of fidelity for both images and 3D models. In Figure 8b, we demonstrate

[2]This phenomenon is also reported in the official repository.

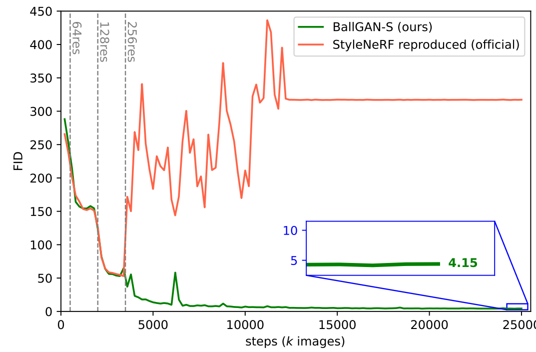


Figure 7: **FID over iterations on CompCars 256^2.** The FID score of StyleNeRF increases at 256^2 and becomes constant around 12K steps. In contrast, BallGAN-S exhibits stable training and achieves notably low FID score.

(a) Qualitative comparison of generated images and their corresponding 3D geometry.

(b) Separate renderings with BallGAN-S

Figure 8: **Results of BallGAN-S on CompCars** 256^2**.**

that our simple yet effective idea ensures successful separation of foreground and background, even for datasets with complex backgrounds and wide camera angles. Quantitative comparisons will be addressed in §4.4

4.3. Faithfulness of the underlying 3D geometry

It is essential for 3D-aware GANs to model the correct 3D geometry of the scenes so that their rendered images on arbitrary camera poses are convincing views of the real 3D scenes. Quantitative comparisons are followed by qualitative comparisons.

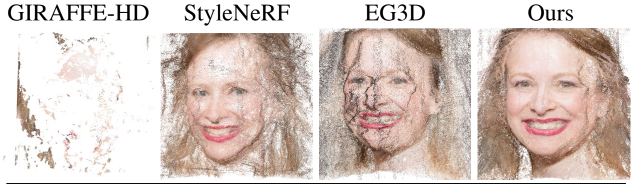
Quantitative results We quantitatively compare the underlying 3D model following the protocols in EG3D [5]. In Table 1, ID measures multi-view facial identity consistency[3], Depth indicates MSE of the expected depth maps from density against estimated depth-maps[4] in frontal view, and Pose implies controllability by MSE between the estimated pose of synthesized image and the input (target) pose. Appendix F describes further details of the protocol. BallGAN outperforms the baselines in all metrics evaluating 3D geometry.

[3]The mean Arcface [9] cosine similarity

[4]Estimations for Depth and Pose are from [10]

	FFHQ 512^2		
	ID ↑	Pose ↓	Depth ↓
MVCGAN	0.58	0.014	0.123
VolumeGAN	0.63	0.025	0.020
StyleSDF	0.50	0.010	0.016
EpiGRAF	0.71	0.013	0.143
EG3D	0.71	0.007	0.011
GIRAFFE-HD	0.69	0.064	0.058
StyleNeRF	0.64	0.018	0.013
Ours	**0.75**	**0.005**	**0.008**

Table 1: Quantitative evaluation on 3D geometry. We report identity consistency (ID), pose accuracy, and depth errors for FFHQ. Our method outperforms baselines in all metrics of 3D-awareness.



Method	GIRAFFE-HD	StyleNeRF	EG3D	Ours
# of rec. (10^4)	17 ± 2.3	53 ± 8.4	78 ± 5.5	79 ± 5.0

Table 2: **COLMAP point cloud reconstruction** is performed using 128 views in $[-\pi/2, \pi/2]$ from the generated scene for each model. A higher number of reconstructed points indicates better multi-view consistency.

We further push the evaluation: the number of reconstructed points from 128 views by COLMAP [34] in five inverted samples of FFHQ training set. Table 2 provides the numbers and example point clouds of the methods. Since COLMAP reconstructs the points with high photometric consistency, the larger number of points indicates higher multi-view consistency. BallGAN demonstrates superior performance in terms of multi-view consistency, especially in the face and hair region where the number of reconstructed points is substantially higher than other methods. While EG3D also achieves a similar number of reconstructed points as BallGAN, a large portion of these points lies on the background walls rather than the face. As the comparison results show, our sphere background induces the synthesis of accurate foreground geometry, thereby improving multi-view consistency.

