

DENOISING DIFFUSION VIA IMAGE-BASED RENDERING

Titas Anciukevičius^{1,2} **Fabian Manhardt**² **Federico Tombari**^{2,3} **Paul Henderson**⁴

¹ University of Edinburgh ² Google ³ Technical University of Munich ⁴ University of Glasgow
<https://anciukevicius.github.io/generative-image-based-rendering>

ABSTRACT

Generating 3D scenes is a challenging open problem, which requires synthesizing plausible content that is fully consistent in 3D space. While recent methods such as neural radiance fields excel at view synthesis and 3D reconstruction, they cannot synthesize plausible details in unobserved regions since they lack a generative capability. Conversely, existing generative methods are typically not capable of reconstructing detailed, large-scale scenes in the wild, as they use limited-capacity 3D scene representations, require aligned camera poses, or rely on additional regularizers. In this work, we introduce the first diffusion model able to perform fast, detailed reconstruction and generation of real-world 3D scenes. To achieve this, we make three contributions. First, we introduce a new neural scene representation, IB-planes, that can efficiently and accurately represent large 3D scenes, dynamically allocating more capacity as needed to capture details visible in each image. Second, we propose a denoising-diffusion framework to learn a prior over this novel 3D scene representation, using only 2D images without the need for any additional supervision signal such as masks or depths. This supports 3D reconstruction and generation in a unified architecture. Third, we develop a principled approach to avoid trivial 3D solutions when integrating the image-based rendering with the diffusion model, by dropping out representations of some images. We evaluate the model on several challenging datasets of real and synthetic images, and demonstrate superior results on generation, novel view synthesis and 3D reconstruction.

1 INTRODUCTION

Generative models of the 3D world learnt from 2D images are powerful tools that enable synthesising 3D content without expensive manual creation of 3D assets. They are also crucial for 3D reconstruction from sparse images. In particular, classical 3D reconstruction techniques like multi-view stereo (Seitz et al., 2006a; Schönberger et al., 2016) and more recent approaches like NeRFs (Mildenhall et al., 2020) can reconstruct a 3D scene from a dense set of images (typically at least 20). However, they are not able to reconstruct regions that are not observed in any of the input images. Even methods like PixelNeRF (Yu et al., 2021) that are designed to generalise across scenes still fail to render plausible details in unobserved regions, typically producing blurry outputs. To mitigate this issue, it is necessary to estimate a *posterior distribution* on 3D scenes, conditioned on one or more images. The posterior distribution assigns high probability to scenes that align with the content in the images, and that are also realistic in unobserved areas. Subsequently, this allows us to *sample* diverse plausible scenes from the posterior, instead of predicting a blurred *average* over all possible scenes.

Despite the importance of the task, so far generative models of real-world 3D scenes have remained elusive due to three challenges. First, real-world scenes are often large, or even unbounded, making it difficult to define a scene representation that can express the details that may be visible, yet also enables learning a generative model. For representations that do scale well, it is typically challenging to learn a prior over them (Müller et al., 2022; Barron et al., 2021), since their representation of 3D structure lacks generality across different spatial locations and scenes. Although some representations such as 3D voxels (Peng et al., 2020) make it simple to learn a prior as they interpret features consistently across different locations and scenes, these methods only represent a bounded 3D volume and allocate modelling capacity uniformly across a finite grid, regardless of the scene content.

A second challenge is that large datasets of real-world 3D scenes are scarce, since they are time-consuming and expensive to obtain (Müller et al., 2022). Thus, some methods aim to build a generative model of 3D scenes using only 2D images for training. While achieving great results for

the task of 3D generation, all these methods exhibit several limitations. First, some works rely on large-scale datasets where all objects are placed in a canonical pose (Anciukevičius et al., 2023). This is possible when training on synthetic, object-centric datasets, but that does not allow generating realistic scenes. Indeed, for real-world scenes, it is very difficult to define a single canonical frame of reference and align all scenes to this. Other works instead do not require canonicalized objects, but still can only operate on object-centric data. Moreover, commonly these approaches even require object masks, as they leverage bounded scene representations such as tri-planes, that only work within a predefined 3D volume. This again significantly restricts their generation capabilities, as these methods can only synthesize isolated 3D objects instead of complete scenes.

A third challenge is that it is difficult to sample from the true posterior distribution over real scenes with unbounded volumes, as opposed to a less-expressive marginal distribution. Existing approaches for unbounded 3D scene sampling commonly follow an “infer, fuse, render, and repeat” paradigm (Wiles et al., 2020). These sample parts of the scene visible in the ‘next’ camera view frustum conditioned on a small marginal observation of the current 3D scene (features or pixels of the scene projected into that image). However, they do not use information from *all* previously seen or generated images to predict a camera view frustum consistent with the complete scene.

