Supplemental Material for BallGAN

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]

We provide the following supplementary materials:

A. Background design choice

This section explains the rationale why our background has a spherical shape rather than anything else. Notably, our goal is not to accurately model the geometry of the background, but rather to ensure that the integrity of the foreground of interest is not compromised. To ensure that the background is taken into consideration from all possible angles, it is imperative that the background encompasses the camera sphere. For instance, a planar background fails to cover the background when the camera rotates beyond 90° from its normal vector.

Even if the view frustum can account for the entire background, any abrupt changes in gradient or inconsistencies in distances from the camera can engender unstable learning.

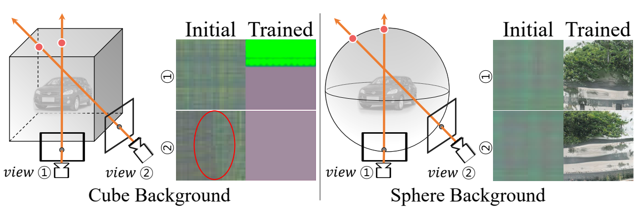


Figure S1: **Background should be modeled spherical rather than cubic.** While the edges of the cube are reflected in the rendered images (*Initial*), the sphere has no such artifacts in the rendered images. While the cubic background fails to produce plausible images, our spherical background produces sensible backgrounds (*Trained*).

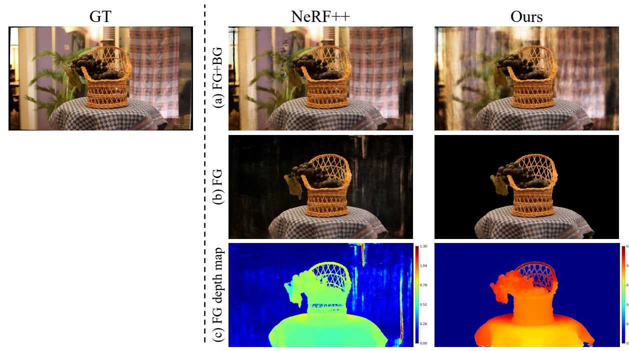


Figure S2: **Effectiveness of our spherical background on single scene overfitting scenario.** The sole foreground rendering and depth map demonstrates our spherical background is beneficial for capturing foreground geometry

To analyze the background effect, we trained BallGAN-S on the CompCars dataset with various complex background representations that occupy a significant portion of the image, using only different representations of the background such as sphere and cube, in Figure S1. The cube background does not converge. Therefore, the sphere background is the only reasonable choice for background representation.

B. Effectiveness of background representation

In this section, we demonstrate the effect of our spherical background representation, which enhances the focus on the foreground. We verify the efficacy of our background representation through a single-scene overfitting (SSO) experiment, in which we overfit a 3D model to a single scene captured by multi-view images, namely lf-basket [49]. We use the vanilla NeRF [26] for the foreground, and keep the spherical background representation. In other words, NeRF++ and Ours differ only in the background representation.

As shown in Figure S2, NeRF++ does not clearly distinguish between foreground and background, and the estimated depth is erroneous, e.g., the table has a lower depth at the deepest end. In contrast, our approach clearly sepa-

	configuration		
	$\mathcal{L}_{\text{fg}}$	$\mathcal{L}_{\text{bg}}$	FID
stage 1	-	-	7.87
	✓	-	6.82
	-	✓	7.88
	✓	✓	6.13

Table S1: **Ablation study on regularization.** This ablation study is conducted with batch size 16 due to the resource shortage. FIDs do not match the main results.

rates foreground and background and better estimates foreground depth. Thus, our design demonstrates effectiveness in focusing resources on learning foreground 3D geometry.

C. Ablation of the losses

We conduct ablation studies to evaluate the impact of each regularization on image quality. Table S1 shows the effects of our foreground and background regularization. Applying the foreground density loss $\mathcal{L}_{\text{fg}}$ improves FID. The background transmittance regularization $\mathcal{L}_{\text{bg}}$ not only facilitates a clearer separation between foreground and background but also enhances FID score.