Qualitative comparison: generated scenes Figure 9 compares how each method renders *generated* scenes on different perspectives, expecting the images to have multi-view consistency and realism. The leftmost column provides meshes of the scene for reference. We notice severe distortions in GIRAFFE-HD and StyleNeRF when the camera rotates more than $\pm 60°$ implying their spurious 3D geometry (red box in Figure 9). This problem is evident in the marching cube results of GIRAFFE-HD, which separately models foreground and background but without their separate ranges. StyleNeRF produces rough geometry and camouflages detailed shapes with color. Discussion on the missing backgrounds is deferred to Appendix G.

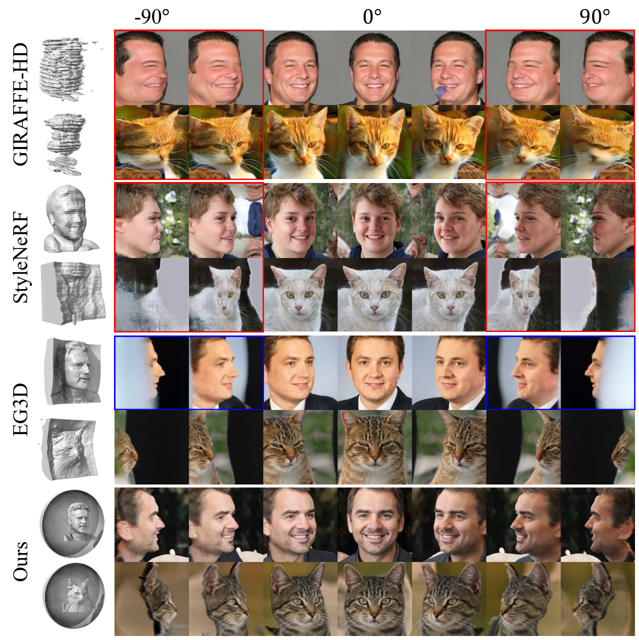


Figure 9: **Images rendered on various camera poses**. GIRAFFE-HD and StyleNeRF show distortions, especially on extreme camera poses (red boxes). The rendered images of EG3D are distorted by concave walls on extreme camera poses (blue boxes). In contrast, BallGAN synthesizes realistic and multi-view consistent images.

Similarly, the rendered images of EG3D show distortions from $\pm 60°$ angles, *e.g.*, the ears are truncated first and then the cheeks at $\pm 90°$ angles (blue box in Figure 9). The mesh explains that the faces are engraved to a concave wall expanding from the ridge of the faces. Furthermore, although the meshes show greater detail compared to StyleNeRF, there are areas of disagreement between the underlying geometry and its rendered images, *e.g.*, the boundary between hair and forehead is fuzzy in the geometry, whereas it becomes clear after color rendering.

On the other hand, BallGAN synthesizes realistic images that maintain consistency across multiple views, even when rendered in extreme side views. It implies that the separate background on a sphere removes the depth ambiguity and does not interfere with the foreground object. Notably, we observe a significant enhancement in fine details, such as hair and whiskers. For a more detailed multi-view comparison with all baseline models, please refer to Appendix I.