In this work we propose, the **first denoising diffusion model that can generate and reconstruct large-scale and detailed 3D scenes**. To achieve this, we make the following technical contributions that respectively address each of the challenges above:

1. We introduce a new neural representation for unbounded 3D scenes, *IB-planes*, which increases expressiveness versus prior image-base rendering representations, by letting the model incorporate information from multiple images, and by adding additional depth and polar features.
2. We introduce a joint multi-view denoising framework incorporating a latent 3D scene. It supports unconditional generation and reconstruction of 3D from varying numbers of images; in both cases it samples from a true joint distribution over full 3D scenes, rather than a less-expressive marginal distribution.
3. We present the first principled approach for integrating image-based rendering into diffusion models: we drop out parts of the image-based scene representation corresponding to the view being rendered to prevent trivial 3D solutions, but introduce a cross-view-attentive architecture that enables the noise from all images to influence the latent 3D scene.

We evaluate our method on four challenging datasets of multi-view images, including CO3D (Reizenstein et al., 2021) and MVIgNet (Yu et al., 2023b). We show that our model *GIBR* (*Generative Image-Based Rendering*) learns a strong prior over complex 3D scenes, and enables generating plausible 3D reconstructions given one or many images. It outputs explicit representations of 3D scenes, that can be rendered at resolutions up to 1024^2 .

2 RELATED WORK

Traditional 3D reconstruction methods output scenes represented as meshes, voxels, or point-clouds (Schönberger & Frahm, 2016; Seitz et al., 2006b; Häne et al., 2013). Recently however, *neural fields* (Xie et al., 2022; Mildenhall et al., 2020) have become the dominant representation. These approaches represent a scene as a function mapping position to density and color; the scene is queried and rendered using volumetric ray marching (Max, 1995). That function may be a generic neural network (Park et al., 2019; Mildenhall et al., 2020; Barron et al., 2021), or a specifically-designed function (Peng et al., 2020; Fridovich-Keil et al., 2023b; Müller et al., 2022; Li et al., 2023; Chen et al., 2022; Xu et al., 2022) to improve performance. Due to their continuous nature, such representations are easily learnt from a dense set of images (> 20), by gradient descent on a pixel reconstruction loss. Some works allow reconstruction from fewer views (Yu et al., 2021; Wang et al., 2021; Chen et al., 2021; Liu et al., 2022; Henzler et al., 2021; Wiles et al., 2020; Liu et al., 2022; Wu et al., 2023), often by unprojecting features or pixels from the images into 3D space. However, typically parts of the scene will be unobserved (e.g. far outside the camera view frustum), and thus ambiguous or uncertain given the observed images. The methods above make a single deterministic prediction, and cannot synthesise details in unobserved regions; instead, they produce a blurred prediction corresponding to the mean over all possible scenes, without an ability to sample individual, plausible scenes. Other approaches incorporate ad-hoc losses or regularizers from pretrained generative models to improve realism of unobserved regions (Zhou & Tulsiani, 2023; Yoo et al., 2023; Zou et al., 2023;

Melas-Kyriazi et al., 2023; Liu et al., 2023a; Wynn & Turmukhambetov, 2023; Niemeyer et al., 2022), however no work has achieved a principled approach to generate samples of large-scale 3D scenes given one or more real images as input. In particular, methods based on score-distillation regularise scenes towards high-probability regions, but do not truly sample the distribution.

Generative models allow sampling from complex, high-dimensional distributions (e.g. a distribution of 3D scenes). A myriad of generative models have been proposed for different domains, including GANs (Goodfellow et al., 2014; Arjovsky et al., 2017; Karras et al., 2019), VAEs (Kingma & Welling, 2014; Van Den Oord et al., 2017), and autoregressive models (Van Den Oord et al., 2016). Diffusion models (Sohl-Dickstein et al., 2015) have recently outperformed their counterparts in most domains, including images (Kingma et al., 2021; Dhariwal & Nichol, 2021; Ho et al., 2022; Saharia et al., 2022; Lugmayr et al., 2022; Jabri et al., 2022), video (Blattmann et al., 2023), and music (Huang et al., 2023). Numerous works have trained diffusion models directly on classical (Luo & Hu, 2021; Vahdat et al., 2022; Chen et al., 2023; Zhou et al., 2021; Hui et al., 2022; Li et al., 2022; Cheng et al., 2023) and neural (Müller et al., 2022; Bautista et al., 2022; Wang et al., 2022b; Kim et al., 2023; Shue et al., 2022; Gupta et al., 2023; Karnewar et al., 2023; Gu et al., 2023) 3D scene representations. However, diverse, high-quality generation has remained elusive since such models are restricted by the lack of suitable datasets of canonically-oriented and bounded 3D scenes. In contrast, we aim to learn a generative 3D model from in-the-wild dataset of images (i.e. that could be easily collected with a camera and COLMAP pose estimation), without assuming canonical orientations, bounding boxes, object segmentations.

