What You See is What You GAN:
Rendering Every Pixel for High-Fidelity Geometry in 3D GANs

Alex Trevithick[*2], Matthew Chan[1], Towaki Takikawa[1], Umar Iqbal[1], Shalini De Mello[1], Manmohan
Chandraker[2], Ravi Ramamoorthi[2], and Koki Nagano[1]

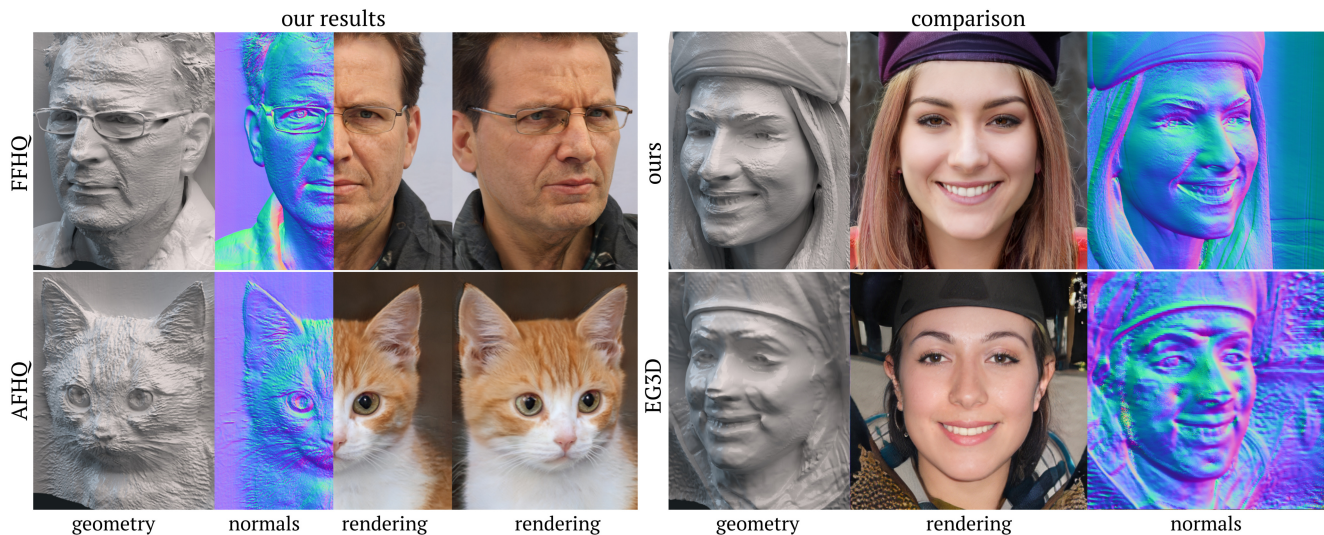[1]NVIDIA   [2]University of California, San Diego

Figure 1. Left: Our results. The split view in the middle demonstrates the high degree of agreement between our 2D rendering and corresponding 3D geometry. Our method can learn fine-grained 3D details (e.g., eyeglass frame and cat's fur) that are geometrically well-aligned to 2D images without multiview or 3D scan data. Right: Comparison with EG3D [7]. Our tight SDF prior provides smooth and detailed surfaces on the face and hat while EG3D exhibits geometry artifacts and discrepancies between geometry and rendering. Please see Fig. 5 and the accompanying video for more examples, and Fig. 6 for comparison to other baselines.

Abstract

3D-aware Generative Adversarial Networks (GANs) have shown remarkable progress in learning to generate multi-view-consistent images and 3D geometries of scenes from collections of 2D images via neural volume rendering. Yet, the significant memory and computational costs of dense sampling in volume rendering have forced 3D GANs to adopt patch-based training or employ low-resolution rendering with post-processing 2D super resolution, which sacrifices multiview consistency and the quality of resolved geometry. Consequently, 3D GANs have not yet been able to fully resolve the rich 3D geometry present in 2D images. In this work, we propose techniques to scale neural volume rendering to the much higher resolution of native 2D images, thereby resolving fine-grained 3D geometry with

unprecedented detail. Our approach employs learning-based samplers for accelerating neural rendering for 3D GAN training using up to 5 times fewer depth samples. This enables us to explicitly "render every pixel" of the full-resolution image during training and inference without post-processing superresolution in 2D. Together with our strategy to learn high-quality surface geometry, our method synthesizes high-resolution 3D geometry and strictly view-consistent images while maintaining image quality on par with baselines relying on post-processing super resolution. We demonstrate state-of-the-art 3D gemetric quality on FFHQ and AFHQ, setting a new standard for unsupervised learning of 3D shapes in 3D GANs.

1. Introduction

Training 3D generative models from the abundance of 2D images allows the creation of 3D representations of real-world objects for content creation and novel view synthe-

---

*This project was initiated and substantially carried out during an internship at NVIDIA.

sis [59]. Recently, 3D-aware generative adversarial networks (3D GANs) [6, 7, 19, 43, 45, 49, 53, 69, 70, 74, 76] have emerged as a powerful way to learn 3D representations from collections of 2D images in an unsupervised fashion. These methods employ differentiable rendering to compare rendered 3D scenes with 2D data using adversarial training [18]. Among the various 3D representations, Neural Radiance Fields (NeRF) [38] have become a popular choice among recent successful 3D GANs. However, the significant computational and memory cost of volume rendering has prevented 3D GANs from scaling to high-resolution output. For instance, generating a single 512x512 image via volume rendering requires evaluating as many as 25 million depth samples, if 96 depth samples are used per ray using importance sampling [7, 10, 38, 53]. Given that GAN training typically requires rendering tens of millions of images, the training process could require evaluating *hundreds of trillions* of depth samples.

During training, all intermediate operations must be stored in GPU memory for every depth sample for the backward pass. Therefore, existing methods resort to working on patches [10, 49, 53] or adopting a low resolution neural rendering combined with post-processing 2D super resolution (SR) [7, 19, 43, 45, 70]. However, patch-based methods have limited receptive fields over scenes, leading to unsatisfactory results, and the hybrid low-resolution rendering and SR scheme inevitably sacrifices the multiview consistency and the accuracy of 3D geometry. While many techniques have been developed to improve the image quality of 3D GANs to match that of 2D GANs, the challenge of resolving the corresponding high-resolution 3D geometry remains unsolved (see Figs. 1 and 6 for our results and comparison to the current state-of-the-art).