D. Implementation details

BallGAN Our implementation mostly follows the official implementation of EG3D[1] including training hyperparameters, dual discrimination, pose-conditioning on discriminator, two-stage training, equalized learning rates [19], a mini-batch standard deviation layer at the end of the discriminator [19], exponential moving average of the generator weights, a non-saturating logistic loss [13], and R1 regularization [25] with $\gamma = 1$. We also use the same camera intrinsic parameters and FFHQ preprocessing from EG3D.

The weights of the foreground density output layer are initialized to zero to guarantee the contribution of the background at the beginning of the training. Figure S3 illustrates the architecture for the background representation. A five-layer 1×1 convolutional network maps the positional encoding ζ of a background point to a feature vector. The style code from an eight-layer MLP, *i.e.*, the mapping network, modulates the weights of the convolutions $\mathbf{g}_{\mathbf{w}_{\text{bg}}}$. The background representation mapping network shares the same design as the mapping network in StyleGAN2 [22]. The number of channels of the intermediate features are in Table S2. The last layer has a sigmoid clamping from MipNeRF [2] as in the foreground neural render of EG3D. We use the positional encoding of $L = 10$ on the background's 2D spherical coordinates. View direction is not considered for our background representation.

[1] https://github.com/NVlabs/eg3d

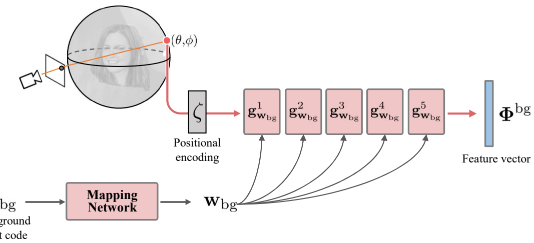


Figure S3: **Background architecture**

	input channel	output channel
PE	2	40
$\mathbf{g}^1_{\mathbf{w}_{\text{bg}}}$	40	64
$\mathbf{g}^2_{\mathbf{w}_{\text{bg}}}$	64	64
$\mathbf{g}^3_{\mathbf{w}_{\text{bg}}}$	64	64
$\mathbf{g}^4_{\mathbf{w}_{\text{bg}}}$	64	64
$\mathbf{g}^5_{\mathbf{w}_{\text{bg}}}$	64	32

Table S2: **Detail of background network.** *PE* means positional encoding ζ, not a layer.

On FFHQ, we schedule the coefficient of the foreground density loss λ_{fg} to exponentially grow from 0 to 0.25 and the coefficient of the background transmittance regularization λ_{bg} to exponentially grow from 0 to 1 in the first stage. We set the coefficients $\lambda_{\text{fg}} = 1$ and $\lambda_{\text{bg}} = 0.5$ in the second stage.

For AFHQv2-Cats, we start from the weights pretrained on FFHQ for the first step and fine-tune them on AFHQv2-Cats as done in EG3D. We set $\lambda_{\text{fg}} = \lambda_{\text{bg}} = 0$ to let the foreground better capture the fine details such as whiskers.

BallGAN-S BallGAN-S is a variant using StyleNeRF as a baseline instead of EG3D. We add the same background network on top of the official StyleNeRF implementation[2]. We set $\lambda_{\text{fg}} = 0.25$ and $\lambda_{\text{bg}} = 0$.

Competitors In the comparison experiments, we reported the best FIDs among the available sources: reported, official checkpoints, and official training code. We used the official training codes as-is to reproduce FIDs if the official repository does not provide the checkpoints[3][4][5][6].

StyleNeRF, StyleSDF, EpiGRAF, and VolumeGAN do not provide training guidelines for AFHQv2-cats [8]. For StyleNeRF and StyleSDF, we adopted the same training settings as used for AFHQv2 training, given that AFHQv2-cats constitutes a subset of AFHQv2. For VolumeGAN, we followed the same settings as Cats [51] in pi-gan, including

[2] https://github.com/facebookresearch/StyleNeRF
[3] https://github.com/genforce/volumegan
[4] https://github.com/universome/epigraf
[5] https://github.com/royorel/StyleSDF
[6] https://github.com/AustinXY/GIRAFFEHD