To mitigate the lack of 3D data, others methods use pretrained generative models of 2D images to guide the optimization of a 3D scene (Jain et al., 2022; Poole et al., 2022; Höllein et al., 2023; Fridman et al., 2023; Wang et al., 2023; 2022a; Metzger et al., 2022; Lin et al., 2023; Shi et al., 2023). However, such approaches do not scale to large scenes, nor allow posterior sampling of 3D scenes given one or more images as conditioning. An alternative approach is to learn a density jointly over 2D images and their latent 3D representations; this allows them to be trained from widely-available 2D image datasets, yet still sample 3D scenes (Skorokhodov et al., 2023; Xiang et al., 2023; Shi et al., 2022). Initially based on GANs (Chan et al., 2022; Deng et al., 2022; Nguyen-Phuoc et al., 2020; 2019; Schwarz et al., 2020; Zhao et al., 2022; Devries et al., 2021) or VAEs (Anciukevičius et al., 2022; Kosiorek et al., 2021; Henderson & Lampert, 2020; Henderson et al., 2020), recently diffusion-based methods have achieved the most promising results. Notably, (Anciukevičius et al., 2023) showed that diffusion can also perform 3D reconstruction by inferring a latent 3D representation given an image (unlike GANs), yet also generates sharp, detailed 3D assets and images (unlike VAEs). However, (Anciukevičius et al., 2023; Szymanowicz et al., 2023) are limited to object-centric and canonically-aligned scenes, due to their use of canonically-placed voxel grids or triplanes as 3D representations. Other works therefore uses a pipeline of “infer, fuse, render, and repeat” (Wiles et al., 2020): the model generates the content visible in a camera view frustum conditioned on a rendering of the current scene at that frustum, then renders it to another viewpoint, and repeats. However, this only conditions on a marginal observation (since only the most recent view is seen, not the entire history of generated views nor an explicit 3D representation). Instead we aim to sample from a joint distribution of scenes. Moreover, they are slow to perform 3D reconstruction, e.g. concurrent work (Tewari et al., 2023) takes 2 hours, and do not support unconditional generation of 3D scenes nor conditional generation with arbitrary numbers of conditioning images.

Some works circumvent the difficulty of learning a 3D representation entirely by training conditional generative models to output images from novel viewpoints conditioned on one or more input images and a camera pose (Eslami et al., 2018; Kulhánek et al., 2022; Rombach et al., 2021; Du et al., 2023; Ren & Wang, 2022; Watson et al., 2022; Chan et al., 2023; Tseng et al., 2023; Liu et al., 2023b; Cai et al., 2022; Tang et al., 2023; Yu et al., 2023a). However, as such methods do not explicitly represent the underlying 3D scene, they cannot guarantee the resulting images depict a single consistent scene, and existing methods fail to generalize to camera poses far from the training distribution.

3 METHOD

Our goal is to build a generative 3D scene model that supports two tasks: (i) unconditional generation, (sampling 3D scenes a priori) (ii) 3D reconstruction (generation conditioned on one or more images). We aim to learn this model without 3D supervision by only assuming access to a dataset of multi-view images with relative camera poses (which can be easily obtained via structure-from-motion).

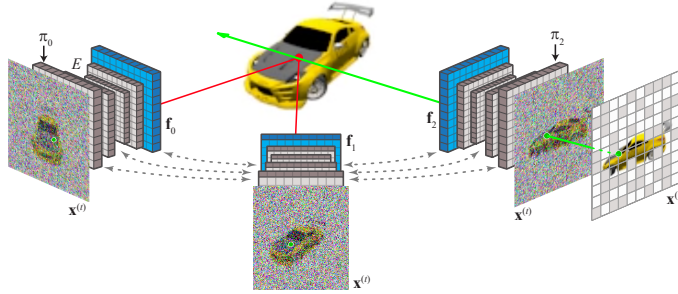


Figure 1: Our neural scene representation *IB-planes* defines 3D content using image-space features. Each camera π_v is associated with a feature-map \mathbf{f}_v (blue); together both parametrise a neural field that defines density and color for each 3D point p (red dot). We incorporate this representation in a diffusion model over multi-view images. At each denoising step, noisy images $\mathbf{x}^{(t)}$ are encoded by a U-Net E with cross-view attention (gray dashed arrows), that yields pixel-aligned features \mathbf{f}_v (blue). To render pixels of denoised images (only one $\mathbf{x}^{(0)}$ is shown for clarity), we use volumetric ray-marching (green arrow), decoding features unprojected (red lines) from the other viewpoints.

To this end, we first describe a novel image-based 3D scene representation that adapts its capacity to capture all the detail in a set of images, yet is suitable for learning a prior over (Sec. 3.1). This enables us to define a denoising diffusion model over multi-view images depicting real-world scenes, that builds and renders an explicit 3D representation of the latent scene at each denoising step (Sec. 3.2). This ensures the generated multi-view images depict a single, consistent 3D scene, and allows rendering the final scene efficiently from any viewpoint. We name our model *Generative Image-Based Rendering* (GIBR).