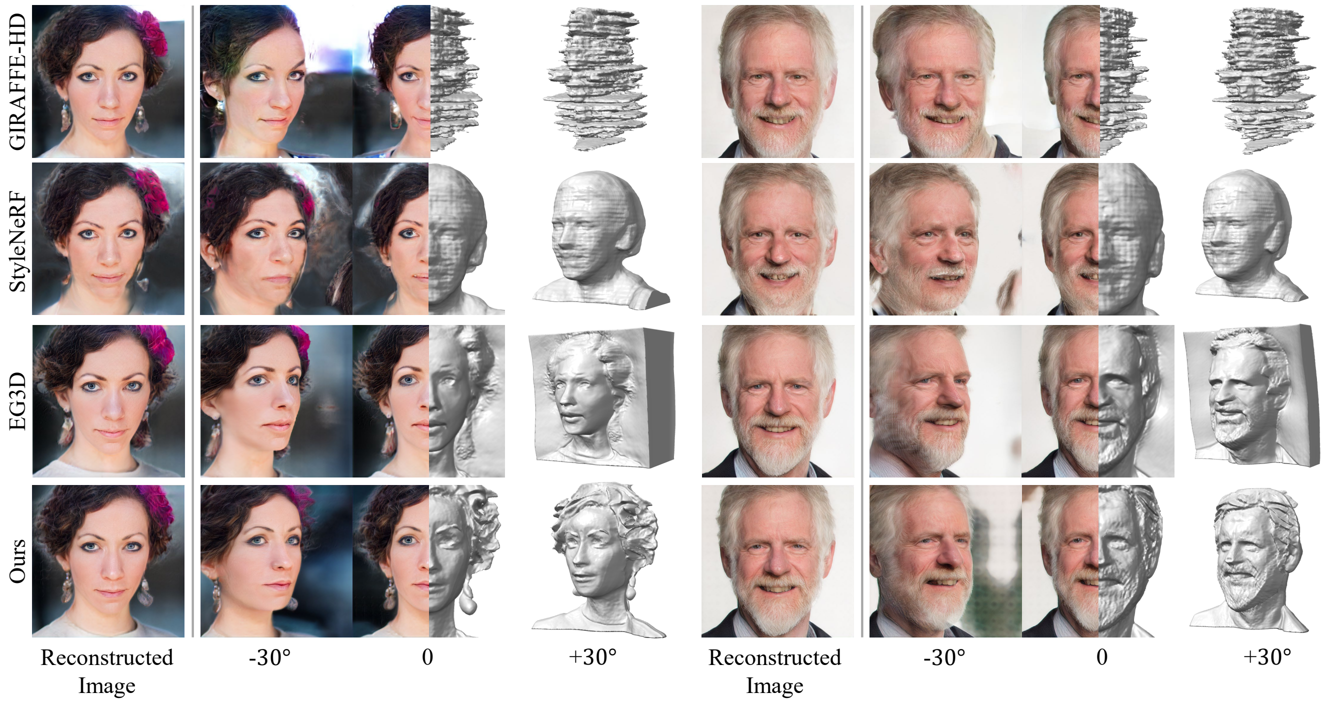


Figure 10: **Renderings and marching cubes of the same samples.** Given real image omitted as all models faithfully reconstruct it. Although all methods render the target image close by inversion, the underlying 3D geometries of previous methods are all different. We adjusted the threshold for each mesh at the line where the pupils do not break.

Qualitative comparison: inversion of real images Figure 10 compares renderings and meshes of the same scenes through pivotal tuning inversion (PTI) [33] of *real* images from the training set. Although the image reconstructions of all methods are similar in target pose, the differences become more visible in different viewpoints and in their underlying 3D geometries. GIRAFFE-HD apparently produces geometry that least fits the rendered image and thus renders inconsistent images in different views. StyleNeRF captures only rough outlines and placements in the geometry so that color makes the rendered scene realistic. Especially, the mesh does not reveal the beard and the boundary between hair and forehead. While EG3D can recover realistic geometry that mostly fits the given image, it has limitations such as faces being stuck to a wall. Moreover, it fails to accurately represent details such as eyebrows or accessories, which are evident in the input image. In contrast, BallGAN excels at accurately modeling the foreground in 3D space, and even faithfully represents the details shown in the images, such as wavy hair, earrings, and eyebrows.

4.4. Image quality

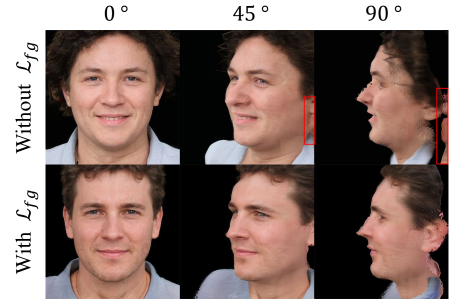
We evaluate generated image quality on the FFHQ 512^2, AFHQv2-Cats 512^2, CompCars 256^2 datasets. Images for FFHQ 512^2, AFHQv2-Cats 512^2 are generated by BallGAN and images for CompCars 256^2 are generated by BallGAN-S.

Sep. FG/BG		FFHQ 512^2	AFHQv2-Cats 512^2	CompCars 256^2
×	MVCGAN	13.4†	26.57‡	-
	VolumeGAN	15.74	44.55	12.9†
	StyleSDF	19.56	19.44	-
	EpiGRAF	9.92†	6.46	-
	EG3D	**4.7**†	**2.77**†	N/A
✓	GIRAFFE-HD	6.47	7.33	<u>7.1</u>‡
	StyleNeRF	10.51‡	21.56	8† (284±96)
	Ours	<u>5.67</u>	<u>4.72</u>	**4.26**

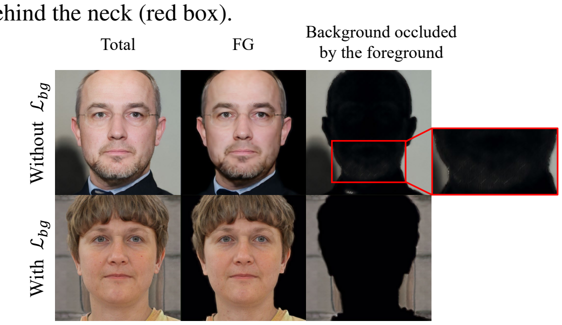
Table 3: Quantitative comparison using FID [17] on three datasets. † denotes the reported FID, and ‡ denotes the FID calculated by the official checkpoint. In other cases, we train each baseline using their official codes. In the case of StyleNeRF on CompCars, we report FID of diverged models over 3 experiments in the parenthesis. N/A denotes the model can not be trained. Bold and underline indicate the best and second-best performance. Our method shows the best score in CompCars and comparable scores with EG3D.