Scaling 3D GANs to operate natively at the 2D pixel resolution requires a novel approach for sampling. Fig. 2 compares the state-of-the-art 3D GAN, EG3D model, trained with and without[1] SR. EG3D employs 96 dense depth samples in total using two-pass importance sampling [38] during training, which requires half a terabyte of GPU memory at 256×256 resolution, making scaling to higher resolutions infeasible. Furthermore, Fig. 2 demonstrates that using 96 dense samples still results in undersampling, as evidenced by the speckle noise patterns visible in the zoomed-in view, leading to considerably worse FID (inset in Fig. 2). 3D GANs relying on post-processing SR layers can repair these undersampling artifacts at the cost of a high-fidelity 3D representation.

In this work, we address the challenge of scaling neural volume rendering to high resolutions by explicitly rendering every pixel, ensuring that "*what you see in 2D, is what you get in 3D*" — generating an unprecedented level of geometric details as well as strictly multiview-consistent images. Our contributions are the following:

- We introduce an SDF-based 3D GAN to represent high-

Figure 2. Samples from EG3D 256 model. Right: Volume rendering with 48 coarse samples and 48 fine samples per ray with two-pass importance sampling [38] results in undersampling, leading to noticeable noisy artifacts. Left: These artifacts are repaired by super resolution (SR). An unsharp mask has been applied to the zoomed views for presentation purposes.

frequency geometry with spatially-varying surface tightness that increases throughout training (subsection 4.1 and 4.5), in turn facilitating low sample rendering.
- We propose a generalizable learned sampler conditioned on cheap low-resolution information to enable full-resolution rendering *during training* for the first time (subsection 4.2 and 4.3).
- We show a robust sampling strategy for the learned sampler (subsection 4.4) that produces stable neural rendering using significantly fewer depth samples (see Fig. 8). Our sampler can operate with just 20 samples per ray compared to existing 3D GANs which must use at least 96 samples per ray.
- Together, our contributions result in the state-of-the-art geometry for 3D GANs while rendering with quality on par with SR baselines (see Fig. 1). For more results, see Fig. 5 and for comparison to other baselines, see Fig. 6.

2. Related Work

We begin by reviewing the prior-arts of 3D generative models and their current shortcomings. We then cover foundational techniques for 3D geometry representation and neural rendering from which we take inspiration. We then discuss existing methods for accelerating neural volume rendering, which usually operate *per-scene*.

2.1. 3D Generative Models

Just like 2D GANs, 3D-aware GANs train from a collection of 2D images, but employ a 3D representation and differentiable rendering to learn 3D scenes without requiring multiview images or ground truth 3D scans. Some of the most successful works use a neural field [68] in combination with a feature grid [7] as their 3D representation, and use neural volume rendering [38] as the differentiable renderer.

However, due to the significant memory and computational cost of neural volume rendering, many previous works perform rendering at low-resolution and rely on a 2D post-processing CNN [7, 19, 43, 45, 70], which *hallucinates* the high-frequency details in a view-inconsistent manner while sacrificing 3D consistency and the quality of the resolved 3D geometry.

To ensure strict 3D consistency, other previous works seek to render at high-resolutions and propose techniques to address the prohibitive computational costs. One line of work leverages the sparse nature of 3D scenes to speed up rendering, in particular, structures such as 2D manifolds [14, 67], multiplane images [75] and sparse voxels [50]. Although they are more efficient, sparse representations provide only coarse [50] or category-specific [14, 67] acceleration structures, which poses constraints on the diversity and viewing angles of the generated scenes. Our sampling-based method, on the other hand, generalizes to every new scene and adaptively accelerates rendering on a per ray basis. Another line of work enables high-resolution rendering with patch-based training [10, 53]. In particular, Mimic3D [10] achieves significantly improved 2D image quality, but the patch-based training limits the receptive fields, and the generated geometry does not faithfully represent the 2D data due to the patch-wise perceptual loss. Our method renders the entire image at once and the resulting geometry is aligned with the rendering (see Figs. 1 and 6).

Recently, a new family of generative approaches using diffusion models has been proposed to tackle conditional tasks including novel view synthesis [8, 58] and text-based 3D generation[63]. Most of these 3D-aware diffusion models combine a 3D inductive bias modeled via neural field representations and a 2D image denoising objective to learn 3D scene generation. While these models enable unconditional 3D generation, they require multiview images [24, 55, 63] or 3D data, such as a point cloud [42]. Score Distillation Sampling [47] may be used for distillation from a pre-trained 2D diffusion model when only monocular 2D data is available, but diffusion models incur significant computational costs due to their iterative nature and most of the existing methods require optimization per scene [9, 20, 33, 61, 66].

2.2. Learning High-Fidelity Geometry

Prior works on 3D GANs have typically represented the geometry as a *radiance field* [38], which lacks a concrete definition for where the surface geometry resides in the field, resulting in bumpy surfaces. A number of works [44, 62, 71] have proposed alternate representations based on implicit surfaces (such as signed distance functions, or SDFs) that can be used with neural volume rendering. In these works, the implicit surface is typically *softened* by a parameter for volume rendering.

Other works [32, 64] improve on these implicit surface representations by leveraging feature grids for higher computational efficiency and resolution. Adaptive Shells [65]

further improve on quality by making the *softness* parameter spatially-varying, as many objects have hard boundaries only in certain parts. We use an implicit surface representation based on VolSDF [72], and leverage a spatially-varying parameter similar to Adaptive Shells [65] to control the softness of the surface, as humans and animals benefit from both hard surfaces (e.g. skin, eyes) and *soft*, volumetric representations (e.g. hair). Although other works such as StyleSDF [45] have similarly leveraged implicit surfaces in a 3D GAN framework, the lack of spatial-variance and high-resolution rendering led to over-smoothed geometry not faithful to the rendered images.