	FFHQ 512^2			FFHQ other res.
	reported	reproduced	official ckpt.	reported
GRAM	-	-	-	(256^2) 29.8
MVCGAN	**13.4**	-	21.3	
VolumeGAN	-	**15.7**	-	(256^2) 9.1
StyleSDF	-	**19.5**	-	(256^2) 11.5
EpiGRAF	**9.9**	-	-	(256^2) 9.7
EG3D	**4.7**	4.7	-	
GIRAFFE-HD	-	**6.4**	-	(1024^2) 10.13
StyleNeRF	13.2	-	**10.5**	
Ours	**5.64**			

Table S3: **FIDs of competitors from various sources.** We report the best FID among the reported, reproduced and official checkpoint for each model with 512^2 resolutions in Table 3.

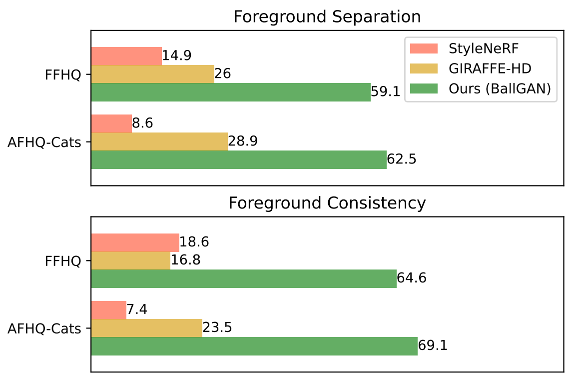


Figure S4: **User study.**

FOV, ray's near/far distances, and camera pose sampling distribution. For EpiGRAF, we employed the landmark detector[7] used in EG3D to label camera poses, while following the guidelines from the EpiGRAF's official repository for other training settings. The FOV and ray's near/far distances used in EpiGRAF are almost identical to those in pi-gan.

For GIRAFFE-HD on CompCars, we applied transfer-learning from the official checkpoint for 256^2 resolution to 512^2 resolution following the authors' guidelines. We trained the model until it achieved the FID reported in the original paper. Table S3 provides the FIDs we obtained from various sources.

E. User study

We asked 57 participants to choose the best model in terms of foreground separation and consistency. We prepared the following questionnaire for our user study in Figure S4. We randomly sampled ten scenes from each method and rendered foregrounds in seven different viewing directions; the entire samples are shown in §F. Then we asked 57 participants to answer two questions: (1:Foreground Separation) Which set of foreground fully includes the whole person (or cat) and excludes the background? (2 : Foreground Consistency) Which set of foregrounds is consistent across different views?

Figure S4 shows that ours outperforms competitors by a large margin with respect to both criteria. See §F for how we prepared images for the user study.

F. Evaluation protocols

We mostly follow the evaluation protocols of EG3D[5]. Below enumerates the protocols.

Real image inversion We use the same configuration of EG3D for pivotal tuning inversion [33].

ID ID measures the cosine similarity of the ArcFace embedding [9] between different views of the same scene. For each method, we generate 1000 random scenes in pairs of random poses from the training dataset pose distribution. Then we compute the average.

Pose Pose computes the difference between the intended (input) pose and the synthesized pose, implying how accurately the input poses are reflected in the rendered poses. We sample 1000 latent codes and render them in varying yaws and estimate the resulting yaws with a pre-trained face reconstruction model [10]. Instead of random yaws, we remove the stochasticity of the evaluation by specifying nine yaw angles evenly separated in [-0.9rad, 0.9rad]. $\pm$0.9rad covers the [0.3, 99.7] percentile of the training dataset's yaw distribution. We report a mean absolute error (L1) instead of L2 distance to equally capture the error near zero.

Depth Depth measures the difference between the underlying 3D geometry (volume-rendered depth) and the rendered image. We consider depth maps of rendered images in frontal views of 1000 samples estimated by a pre-trained 3D face reconstruction model [10] as pseudo ground truth. The depth maps are normalized to compute their mean squared error.