3.1 REPRESENTING 3D SCENES WITH IB-PLANES

We represent a 3D scene as a neural field (Mildenhall et al., 2020) – a function mapping world-space positions to a density (i.e. opacity) and color, which can be rendered using the standard emission-absorption method (Max, 1995). Inspired by recent success of image-based rendering (Lensch et al., 2003; Yu et al., 2021) and K-planes (Fridovich-Keil et al., 2023a), the density and color at each 3D point are defined via features placed in the view space of a set of images (Fig. 1). Specifically, we represent a scene by a set of 2D feature-maps $\{\mathbf{f}_v\}_{v=1}^V$ and corresponding poses $\{\pi_v\}_{v=1}^V$ for V cameras. These per-view feature-maps and poses parametrize a single neural field that defines the density and color at each point $p \in \mathbb{R}^3$ in 3D space. To calculate these, we project p into each camera view based on its pose π_v (which includes both extrinsics and intrinsics), finding the corresponding pixel-space location $\phi(p, \pi_v)$. Then, we extract the feature vector at that location in \mathbf{f}_v using bilinear interpolation, setting $f_v(p) = \mathbf{f}_v[\phi(p, \pi_v)]$.

Notably, unlike PixelNeRF (Yu et al., 2021), our IBR feature planes (which we name *IB-planes*) are output jointly by a U-Net that attends over multiple views. Hence, IB-planes are strictly more expressive than prior IBR approaches, such as PixelNeRF and IBRNet (Wang et al., 2021), that calculate features independently for each image. This is because the multi-view U-Net can arrange different IBR features for a viewpoint depending on other input images, and remove the depth ambiguity that is present when given only one image. On the other hand, unlike K-planes (Fridovich-Keil et al., 2023a), our IB-planes are placed in the camera view frusta to facilitate learning a generalizable model that maps images to scene representations. As a result, we can use a simple and fast max-pooling operation to fuse features, instead of needing a large, expensive feature-fusion model (e.g. IBRNet has a deep attention network over point features and nearby 3D points).

To ensure the scene geometry is well-defined outside the union of the camera view frusta, for each camera we also calculate a polar representation of the displacement of p relative to the camera’s center, and use the resulting angles to interpolate into a second feature map (with an equirectangular projection), giving a vector $f'_v(p)$. We concatenate the feature vectors $f_v(p)$ and $f'_v(p)$ with an embedding of the distance of p from the corresponding camera origin, and process this with an MLP to give a feature vector $f_v^*(p)$. We next max-pool these feature vectors across views, to give a single unified feature $f(p) = \max_v f_v^*(p)$ that fuses information from all views; the max is computed element-wise. Finally, this is mapped by an MLP to the density and RGB color at p .

3.2 MULTI-VIEW DENOISING DIFFUSION

We next describe our generative model of multi-view images, then discuss how we incorporate our scene representation into this to ensure 3D consistency while retaining expressiveness. We want to learn a generative model over sparse multi-view images \mathbf{x}^s drawn from some unknown distribution \mathcal{X} , where each $\mathbf{x}^s \in \mathbb{R}^{V \times H \times W \times 3}$ depicts a different scene, and consists of V RGB images each of size $W \times H$ (note that V may vary between scenes). Associated with each view x_v^s is a camera pose π_v^s , specified relative to x_0^s (i.e. we do *not* assume existence of a canonical coordinate system common to all scenes, unlike e.g. [Anciukevičius et al. \(2023\)](#) and [Chan et al. \(2022\)](#)). In the following description we omit the scene index s for clarity.

In order to define a generative model over multi-view images \mathbf{x} , we define forward (noising) and reverse (denoising) diffusion processes ([Ho et al., 2020](#)). The forward process is a sequence of stochastic transformations that progressively add Gaussian noise to the original pixels, resulting in a unit Gaussian sample over time. Formally, for a time step t and noise level β_t determined by a predefined noise schedule, the noisy multi-view image at diffusion time step t is:

$$\mathbf{x}^{(t)} = \sqrt{1 - \beta_t} \mathbf{x}^{(t-1)} + \sqrt{\beta_t} \epsilon^{(t)}, \quad \epsilon^{(t)} \sim \mathcal{N}(\mathbf{0}, I) \quad (1)$$

To sample from the original distribution \mathcal{X} , we learn a reverse process that reconstructs multi-view images from their noised versions. Specifically, we train a denoising function $\mu_\theta(\mathbf{x}^{(t)}, t)$ to predict the original multi-view image \mathbf{x} from the noisy image $\mathbf{x}^{(t)}$ and the diffusion step t (note we predict the image, not the noise as is common). To sample new multi-view images, we begin from a sample of pure Gaussian noise, and repeatedly apply μ_θ following the DDIM sampler of [Song et al. \(2020\)](#).