2.3. Accelerating Neural Volume Rendering

As mentioned in Section 2.1, accelerating 3D GANs typically relies on acceleration structures such as octrees [31, 56, 73], which in generative settings [50] are limited due to computational requirements. In another direction, GRAM-based methods [14, 67] are limited due to sharing one structure across all samples. Instead, we look to a class of methods that do not rely on an acceleration structure. Some of these works learn a per-scene sampling prior on a per-ray basis using a binary classifier [41], a density estimator [30, 46], a coarse proxy from a 3D cost volume [34], or an interval length estimator [35] on discrete depth regions along the ray. Other works use importance sampling [21, 38] to sample additional points. We take inspiration from works on density estimators [30] and propose to learn a *scene-conditional* proposal network that generalizes across scenes instead of being category-specific or optimized per-scene.

There are also other methods to accelerate rendering by utilizing more efficient representations, such as gaussian splats [29] and light field networks [52]. More efficient feature grids [40, 57] based on hashing can also be used to accelerate rendering. However, mapping to these representations in a GAN framework is not straightforward. In contrast, our sampling strategy can be used for any NeRF representation.

3. Background

We begin with background on the methodology of the state-of-the-art 3D-aware GANs as our method relies on a similar backbone for mapping to 3D representations. 3D GANs typically utilize a StyleGAN-like [27] architecture to map from a simple Gaussian prior to the conditioning of a NeRF, whether that be an MLP [19, 45], MPI [75], 3D feature grid [50], manifolds [14, 67] or triplane [7, 10, 53]. We inherit the latter triplane conditioning for its high expressivity and efficiency, in which three axis-aligned 2D feature grids (f_{xy}, f_{xz}, f_{yz}), provide NeRF conditioning by orthogonal projection and interpolation. As in the previous methods, the mapping and synthesis networks from StyleGAN2 can easily be adapted to create the 2D triplane representation from noise $z \in \mathbb{R}^{512}$. Specifically, $\mathbf{w} = \text{Mapping}(z)$ conditions the creation of the triplane $T(\mathbf{w}) \in \mathbb{R}^{3 \times 32 \times 512 \times 512}$
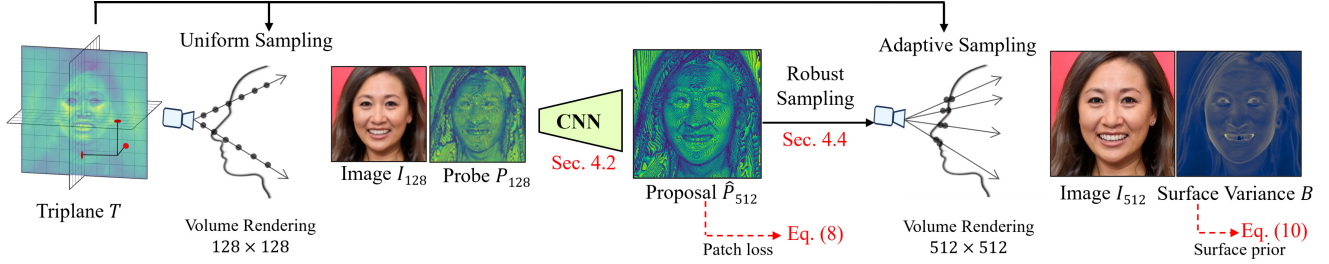
Figure 3. Here we show our proposed pipeline and its intermediate outputs. Beginning from the triplane T, we trace uniform samples to probe the scene, yielding low-resolution I_{128} and weights P_{128}. These are fed to a CNN which produces high-resolution proposal weights $\hat{P}_{512}$ (weights are visualized as uniform level sets). We perform robust sampling and volume render to get the final image I_{512} and the surface variance B.

from a Synthesis network, corresponding to three axis-aligned 2D feature grids of spatial resolution 512×512 and feature size 32.

To create high-fidelity geometry, our method builds upon VolSDF [72]: Instead of directly outputting the opacity σ of a point $x \in \mathbb{R}^3$, an SDF value s is output and transformed to σ using a Laplacian CDF:

$$\sigma = \frac{1}{\beta} \begin{cases} \frac{1}{2} \exp\left(\frac{s}{\beta}\right) & \text{if } s \leq 0 \\ 1 - \frac{1}{2} \exp\left(-\frac{s}{\beta}\right) & \text{if } s > 0 \end{cases} \tag{1}$$

where β is the variance of the Laplacian distribution governing the "tightness" of the representation. One distinct benefit of an SDF-based representation is the ease of extracting the surface. StyleSDF [45] also utilizes this intermediate geometry representation without a triplane, enforcing the usual Eikonal constraint.

Using the triplane conditioning and Eq. 1, we can assign each point in the volume with its opacity σ and radiance c using a lightweight MLP. For a given camera ray $r(t) = o + td$, we approximate the volumetric rendering integral $C(r)$ [36] by sampling ray distances t_i with their corresponding σ_i and c_i before computing

$$\hat{C}(r) = \sum_{i=1}^{N} w_i c_i \text{ where } w_i = T_i(1 - \exp(-\sigma_i \delta_i)),$$
$$T_i = \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right) \text{ and } \delta_i = t_{i+1} - t_i. \tag{2}$$

Here, T_i denotes the accumulated transmittance and δ_i is the distance between adjacent samples along the ray.

It is possible to develop a more efficient estimator for this sum with fewer samples by using importance sampling techniques in computer graphics. Typically, one computes a piecewise constant probability distribution $p_j = \frac{\hat{w}_j}{\sum_j \hat{w}_j}$, where j refers to the jth bin or region, and $\hat{w}_j$ is an estimate of w_j for that region, for example obtained by explicitly tracing coarse samples. For a given t, we first find the region $j(t)$ and then set $p(t) = p_j$. From this, one can compute a (piecewise linear) cumulative distribution function or CDF

$\Phi(t)$ which has a range from 0 to 1. We can then perform inverse CDF sampling to define the sample points,

$$t_i = \Phi^{-1}(u_i), \tag{3}$$

where u_i is a random number from 0 to 1 (sorted to be an increasing sequence).

Discussion We improve on previous works such as NeRF [38] and EG3D [7] by stratifying[2] the random numbers u_i during training; this leads to significantly lower rendering variance, especially at low sample counts N [39]. We also develop a neural method to predict a good distribution p_j (and hence Φ) for importance sampling at high spatial resolution, without needing to exhaustively step through the ray.