Foreground separation We describe the procedure to obtain the foreground image used in §4.1. Although our goal is to compare the separation of foreground and background in the 3D space, it is prohibitive to visualize the separation in 3D space on paper or screen. Therefore, we visualize by separately synthesizing the foreground scene for each method. Note that GIRAFFE-HD produces extra alpha masks in 2D space. We visualize their foreground part with their alpha masks to demonstrate their best performance. Their foreground densities are only in the central region of the image canvas, and their aggregated densities do not match the shape of the salient object. For StyleNeRF, the foreground densities along the ray do not sum to one, *i.e.*, the foreground is semi-transparent. Therefore, we manually searched for a density threshold that best divides

[7] https://github.com/kairess/cat_hipsterizer

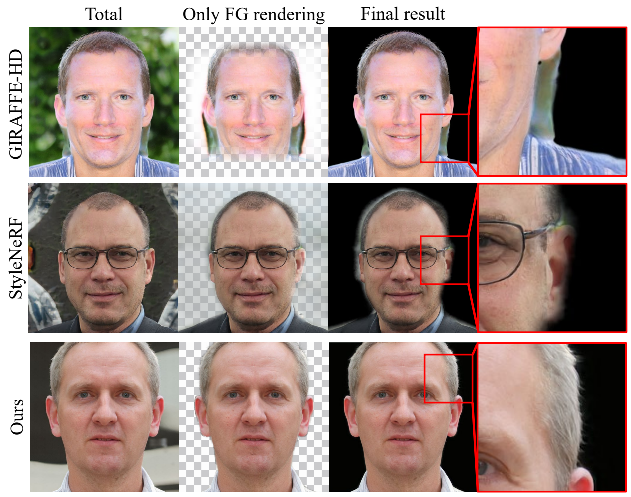
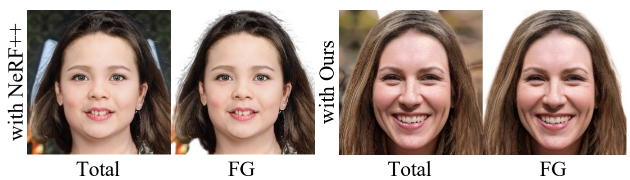
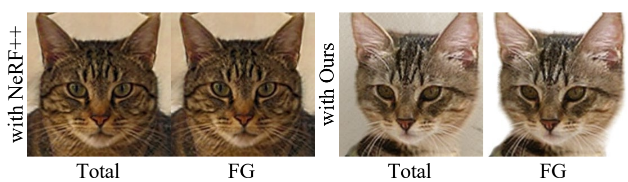


Figure S5: **Foreground separation examples.** The densities along a ray do not sum to one in GIRAFFE-HD and StyleNeRF. Hence, we apply postprocessing to compare their full potential for separation. Ours does not require such postprocessing. The rightmost column shows zoomed-in images of red box regions for detailed comparison.



(a) Comparison on FFHQ



(b) Comparison on Cats

Figure S6: **Comparison of foreground and background separation with EpiGRAF backbone** NeRF++ BG struggles on hair, shoulder, and cat. Our BG excels in all cases.

the foreground region for each image. Ours do not require such workarounds as the foreground densities aggregate to one along the rays well on the foreground regions. Figure S5 provides examples.

G. Detailed qualitative comparison

We only visualize the foreground meshes in Figure 8, Figure 10, Figure S7, and Figure 6 for methods that separately model on foreground and background. Figure 1, Figure 2 and Figure 9 show the full 3D scene, including both foreground and background. As EG3D does not separate foreground and background, the full 3D geometry is visualized on all mesh figures.

(a) EG3D

(b) Ours

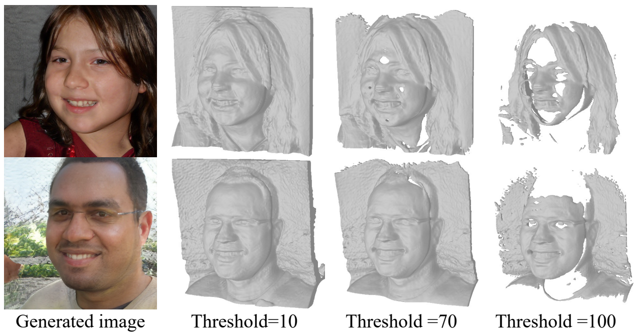
Figure S7: **3D geometry comparison between EG3D and BallGAN**

However, we only visualize the foreground mesh of StyleNeRF in Figure 9 as we discover that the background densities of StyleNeRF are close to zero, thus negligible. Yet, the background appears on rendered images of StyleNeRF as the last sample on the background ray is set to have an alpha value of 1 before volume rendering, i.e., the alpha value for the last sample is tweaked to 1 regardless of the actual density produced by the background NeRF.