Quantitative results Table 3 compares image quality in FID. For FFHQ, AFHQv2-Cats, BallGAN outperforms all the baselines except EG3D. Although EG3D achieves the best FID, it does not support foreground-background separation and suffers in generating 3D geometry (§4.3). Furthermore, EG3D requires camera poses of real images,

Figure 11: **Set of images generated by BallGAN.** We sample images of 512^2 resolution from BallGAN on FFHQ 512^2 and AFHQv2-Cats 512^2, as well as 256^2 resolution images from BallGAN-S on CompCars 256^2. Each image is rendered with randomly sampled camera pose.



(a) **Visual comparison on the effect of foreground density regularization.** Removing $\mathcal{L}_{\text{fg}}$ introduces occasional floating objects behind the neck (red box).



(b) **Visual comparison on the effect of background transmittance regularization.** The use of $\mathcal{L}_{\text{fg}}$ results in a completely opaque foreground, rendering the background occluded by the foreground as entirely black.

Figure 12: **Ablations for two regularizations.**

which are not always available, e.g., CompCars. On the other hand, we achieve the state-of-the-art FID on CompCars with BallGAN-S and the second-best FID on FFHQ and AFHQv2-Cats closely following EG3D. We note that CompCars has more complex backgrounds and 360° camera poses.

Qualitative results Figure 11 provides example images generated by BallGAN and BallGAN-S. Our models faithfully generate diverse samples in multiple views. More examples can be found in Appendix J.

4.5. Ablation of the losses

We conduct ablation studies to evaluate the effect of the regularizers. Figure 12 shows the effects of our foreground and background regularization. Without $\mathcal{L}_{\text{fg}}$, BallGAN on FFHQ occasionally generates small floating objects behind faces. $\mathcal{L}_{\text{fg}}$ mitigates scene diffusion, thus inhibiting the formation of subtle shape artifacts such as floating objects behind the object. Additionally, using the background regularization $\mathcal{L}_{\text{bg}}$, we get clearer foreground-background separation. Figure 12b shows that removing $\mathcal{L}_{\text{bg}}$ allows the background to participate in synthesizing the foreground. For the result without $\mathcal{L}_{\text{bg}}$, the beard is not entirely black, indicating partial influence from the background (red box in Figure 12b). In other words, the foreground is not fully opaque. This is because the background transmittance loss $\mathcal{L}_{\text{bg}}$ encourages the foreground density to either completely block or leave the space empty before the rays hit the background.

5. Conclusion

We propose a 3D-aware GAN framework named BallGAN, which represents a scene as a 3D volume within a spherical surface, enabling the background representation to lie on a 2D coordinate system. This approach resolves the challenges of training a generator to learn a 3D scene from only 2D images. Our proposed framework successfully separates the foreground in a 3D-aware manner, which enables useful applications such as rendering foregrounds from arbitrary viewpoints on top of given backgrounds. BallGAN also achieves superior performance in 3D awareness, including multi-view consistency, pose accuracy, and depth reconstruction. Additionally, our approach shows significant improvement in capturing fine image details in 3D space, compared to existing methods.

Acknowledgements This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2022R1F1A1076241). The part of experiments was conducted on NAVER Smart Machine Learning (NSML) platform [23, 41].

References

[1] Jeongmin Bae, Mingi Kwon, and Youngjung Uh. Furrygan: High quality foreground-aware image synthesis. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIV*, pages 696–712. Springer, 2022. 2

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 14

[3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022. 4

[4] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[5] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022. 1, 2, 3, 4, 5, 6, 15

[6] 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. 1, 2

[7] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, pages 333–350. Springer, 2022. 2

[8] 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 (CVPR)*, June 2020. 5, 14

[9] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 6, 15

[10] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 6, 15

[11] Sara Fridovich-Keil, Giacomo Meanti, Frederik Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. *arXiv preprint arXiv:2301.10241*, 2023. 3

[12] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *2017 International Conference on 3D Vision (3DV)*, pages 402–411. IEEE, 2017. 2

[13] 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, 2014. 4, 14

[14] 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. 1, 2, 4, 5

[15] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *Advances in Neural Information Processing Systems*, 33:9841–9850, 2020. 1