4. Method

In this section, we describe our method beginning with our SDF-based NeRF parametrization (subsection 4.1). We then overview how we render at high-resolution in three stages: first, a low-resolution probe into the 3D scene (subsection 4.2); second a high-resolution CNN proposal network (subsection 4.3); and third a robust sampling method for the resultant proposals (subsection 4.4). Next we describe regularizations (subsection 4.5) for stable training, and finally our entire training pipeline (subsection 4.6).

4.1. Mapping to a 3D Representation

Beginning from a noise vector z, we synthesize the initial triplane T' with StyleGAN [25] layers as detailed in Sec. 3. In contrast to previous methods, we then generate more expressive triplane features T with an extra synthesis block for each orthogonal plane: $f_{ij} = \text{SynthesisBlock}(f'_{ij})$ where $ij \in \{xy, xz, yz\}$ and f' are the features of T'. This design choice allows disentanglement between the intermediate triplane features as plane-specific kernels can attend to the features in each separate plane.

Given the triplane T (the left side of Fig. 3) and a point $x \in \mathbb{R}^3$, we utilize an MLP to map to the SDF value s, variance β and geometry features f_{geo}:

$$(s, \beta, f_{geo}) = \text{MLP}_{\text{SDF}}(\text{PosEnc}(x), T_x) \tag{4}$$

---

[2]dividing the unit interval into bins and taking a sample from each bin.

where PosEnc is the positional encoding from NeRF [38] and T_x are the features corresponding to x gathered from T by projecting x to each of the axis-aligned planes and taking the Hadamard product of the three resultant vectors [16]. We initialize the ReLU MLP_{SDF} as in SAL [4] to ensure an approximately spherical SDF in the early part of training. Additionally note that β varies spatially in the volume unlike [45, 72], allowing us to regularize its values later on.

Using Eq. 1, we transform s and β into opacity σ. We can now predict the radiance with a separate MLP_c conditioned on the geometry features and viewing direction v as

$$c = \text{MLP}_c(\text{PosEnc}(v), f_{geo}). \quad (5)$$

Note that unlike EG3D [7], we condition radiance on the viewing direction, allowing a more expressive generator. Thus, given a triplane, we can render any pixel by computing σ and c for points along the ray to approximate the volumetric rendering integral as described in Sec. 3.

4.2. High-Resolution Proposal Network

We now have our mapping from a latent code z to a 3D NeRF representation. However, volumetric rendering of NeRFs at higher resolutions requires extremely large numbers of samples, and thus both memory and time. Instead of naive dense sampling at a high resolution, we propose to leverage low-resolution renderings to cheaply probe the 3D representation (visualized on the left of Fig. 3) for the creation of proposal distributions at high-resolution. Given a target camera and triplane T, we first trace 192 coarse samples at low-resolution (128×128) to compute a low-resolution RGB image $I_{128} \in \mathbb{R}^{3 \times 128 \times 128}$ and a tensor of weights $P_{128} \in \mathbb{R}^{192 \times 128 \times 128}$ (visualized after low-resolution rendering in Fig. 3). Each 192-dimensional vector corresponds to a piecewise constant PDF with CDF Φ as seen in Eq. 3.

Conditioned on the low-resolution probe, we predict a tensor of proposal volume rendering weights at the high-resolution (512×512):

$$\hat{P}_{512} = \text{Softmax}(\text{CNN}(P_{128}, I_{128})) \in \mathbb{R}^{192 \times 512 \times 512}, \quad (6)$$

where CNN is a lightweight network that up-samples the low-resolution weights, Softmax produces discrete distributions along each ray, and the ˆ denotes that this is an estimated quantity. This corresponds to the Proposal in Fig. 3 and the yellow distribution in Fig. 4. Note that allocating 192 samples at 128×128 is equivalent to allocating just 12 at 512×512.

4.3. Supervising the Proposal Network

Having described the input and output of our high-resolution proposal network, we now show its supervision. From the target camera, we can also trace 192 coarse samples at high resolution for a small 64×64 patch, giving us a ground truth tensor of volume rendering weights

$P_{\text{patch}} \in \mathbb{R}^{192 \times 64 \times 64}$. We then prepare this tensor for supervision by computing:

$$\bar{P}_{\text{patch}} = \text{Normalize}(\text{Suppress}(\text{Blur}(P_{\text{patch}}))) \quad (7)$$

where Blur applies a 1D Gaussian kernel to the input distributions, $\text{Suppress}(x) = x$ if $x \geq 5e-3$ and 0 otherwise, and Normalize is L1 normalization to create a valid distribution. This corresponds to the patch loss in Fig. 3 and the purple distribution in Fig. 4. These operations create less noisy distributions to facilitate accurate learning of the high-frequency integrand which may be undersampled in the coarse pass.

We can then compare the predicted and cleaned ground truth distributions with a cross-entropy loss:

$$L_{\text{sampler}} = \text{CrossEntropy}(\bar{P}_{\text{patch}}, \hat{P}_{\text{patch}}) \quad (8)$$

where $\hat{P}_{\text{patch}}$ is the corresponding patch of weights in $\hat{P}_{512}$ and CrossEntropy denotes the average cross-entropy between all pairs of pixelwise discrete distributions; for each pair $(\bar{p}, \hat{p})$, we compute $\sum -\bar{p}_j \log \hat{p}_j$. Since we only need to compute this supervision for a small patch, the overhead of sampler training is not significant.

4.4. Robustly Sampling from the Proposal Network

Having shown how to train and predict high-resolution distributions, we now overview how to *sample* the resultant proposals. As seen in Fig. 4, the proposals are often slightly off; this is due to the high frequency nature of the underlying integrand in blue.

In order to utilize the information from the sampler, we propose to filter predicted PDFs for better estimation. Specifically, for each discrete predicted PDF $\hat{p}$, we compute the smallest set of bins whose probability exceeds a threshold $\tau = 0.98$: We find the smallest subset $I \subseteq \{1, 2, ..., 192\}$ such that

$$\sum_{i \in I} \hat{p}_i \geq \tau. \quad (9)$$

This operation resembles nucleus sampling in NLP [23]. We define our sampling PDF q with probability $q_i = \frac{1}{|I|}$ if $i \in I$ and 0 otherwise (the green distribution in Fig. 4).