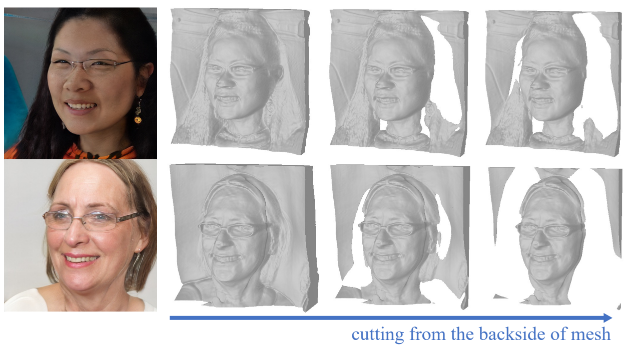
Despite the sole visualization of foreground mesh for StyleNeRF in Figure 9, densities accountable for background is noticeable on StyleNeRF's mesh for AFHQv2-Cats. This shows the case of the background being erroneously modeled through the foreground.

EpiGRAF employs NeRF++'s inverse sphere parameterization for the background, the same as StyleNeRF. Figure S6 shows a comparison between our background representation and NeRF++ when using EpiGRAF as the backbone. The term "with NeRF++" refers to the original EpiGRAF, while "with Ours" indicates the model where our sphere background representation is applied to EpiGRAF's foreground representation. Except for the background representation, all settings remain the same and adhere to the guidelines provided in the official repository.

In FFHQ, EpiGRAF with Ours separates the FG cleaner. On the Cats [51] dataset, which contains a significant amount of fine-grained details, EpiGRAF with NeRF++ fails to separate the FG and BG, whereas EpiGRAF with Ours shows clear separation.



(a) 3D comparison of density threshold for EG3D



(b) 3D comparison of cutting mesh for EG3D

Figure S8: **Difficulty of separating foreground in EG3D** (a) The background cannot be removed by thresholding density, i.e., the foreground is cut off before the background is fully removed. (b) As the background wall has a concave shape and is not always behind the foreground, clipping with depth tends to carve out the foreground before full background removal.

H. More comparison with EG3D

EG3D does not separately model foreground and background. Figure S7 highlights the drawback of this representation for learning 3D scenes. The ears and hair in 3D space are attached to the background. Some parts of the hair are flat and lack curls. In contrast, ours separates the hair from the background and correctly models the 3D geometry of the hair that matches the 2D observation.

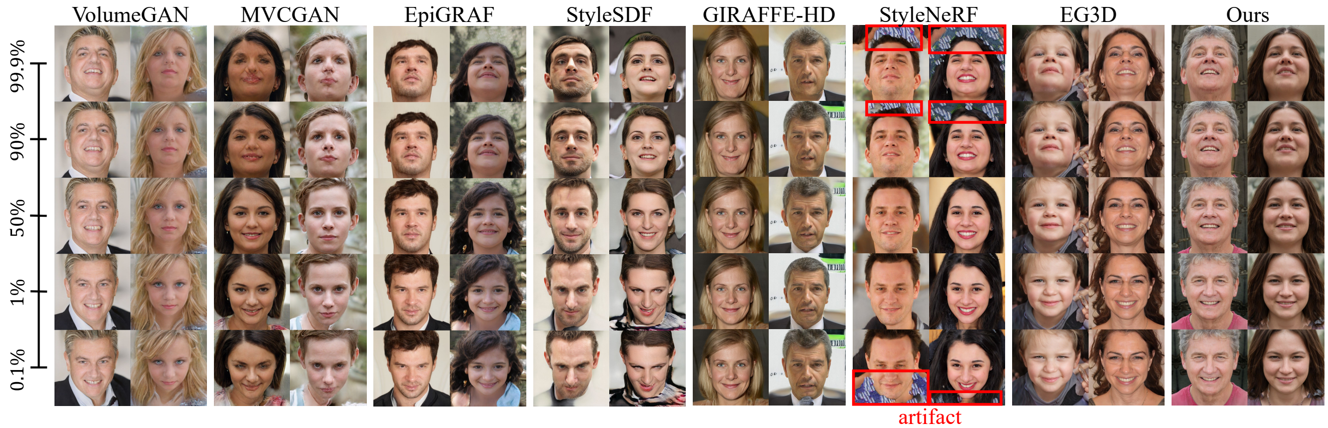
Figure S8 shows that foreground separation is not straightforward in EG3D's 3D space. Thresholding the density or carving the mesh from the back does not correctly separate the foreground, and damages the facial/hair regions first. This demonstrates that the foreground and background must be perfectly separated at the representation level.