Typically diffusion models implement μ_θ as a neural network, often a U-Net ([Ronneberger et al., 2015](#)). This could be applied in our multi-view setting, provided we allow different views to exchange information, e.g. using a 3D U-Net, or cross-attention between the views. However, it does not *guarantee* that the resulting images are 3D-consistent, i.e. that the same 3D scene is visible in each view – the model must instead learn to approximate this, and often fails (see our ablation study). We next describe a denoiser μ_θ that ensures the views are 3D-consistent throughout the diffusion process.

3D-consistent denoising. To ensure 3D consistency of the multi-view images reconstructed during the diffusion process, and to enable access to a 3D model of the final scene, we incorporate an explicit intermediate 3D representation into the architecture of our multi-view denoiser μ_θ . During each denoising step, an encoder E estimates a single noise-free 3D scene $\{(\mathbf{f}_v, \pi_v)\}_{v=1}^V = E(\mathbf{x}^{(t)}, t)$ parametrized according to [Sec. 3.1](#) that incorporates information from all the views. The denoiser then renders this scene from each viewpoint to yield the denoised views, so we have

$$\mu_\theta(\mathbf{x}^{(t)}, t) = \text{render} \left(E(\mathbf{x}^{(t)}, t) \right). \quad (2)$$

Setwise multi-view encoder. The encoder $E(\mathbf{x}^{(t)}, t)$ calculates pixel-aligned features \mathbf{f}_v for each view $\mathbf{x}_v^{(t)}$ in $\mathbf{x}^{(t)}$ using a multi-view U-Net architecture. We adapt the U-Net architecture of ([Ho et al., 2020](#)), modifying the output layer yield features instead of RGB values. We also introduce attention between views, allowing them to exchange information. We replace each attention layer with a multi-headed linear attention ([Vaswani et al., 2017](#); [Katharopoulos et al., 2020](#)) that jointly attends across all feature locations in all views. Aside from these attention layers, the rest of the network processes each view independently; this is more computationally efficient than a full 3D CNN. It also avoids any undesirable inductive bias toward smoothness across adjacent views, which is important since we do not assume views have any particular spatial relation to each other. We also provide the encoder with a setwise embedding of the camera poses π_v , specified relative to some arbitrary view. We flatten the extrinsics and intrinsics matrices to vectors, pass them to small MLPs, and concatenate the results, to give a per-view relative pose embedding π_v^* . When encoding each view $\mathbf{x}_v^{(t)}$, we input the corresponding embedding π_v^* , and also the result of max-pooling the embeddings for other views. This is injected into the network similarly to the Fourier embedding of the timestep t , by concatenating it with the features at each layer. Importantly, our encoder architecture jointly reasons over all images in the scene; unlike autoregressive methods (e.g. [Wiles et al., 2020](#)), all information from all views is accounted for simultaneously to ensure the scene is coherent. Moreover, use of pooling operations in the encoder and the scene representation (feature fusion) to integrate information from different views ensures that it supports varying numbers of images.

Conditional generation. We can adapt this model to the conditional setting, where we are provided with one or more input views and must generate complete scenes. In this case, some views passed to

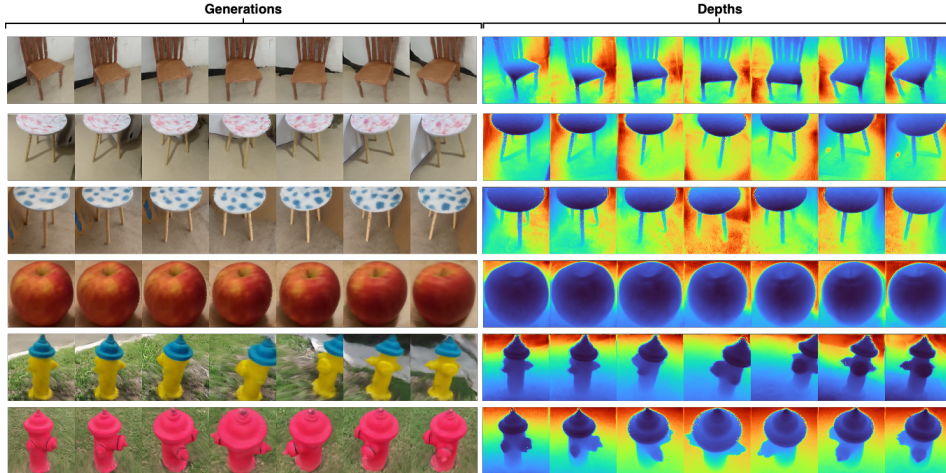


Figure 2: Samples generated by our method trained on MVImgNet (first three rows), CO3D (last three rows). Note that each multi-view image depicts a single coherent scene, with plausible appearance and detailed geometry. Please see the supplementary material for 1024×1024 video visualisations.