For each PDF q, we compute its CDF Φ and perform stratified inverse transform sampling to create the samples (illustrated as adaptive sampling near the surface in Fig. 3). We refer to these samples as robust stratified samples. In practice, on top of the 12 samples from the coarse probe (for the high-resolution image), we take an additional 18 samples per pixel adaptively based on the variance of the predicted distributions. The details are given in the supplement.

4.5. Regularization for High-Resolution Training

In order to render accurately under low sample budget per ray, we desire the surface to be tight, i.e., the set of points
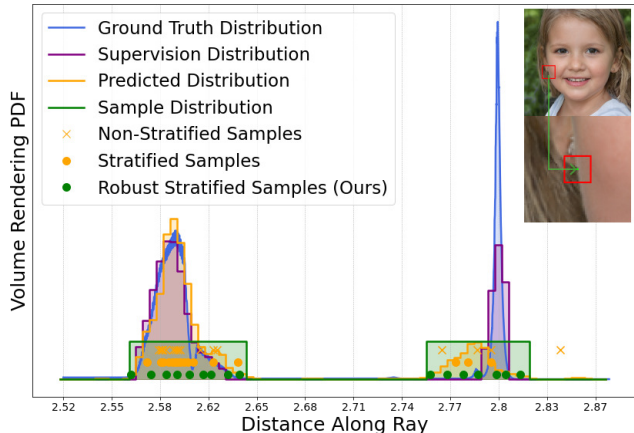
Figure 4. We visualize the volume rendering PDFs for the green pixel in the images on the right along with sampling methods. The ground truth distribution in blue is bimodal due to the discontinuous depth. Without stratification, the samples from the predicted yellow PDF completely miss the second mode. Stratification reduces the variance, yet also misses the second mode. Our robust stratified samples hit both modes despite the inaccurate predictions. The supervision PDF is visualized in purple as well.

along the ray, which contribute to the accumulated radiance to be small. To accomplish this, we introduce a regularization for the spatially-varying β values. Replacing the c_i with the intermediate β_i in Eq. 2, we can volume render an image of scalars, $B \in \mathbb{R}^{512 \times 512}$ (seen on the right of Fig. 3). During training, we regularize B to be small so that the surface tightens:

$$L_{\text{surface}} = \sum_{hw} (B_{hw} - B_{\text{target}})^2, \qquad (10)$$

where B_{target} is a scalar quantity annealed towards $\epsilon > 0$ during optimization. Note that this results in significantly more pronounced and smooth geometry (see Fig. 7).

During training with the proposal network, the rendering networks only receive gradients for points very near the surface (which have high density). We find that this can lead to undesirable density growing from the background into the foreground. In order to combat this, we leverage the low-resolution information from I_{128}, which is computed for uniform samples along the ray, and thus not concentrated at the surface. Considering the SDF values intermediately computed for rendering, $S = S_{128} \in \mathbb{R}^{192 \times 128 \times 128}$, we enforce the SDF decision boundary by minimizing the SDF likelihood under a Laplacian distribution similar to [45, 51]:

$$L_{\text{dec}} = \sum_{zhw} \exp \left(-2|S_{zhw}| \right). \qquad (11)$$

4.6. The Training Pipeline

In order to begin making use of the learned sampler for high-resolution rendering, we need a good NeRF representation from which to train it. Concurrently, we also need a

good sampler in order to allow NeRF to render at 512×512, the input resolution to the discriminator D.

To solve this issue, in the early stages of training, we first learn a low-resolution (e.g., 64×64) 3D GAN through the standard NeRF sampling techniques. We bilinearly upsample our low-resolution renderings to 512×512 and blur the real images to the same level. After converging at the lower resolution, we introduce the sampler training (subsection 4.3). Concretely, we not only render low-resolution images with standard sampling, but also render sampler inputs P_{128} and supervision patches P_{patch}. This results in a good initialization for the sampler.

Having learned an initial low-resolution 3D GAN and high-resolution sampler, we transition to rendering with the sampler predictions. The high-resolution proposals $\hat{P}_{512}$ are downsampled to the current rendering resolution, which is progressively increased to the full 512×512 resolution during training. After introducing all losses, we optimize the parameters of the generator G to minimize the following:

$$L = L_{\text{adv}} + \lambda_{\text{sampler}} L_{\text{sampler}} + \lambda_{\text{surface}} L_{\text{surface}} + \lambda_{\text{dec}} L_{\text{dec}} \quad (12)$$

where L_{adv} is the standard GAN loss [18] Softplus$(-D(G(z))$. Note that we do not enforce the Eikonal constraint as we did not see a benefit. The discriminator D is trained with R1 gradient regularization [37] whose weight is adapted as the resolution changes. Generator and discriminator pose conditioning follow EG3D [7]. The details of all hyperparameters and schedules are presented in the supplementary material.

5. Results

Datasets. We benchmark on two standard datasets for 3D GANs: FFHQ [25] and AFHQv2 Cats [11, 28] both at resolution 512×512. We use the camera parameters extracted by EG3D [7] for conditioning and rendering. Our AFHQ model is finetuned from our FFHQ model using adaptive data augmentation [26]. For more results, please see the accompanying video.

5.1. Comparisons

Baselines. We compare our methods against state-of-the-art 3D GAN methods including those that use low-resolution neural rendering and 2D post-processing CNN super resolution: EG3D [7], MVCGAN [74], and StyleSDF [45]; and methods that operate entirely based on neural rendering: Mimic3D [10], Epigraf [53], GMPI [75], and GramHD [67].