[16] 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. 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. *Advances in neural information processing systems*, 30, 2017. 6, 8

[18] Ajay Jain, Ben Mildenhall, Jonathan T. Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. 2022. 1

[19] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 14

[20] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021. 5

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 5

[22] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *Proc. CVPR*, 2020. 4, 14

[23] Hanjoo Kim, Minkyu Kim, Dongjoo Seo, Jinwoong Kim, Heungseok Park, Soeun Park, Hyunwoo Jo, KyungHyun Kim, Youngil Yang, Youngkwan Kim, et al. Nsml: Meet the mlaas platform with a real-world case study. *arXiv preprint arXiv:1810.09957*, 2018. 9

[24] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. *arXiv preprint arXiv:2211.10440*, 2022. 1

[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. 4, 14

[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. In *European conference on computer vision*, pages 405–421. Springer, 2020. 2, 13

[27] 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. 1, 2

[28] 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. 2

[29] 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. 2, 4

[30] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13503–13513, June 2022. 5

[31] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021. 6

[32] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *International Conference on Learning Representations*, 2023. 1

[33] Daniel Roich, Ron Mokady, Amit H Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Transactions on Graphics (TOG)*, 42(1):1–13, 2022. 5, 8, 15

[34] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 7

[35] 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. 1, 2

[36] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9243–9252, 2020. 1

[37] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1532–1540, 2021. 1

[38] Alon Shoshan, Nadav Bhonker, Igor Kviatkovsky, and Gerard Medioni. Gan-control: Explicitly controllable gans. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14083–14093, 2021. 1

[39] Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, and Yaniv Taigman. Text-to-4d dynamic scene generation. *arXiv:2301.11280*, 2023. 1

[40] Ivan Skorokhodov, Sergey Tulyakov, Yiqun Wang, and Peter Wonka. Epigraf: Rethinking training of 3d gans. *arXiv preprint arXiv:2206.10535*, 2022. 2, 5

[41] Nako Sung, Minkyu Kim, Hyunwoo Jo, Youngil Yang, Jingwoong Kim, Leonard Lausen, Youngkwan Kim, Gayoung Lee, Donghyun Kwak, Jung-Woo Ha, et al. Nsml: A machine learning platform that enables you to focus on your models. *arXiv preprint arXiv:1712.05902*, 2017. 9

[42] 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. 3

[43] 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, 2016. 2

[44] Jiale Xu, Xintao Wang, Weihao Cheng, Yan-Pei Cao, Ying Shan, Xiaohu Qie, and Shenghua Gao. Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models. *arXiv preprint arXiv:2212.14704*, 2022. 1

[45] Yinghao Xu, Sida Peng, Ceyuan Yang, Yujun Shen, and Bolei Zhou. 3d-aware image synthesis via learning structural and textural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18430–18439, 2022. 2

[46] Yinghao Xu, Sida Peng, Ceyuan Yang, Yujun Shen, and Bolei Zhou. 3d-aware image synthesis via learning structural and textural representations. In *CVPR*, 2022. 5

[47] Yang Xue, Yuheng Li, Krishna Kumar Singh, and Yong Jae Lee. Giraffe hd: A high-resolution 3d-aware generative model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18440–18449, 2022. 1, 2, 4, 5

[48] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3973–3981, 2015. 2, 5

[49] Kaan Yücer, Alexander Sorkine-Hornung, Oliver Wang, and Olga Sorkine-Hornung. Efficient 3d object segmentation from densely sampled light fields with applications to 3d reconstruction. *ACM Transactions on Graphics (TOG)*, 35(3):1–15, 2016. 13

[50] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 2

[51] Weiwei Zhang, Jian Sun, and Xiaoou Tang. Cat head detection - how to effectively exploit shape and texture features. In *European Conference on Computer Vision*, 2008. 14, 16

[52] Xuanmeng Zhang, Zhedong Zheng, Daiheng Gao, Bang Zhang, Pan Pan, and Yi Yang. Multi-view consistent generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18450–18459, 2022. 5

[53] Peng Zhou, Lingxi Xie, Bingbing Ni, and Qi Tian. Cips-3d: A 3d-aware generator of gans based on conditionally-independent pixel synthesis. *arXiv preprint arXiv:2110.09788*, 2021. 1

[54] Qiran Zou, Yu Yang, Wing Yin Cheung, Chang Liu, and Xiangyang Ji. Ilsgan: Independent layer synthesis for unsupervised foreground-background segmentation. *arXiv preprint arXiv:2211.13974*, 2022. 2