μ_θ as part of $\mathbf{x}^{(t)}$ are not noisy. The V views are therefore split into V_n noisy views, and V_c noise-free conditioning views. We indicate this to the model by passing a different t for each view, with $t = 0$ indicating a noise-free view. Each noisy view then encodes (in its noise) latent information about parts of the scene that are uncertain even given the noise-free conditioning views. We ensure there is at least one noisy view present, so the model always retains generative behavior. The image-based scene representation ensures there is a direct flow of information from noise at the pixels to corresponding points in the 3D scene, while the joint multi-view encoder means that latent information is correctly fused across different views, also incorporating information from the observed images.

3.3 TRAINING

Our model is trained to reconstruct multi-view images \mathbf{x} given their noised versions $\mathbf{x}^{(t)}$. We use an unweighted diffusion loss \mathcal{L} (Ho et al., 2020) with an L1 photometric reconstruction term:

$$\mathcal{L} = \mathbb{E}_{t, \mathbf{x}} \|\mathbf{x} - \mu_\theta(\mathbf{x}^{(t)}, t)\|_1 \quad (3)$$

We train our model end-to-end to minimize \mathcal{L} using Adam (Kingma & Ba, 2015). We vary V across different minibatches to ensure generality; to allow conditioning on varying numbers of images, we also vary the number V_c of noise-free views between zero and V . Training the model to reconstruct a large number of high-resolution images is computationally expensive since it requires a volumetric ray-marching for $V \times H \times W$ pixels. To overcome this, we approximate the loss (3) by only rendering a small fraction ($\approx 5\%$) of rays. This is still an unbiased estimate of \mathcal{L} , and has surprisingly minimal effect on the number of iterations until convergence, while greatly improving the wall-clock time and allowing us to go beyond prior works by training at 256×256 resolution.

3.4 DROPPING OUT NEURAL REPRESENTATIONS

One major challenge with 3D-aware generative models is that minimizing the loss does not necessarily force the model to accurately understand 3D. The model can instead produce a simple, uninformative pseudo-3D representation, such as a flat plane positioned directly in front of each camera, textured with a projection of the observed scene from that angle. Recent techniques have tried to address this by using various dataset-specific approaches, like requiring camera poses in a canonical frame of reference (Anciukevičius et al., 2023) (which is not possible for in-the-wild scenes). A naïve approach would be to use held-out views for supervision, but this falls short as they prevent the diffusion model from sampling interpretations of these heldout views, instead merely approximating the average observation, much like older non-generative techniques. Instead, we adopt a principled approach that ensures an expressive 3D representation with purely the diffusion loss (3), without any regularizers, heldout views or canonical camera poses. Specifically, we drop out the features \mathbf{f}_v from the v^{th} view when rendering to that same viewpoint. Note that this is not the same as *masking* some noises (as previous methods did), since we still allow latent information in the noise of the i^{th} view to flow to all other views' features and thus the scene itself. During inference, we include features from all views.

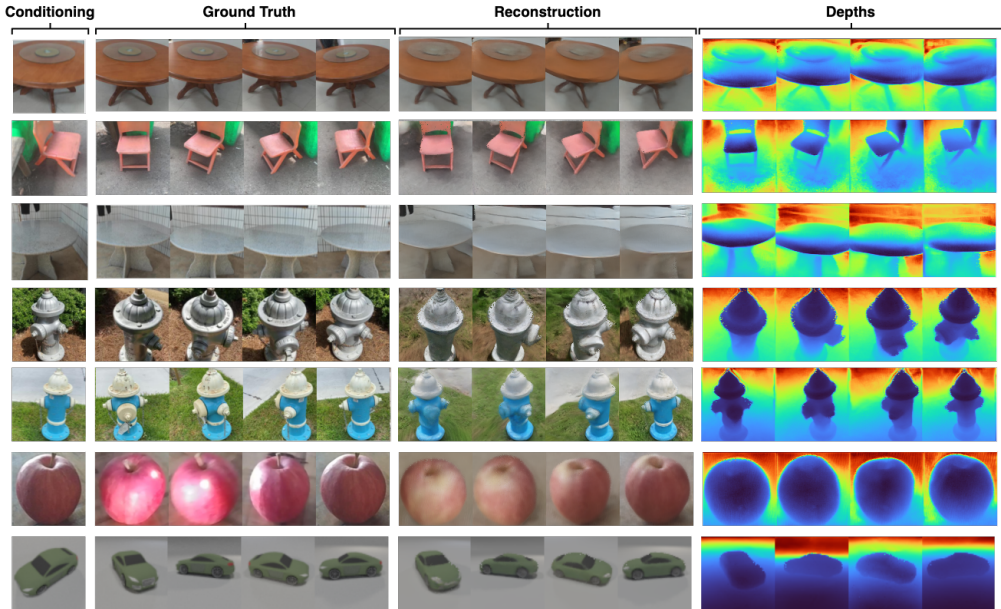


Figure 3: Results from our model on 3D reconstruction from a single image on MVImgNet (first 3 rows), CO3D (next 3 rows) and ShapeNet (last row). The leftmost column is the input; the next four show the ground-truth novel view images. The remaining columns show our model’s prediction from those viewpoints and the predicted depth-maps. Please see the supplementary videos for more results.