Qualitative results Fig. 5 shows the curated samples generated by our method with FFHQ and AFHQ, demonstrating photorealistic rendering as well as high-resolution details that align with the 2D images. Fig 6 provides qualitative comparisons to baselines. EG3D shows significant artifacts and cannot resolve high-resolution geometry since it performs neural rendering at only 128×128. Mimic3D
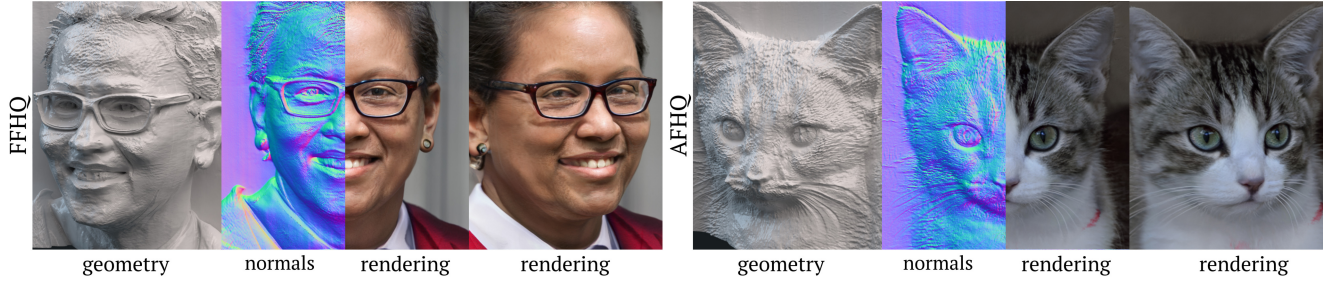
Figure 5. Curated samples on FFHQ and AFHQ. Our method can resolve high-fidelity geometry (e.g., eyeglasses) and fine-grained details (e.g., cat's fur) as seen in the geometry and normal map.
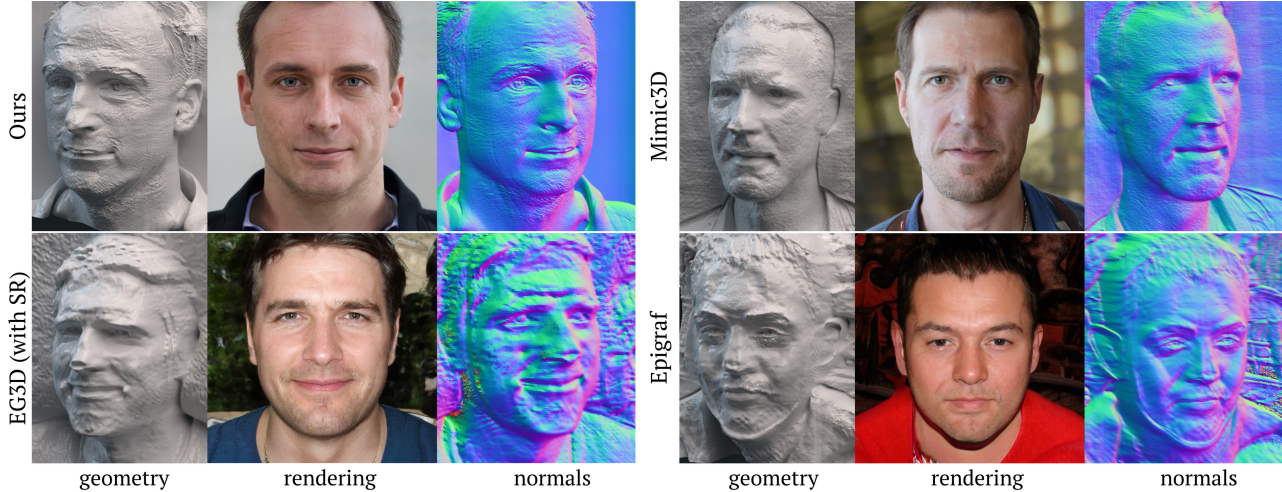

Figure 6. Qualitative comparisons on FFHQ with EG3D [7], Mimic3D [10] and Epigraf [53]. EG3D performs neural rendering at resolution 128×128 and relies on $4\times$ super resolution to generate images. On the right, Mimic3D and Epigraf directly generate the image via neural rendering. While all other baselines use up to 192 dense depth samples per ray, our method can operate at 30 samples per ray.
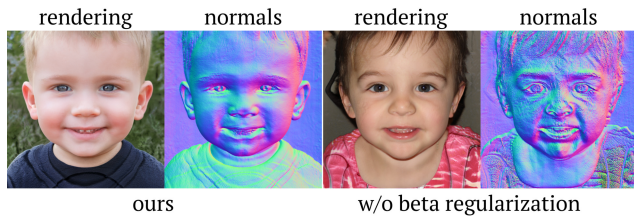

Figure 7. Ablation study on the effect of beta regularization.

and Epigraf render all pixels with neural rendering, but the patch-based nature of these methods harm the overall 3D geometry (e.g., distorted face and missing ear). Our method provides both high-fidelity 3D shape (e.g., well-defined ears and isolated clothing collar) and high-resolution details.

Quantitative results. Tabs. 1 and 2 provide quantitative comparisons against baselines. We measure the image quality with Fréchet Inception Distance (FID) [22]. We assess the quality of the learned geometry with a face-specific Normal FID (FID-N) [15]. We render 10.79k normal maps from the NPHM [17] dataset by solving for the approximate alignment between the average FFHQ 2D landmarks

and the provided 3D landmarks. Examples are given in the supplement. For each mesh, we render two views with approximately 20 degrees of pitch variation relative to the front of the face. These views are processed to remove the background with facer [13]. For each baseline, we render 10k normal maps and remove their background using the predicted mask from facer on the rendered image. We compute the FID between the two sets of images. Finally, we also evaluate the flatness of the learned representations with non-flatness score (NFS) [54].

Our results show the state-of-the-art image quality among the methods that operate only with neural rendering, while achieving FID comparable to the state-of-the-art SR-based method, EG3D. Our geometry quality outperforms all existing methods as indicated by our state-of-the-art FID-N. Additionally, our high NFS scores show the 3D aspect of our geometry. However, since NFS simply measures the variations of depth as a measure of non-flatness, it does not quantify the quality of geometry above a certain threshold.

5.2. Ablation Study

Without our surface tightness regularization (Eq. 10), the SDF surface may get fuzzy, resulting in a less clean surface

	Method	FID ↓	FID-N↓	NFS↑
FFHQ-512 w/o SR	Epigraf [7]	9.92[†]	67.33	**33.95**
	Mimic3D [10]	5.37	64.97	16.76
	GRAM-HD [67]	12.2[†*]	-	-
	Ours	**4.97**	**60.76**	29.35
w SR	EG3D [7]	**4.70**	63.02	17.54
	StyleSDF [45]	11.19[†]	87.42	**22.75**
	MVCGAN [74]	13.4[†*]	-	-

Table 1. Quantitative comparison with baselines with and without super resolution (SR) on the FFHQ dataset. [†] as reported in the previous works. [*] indicates FID evaluated on 20k images.