I. Detailed multi-view comparison

Figure S9a and Figure S9b provide qualitative comparisons with varying camera poses. As FFHQ dataset mainly consists of frontal views, the competitors produce artifacts or show multi-view inconsistency. On the other hand, BallGAN produces images that are multi-view consistent and free from artifacts even in extreme camera poses.

J. Uncurated samples

Figure S10 provides uncurated samples of our method.



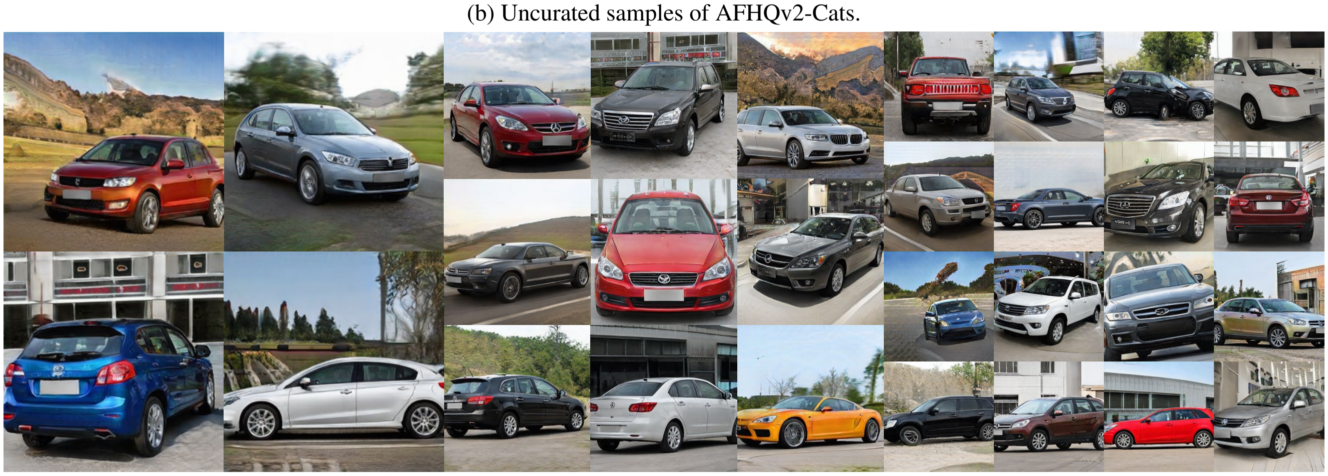
(a) Multi-view comparison with varying pitches

(b) Multi-view comparison with varying yaws

Figure S9: **Multi-view comparison in various poses on FFHQ.** Percentile for camera pitch and yaw in training distribution are shown on the left side of a and below for b.

(a) Uncurated samples of FFHQ.

(b) Uncurated samples of AFHQv2-Cats.



(c) Uncurated samples of CompCars.

Figure S10: **Uncurated samples on the FFHQ, AFHQv2-Cats, and CompCars.** Camera poses are randomly chosen from each training distribution. a and b show outputs of BallGAN. c is outputs from BallGAN-S.

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