4 EXPERIMENTS

Datasets. We evaluate our approach on three datasets: (i) real-world chairs, tables and sofas from MVImgNet (Yu et al., 2023b); (ii) real-world hydrants, apples, sandwiches and teddybears from CO3D (Reizenstein et al., 2021); (iii) the renderings of ShapeNet (Chang et al., 2015) cars from (Anciukevičius et al., 2023). For CO3D, we train single-class models for hydrant and apple, and also a class-conditional model over the four classes; for MVImgNet we train one class-conditional model. Notably, CO3D and MVImgNet show large-scale indoor and outdoor scenes, including objects with fine details and textures. For all datasets, we only use the RGB images and relative camera poses – we do *not* use any masks or depths. During training, we randomly sample 6–8 views per scene. For MVImgNet and CO3D, the images are resized to 96×96 for most experiments and 256×256 for high-resolution runs (only supported by our method); for ShapeNet we use the original 64×64 . For CO3D, prior to resizing, we take a square crop centered on the ground-truth object mask; for MVImgNet, we take a center crop with size equal to the smaller dimension of the image.

Baselines. We compare to the most related diffusion method RenderDiffusion (Anciukevičius et al., 2023) and non-generative method PixelNeRF (Yu et al., 2021); the concurrent Viewset Diffusion (Szymanowicz et al., 2023); and the score-distillation method SparseFusion (Zhou & Tulsiani, 2023). Like ours, RenderDiffusion and Viewset Diffusion perform diffusion in image space. The former uses a triplane representation of 3D shapes and requires scenes to be placed in a canonical world-space, while the latter uses a fixed-size voxel grid. Thus, neither is able to adapt their capacity nor model very large scenes. Hence, we extend them to support our setting, and denote them as RenderDiffusion++, PixelNeRF++ and VSD*. Further details on how we extend them to our setting are in App. D.

4.1 GENERATIVE 3D RECONSTRUCTION

We first evaluate performance on 3D reconstruction from one or few images. We measure PSNR, SSIM and LPIPS between predicted and ground-truth images, and the rank-correlation of depths (DRC) (we use rank-correlation since absolute scale may differ between ground-truth and predicted scenes). Reconstruction from sparse images is ambiguous – there are many plausible completions of unobserved regions. We therefore follow other works on stochastic prediction (e.g. Denton & Fergus, 2018) and draw multiple (8) samples from the model, calculate the metrics for each, and take the best sample with respect to the ground-truth. For the diffusion-based methods, we render images and calculate metrics for two sets of viewpoints – the views in which the diffusion was performed (with

	Single-view reconstruction								Multi-view reconstruction			
	PSNR _D ↑	SSIM _D ↑	LPIPS _D ↓	DRC _D ↑	PSNR _H ↑	SSIM _H ↑	LPIPS _H ↓	DRC _H ↑	PSNR _D ↑	SSIM _D ↑	LPIPS _D ↓	DRC _D ↑
CO3D hydrant												
RenderDiff++	15.70	0.317	0.598	0.832	16.28	0.333	0.587	0.837	18.60	0.399	0.533	0.882
PixelNeRF++	15.06	0.278	0.615	0.527	–	–	–	–	16.86	0.366	0.545	0.595
Viewset Diffusion	13.18	0.144	0.714	–	13.50	0.149	0.718	–	–	–	–	–
SparseFusion	12.06	–	0.630	–	–	–	–	–	–	–	–	–
GIBR (ours)	16.07	0.329	0.456	0.821	17.12	0.403	0.449	0.829	20.22	0.571	0.283	0.882
CO3D apple												
RenderDiff++	16.71	0.601	0.475	0.708	17.20	0.608	0.464	0.730	18.97	0.638	0.427	0.648
PixelNeRF++	16.25	0.546	0.548	0.513	–	–	–	–	17.73	0.601	0.476	0.542
Viewset Diffusion	13.99	0.416	0.633	–	13.31	0.393	0.674	–	–	–	–	–
GIBR (ours)	18.09	0.616	0.396	0.739	18.92	0.647	0.372	0.743	21.04	0.712	0.296	0.746
CO3D multi-class												
RenderDiff++	15.94	0.314	0.686	0.836	16.52	0.324	0.676	0.843	17.81	0.356	0.643	0.848
PixelNeRF++	15.62	0.303	0.655	0.580	–	–	–	–	17.25	0.394	0.572	0.640
GIBR (ours)	16.70	0.360	0.481	0.863	17.90	0.434	0.465	0.872	21.54	0.634	0.281	0.898
ShapeNet car												
RenderDiff++	25.50	0.802	0.266	0.660	25.31	0.792	0.267	0.720	26.89	0.850	0.245	0.790
PixelNeRF++	26.81	0.860	0.218	0.889	–	–	–	–	25.69	0.848	0.226	0.827
Viewset Diffusion	28.00	0.871	0.167	–	26.06	0.817	0.227	–	–	–	–	–
GIBR (ours)	29.74	0.906	0.139	0.993	28.96	0.883	0.162	0.992	33.46	0.961	0.096	0.998
MVImgNet furniture												
RenderDiff++	17.37	0.468	0.622	–	18.11	0.483	0.610	–	18.44	0.487	0.601	–
PixelNeRF++	16.57	0.412	0.582	–	–	–	–	–	15.71	0.350	0.647	–
Viewset Diffusion	17.58	0.409	0.540	–	18.02	0.434	0.530	–	–	–	–	–
GIBR (ours)	18.54	0.518	0.414	–	19.89	0.590	0.369	–	22.09	0.730	0.284	–