	Method	FID ↓	NFS↑
AFHQ-512 w/o SR	GRAM-HD [67]	7.67[†]	-
	GMPI [45]	7.79[†]	-
	Mimic3D [10]	4.29	12.67
	Ours	**4.23**	**21.89**
w SR	EG3D [7]	**2.77**	14.14
	StyleSDF [45]	7.91	**33.89**

Table 2. Quantitative results on AFHQv2 Cats. [†] as reported in the previous works.

Method	FID ↓	FID-N ↓	NFS ↑
- Learned Sampler	38.29	93.88	**30.95**
- Stratification	5.60	86.02	5.97
- Robust Sampling	5.67	60.78	24.79
- Beta Regularization	5.27	64.25	28.88
Ours	**4.97**	**60.76**	29.35

Table 3. Ablation study.

Figure 8. Qualitative comparisons for ablation study.

(see Fig. 7) and worse geometry scores (see Tab. 3). Without our sampler or stratification during sampling, the model cannot learn meaningful 3D geometry with limited depth budgets, creating degenerated 3D geometry as can be seen in Fig. 8 and significantly worse FID-N. Without our robust sampling strategy, the sampling becomes more susceptible to slight errors in the sampler due to the high-frequency nature of the PDF (see Fig. 4), resulting in a noticeable drop in FID and occasionally creating floater geometry artifacts (see Fig. 8), while geometry scores remain similar.

6. Discussion

Limitations and future work. While our method demonstrates significant improvements in 3D geometry generation, it may still exhibit artifacts such as dents in the presence of specularities, and cannot handle transparent objects such as lenses well. Future work may incorporate more advanced material formulations [5] and surface norm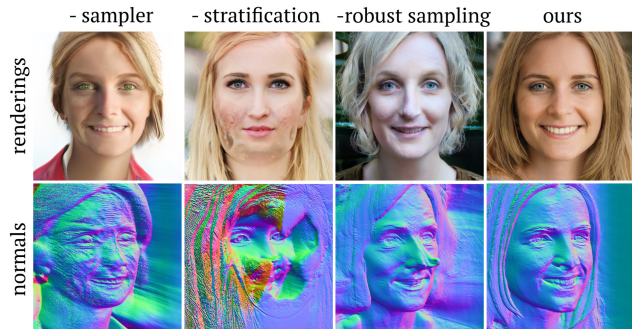al regularization [60]. While 3D GANs can learn 3D represen-tations from single-view image collections such as FFHQ and AFHQ with casual camera labels [7], the frontal bias and inaccurate labels can result in geometry artifacts, especially on the side of the faces. Fruitful future directions may include training 3D GANs with large-scale Internet data as well as incorporating a more advanced form of regularization [47] and auto-camera calibration [3] to extend the generations to 360 degrees. Finally, our sampling-based acceleration method may be applied to other NeRFs.

Ethical considerations. While existing methods have demonstrated effective capabilities in detecting unseen GANs [12], our contribution may remove certain characteristics from generated images, potentially making the task of detection more challenging. Viable solutions include the authentication of synthetic media [1, 2, 48].

Conclusion. We proposed a sampler-based method to accelerate 3D GANs to resolve 3D representations at the native resolution of 2D data, creating strictly multi-view-consistent images as well as highly detailed 3D geometry learned from a collection of in-the-wild 2D images. We believe our work opens up new possibilities for generating high-quality 3D models and synthetic data that capture in-the-wild variations and for enabling new applications such as conditional view synthesis.

Acknowledgements

References

[1] Coalition for content provenance and authenticity. https://c2pa.org/. 8

[2] Content authenticity initiative. https://contentauthenticity.org/. 8

[3] Sizhe An, Hongyi Xu, Yichun Shi, Guoxian Song, Umit Y. Ogras, and Linjie Luo. Panohead: Geometry-aware 3d full-head synthesis in 360deg. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 8

[4] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2565–2574, 2020. 5

[5] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P.A. Lensch. Nerd: Neural reflectance decomposition from image collections. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 8

[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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[7] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[8] Eric R. Chan, Koki Nagano, Matthew A. Chan, Alexander W. Bergman, Jeong Joon Park, Axel Levy, Miika Aittala, Shalini De Mello, Tero Karras, and Gordon Wetzstein. GeNVS: Generative novel view synthesis with 3D-aware diffusion models. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 3

[9] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*, 2023. 3

[10] Xingyu Chen, Yu Deng, and Baoyuan Wang. Mimic3d: Thriving 3d-aware gans via 3d-to-2d imitation. *arXiv preprint arXiv:2303.09036*, 2023. 2, 3, 6, 7, 8

[11] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 6

[12] Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023. 8

[13] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5203–5212, 2020. 7

[14] Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. Gram: Generative radiance manifolds for 3d-aware image generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[15] Zijian Dong, Xu Chen, Jinlong Yang, Michael J Black, Otmar Hilliges, and Andreas Geiger. Ag3d: Learning to generate 3d avatars from 2d image collections. *arXiv preprint arXiv:2305.02312*, 2023. 7

[16] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023. 5

[17] Simon Giebenhain, Tobias Kirschstein, Markos Georgopoulos, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Learning neural parametric head models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21003–21012, 2023. 7

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014. 2, 6

[19] 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. 2, 3

[20] Jiatao Gu, Alex Trevithick, Kai-En Lin, Joshua M Susskind, Christian Theobalt, Lingjie Liu, and Ravi Ramamoorthi. Nerfdiff: Single-image view synthesis with nerf-guided distillation from 3d-aware diffusion. In *International Conference on Machine Learning*, pages 11808–11826. PMLR, 2023. 3

[21] Kunal Gupta, Miloš Hašan, Zexiang Xu, Fujun Luan, Kulyan Sunkavalli, Xin Sun, Manmohan Chandraker, and Sai Bi. Mcnerf: Monte carlo rendering and denoising for real-time nerfs. In *ACM SIGGRAPH Asia 2023 Conference Proceedings*, 2023. 3

[22] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 7

[23] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019. 5

[24] Animesh Karnewar, Andrea Vedaldi, David Novotny, and Niloy Mitra. Holodiffusion: Training a 3D diffusion model using 2D images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[25] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 4, 6

[26] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 6

[27] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[28] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 6

[29] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (ToG)*, 2023. 3

[30] Andreas Kurz, Thomas Neff, Zhaoyang Lv, Michael Zollhöfer, and Markus Steinberger. Adanerf: Adaptive sampling for real-time rendering of neural radiance fields. 2022. 3

[31] Ruilong Li, Hang Gao, Matthew Tancik, and Angjoo Kanazawa. Nerfacc: Efficient sampling accelerates nerfs. *arXiv preprint arXiv:2305.04966*, 2023. 3

[32] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[33] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 3

[34] Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance fields for interactive free-viewpoint video. In *ACM Transactions on Graphics (SIGGRAPH ASIA)*, 2022. 3

[35] David B Lindell, Julien NP Martel, and Gordon Wetzstein. AutoInt: Automatic integration for fast neural volume rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[36] N. Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 1995. 4

[37] 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. 6

[38] 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 (ECCV)*, 2020. 2, 3, 4, 5

[39] Don P Mitchell. Consequences of stratified sampling in graphics. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 277–280, 1996. 4

[40] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 3

[41] Thomas Neff, Pascal Stadlbauer, Mathias Parger, Andreas Kurz, Joerg H. Mueller, Chakravarty R. Alla Chaitanya, Anton S. Kaplanyan, and Markus Steinberger. DONeRF: Towards Real-Time Rendering of Compact Neural Radiance Fields using Depth Oracle Networks. *Computer Graphics Forum*, 2021. 3

[42] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts, 2022. 3

[43] Michael Niemeyer and Andreas Geiger. GIRAFFE: Representing scenes as compositional generative neural feature fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3

[44] Michael Oechsle, Songyou Peng, and Andreas Geiger. UNISURF: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[45] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3, 4, 5, 6, 8

[46] M. Piala and R. Clark. Terminerf: Ray termination prediction for efficient neural rendering. In *International Conference on 3D Vision (3DV)*, 2021. 3

[47] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *International Conference on Learning Representations (ICLR)*, 2022. 3, 8

[48] Ekta Prashnani, Koki Nagano, Shalini De Mello, David Luebke, and Orazio Gallo. Avatar fingerprinting for authorized use of synthetic talking-head videos. *arXiv preprint arXiv:2305.03713*, 2023. 8

[49] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. GRAF: Generative radiance fields for 3D-aware image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 2

[50] Katja Schwarz, Axel Sauer, Michael Niemeyer, Yiyi Liao, and Andreas Geiger. Voxgraf: Fast 3d-aware image synthesis with sparse voxel grids. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 3

[51] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 6

[52] Vincent Sitzmann, Semon Rezchikov, Bill Freeman, Josh Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. *Advances in Neural Information Processing Systems*, 34:19313–19325, 2021. 3

[53] Ivan Skorokhodov, Sergey Tulyakov, Yiqun Wang, and Peter Wonka. Epigraf: Rethinking training of 3d gans. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2, 3, 6, 7

[54] Ivan Skorokhodov, Aliaksandr Siarohin, Yinghao Xu, Jian Ren, Hsin-Ying Lee, Peter Wonka, and Sergey Tulyakov. 3d generation on imagenet. *arXiv preprint arXiv:2303.01416*, 2023. 7

[55] Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Viewset diffusion: (0-)image-conditioned 3D generative models from 2D data. In *ICCV*, 2023. 3

[56] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3D shapes. In

IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021. 3

[57] Towaki Takikawa, Thomas Müller, Merlin Nimier-David, Alex Evans, Sanja Fidler, Alec Jacobson, and Alexander Keller. Compact neural graphics primitives with learned hash probing. In *ACM SIGGRAPH Asia 2023 Conference Proceedings*, 2023. 3

[58] Ayush Tewari, Tianwei Yin, George Cazenavette, Semon Rezchikov, Joshua B. Tenenbaum, Frédo Durand, William T. Freeman, and Vincent Sitzmann. Diffusion with forward models: Solving stochastic inverse problems without direct supervision. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 3

[59] Alex Trevithick, Matthew Chan, Michael Stengel, Eric R. Chan, Chao Liu, Zhiding Yu, Sameh Khamis, Manmohan Chandraker, Ravi Ramamoorthi, and Koki Nagano. Real-time radiance fields for single-image portrait view synthesis. In *ACM Transactions on Graphics (SIGGRAPH)*, 2023. 2

[60] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 8

[61] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12619–12629, 2023. 3

[62] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 3

[63] Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, and Baining Guo. Rodin: A generative model for sculpting 3d digital avatars using diffusion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[64] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 3

[65] Zian Wang, Tiancheng Shen, Merlin Nimier-David, Nicholas Sharp, Jun Gao, Alexander Keller, Sanja Fidler, Thomas Müller, and Zan Gojcic. Adaptive shells for efficient neural radiance field rendering. In *ACM Transactions on Graphics (SIGGRAPH ASIA)*. 3

[66] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023. 3

[67] Jianfeng Xiang, Jiaolong Yang, Yu Deng, and Xin Tong. Gram-hd: 3d-consistent image generation at high resolution with generative radiance manifolds. *arXiv preprint arXiv:2206.07255*, 2022. 3, 6, 8

[68] Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in

visual computing and beyond. In *Computer Graphics Forum*. Wiley Online Library, 2022. 2

[69] Yinghao Xu, Sida Peng, Ceyuan Yang, Yujun Shen, and Bolei Zhou. 3d-aware image synthesis via learning structural and textural representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[70] Yang Xue, Yuheng Li, Krishna Kumar Singh, and Yong Jae Lee. Giraffe hd: A high-resolution 3d-aware generative model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3

[71] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Ronen Basri, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 3

[72] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 3, 4, 5

[73] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. PlenOctrees for real-time rendering of neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[74] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 6, 8

[75] Xiaoming Zhao, Fangchang Ma, David Güera, Zhile Ren, Alexander G. Schwing, and Alex Colburn. Generative multiplane images: Making a 2d gan 3d-aware. In *European Conference on Computer Vision (ECCV)*, 2022. 3, 6

[76] 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. 2