Table 1: Results on 3D reconstruction from single and multiple images, for our method GIBR and baselines. Metrics suffixed D are calculated on the same views as we perform diffusion in; metrics suffixed with H are calculated in other, held-out views (except for PixelNeRF, which does not make this distinction). Note ground-truth depths are not available for MVImgNet, and Viewset Diffusion cannot perform reconstruction from six views. The SparseFusion result is from (Tewari et al., 2023).

	GIBR (ours)		RenderDiff++		VSD*	Generation		Single-view reconstruction			
	FID _D ↓	FID _H ↓	FID _D ↓	FID _H ↓	FID _D ↓	FID _D ↓	FID _H ↓	PSNR _D ↑	SSIM _D ↑	LPIPS _D ↓	DRC _D ↑
CO3D hydrant	91.9	118.1	185.4	182.9	217.0	183.8	194.7	15.10	0.320	0.636	0.785
CO3D apple	50.5	51.8	149.2	148.9	101.7	194.5	202.0	17.96	0.554	0.519	–
CO3D multi	121.5	123.4	201.6	201.2	–	(b)					
ShapeNet car	62.8	90.1	163.5	160.2	56.0						
MVImgNet	99.8	107.3	234.1	232.1	191.4						

Table 2: (a) Results on generation for our method and two baselines. (b) Results on generation and 3D reconstruction for our method on high-resolution images (256×256).

subscript D on the metric names), and a disjoint set of held-out viewpoints (subscript H). The latter show whether methods generate consistent 3D geometry that can be viewed from any angle.

Note that in App. A.2 we measure the impact of training our model with different numbers of views. Additional qualitative results are presented in App. A.3.

Reconstruction from a single image. We first evaluate reconstruction from one input image with unknown camera pose, meaning there is a high degree of uncertainty in the resulting scene, since much of it is unobserved. Quantitatively, our model GIBR out-performs both the recent generative 3D diffusion model RenderDiffusion (Anciukevičius et al., 2023), and the non-probabilistic PixelNeRF (Yu et al., 2021), across all datasets in terms of PSNR, SSIM, LPIPS and DRC (‘single-view reconstruction’ columns in Tab. 1). We attribute this to GIBR’s generative capabilities (in contrast to deterministic PixelNeRF that must make blurry, averaged predictions), and to its flexible image-based scene representation (in contrast to RenderDiffusion which relies on fixed-size triplanes). Qualitative results (Fig. 3) confirm that not only does our model successfully reconstruct sharp and visually convincing 3D scenes, but it also excels at generating plausible details in regions that are not visible in the input view. The depth-maps show that even fine details such as chair legs are accurately captured. In Tab. 2b we evaluate our model on higher resolution images than supported by prior works (256×256), showing that it retains competitive performance even in this more challenging setting, particularly on the multi-class MVImgNet dataset. Moreover, in the supplementary material, we show renderings of our reconstructed scenes at an even higher resolution (1024×1024), which is only possible as our IB-planes representation explicitly captures the latent 3D scene.

Reconstruction from multiple images. Next, we evaluate performance on 3D reconstruction from six views. We see (Tab. 1, right four columns) that GIBR successfully makes use of the additional information in the larger number of conditioning images to improve the quantitative results versus reconstruction from a single image. This is akin to single-scene overfitting methods such as NeRF (though still with fewer images than they typically require), but still leverages our multi-view denoising U-Net architecture to ensure the scene remains close to the learnt prior distribution. Qualitative results are shown in Fig. 4 in the appendix; we see that while our model makes use of its learnt prior to complete unobserved regions, it still faithfully integrates the detailed texture and geometry visible in all observed viewpoints to reconstruct a coherent scene.

4.2 UNCONDITIONAL GENERATION OF 3D SCENES

We now evaluate performance on unconditional generation of 3D scenes. We measure performance with two variants of Fréchet Inception Distance (Heusel et al., 2017). FID_D is calculated using renderings at the viewpoints at which diffusion was performed, i.e. the exact multi-view images output by the diffusion model. FID_H instead uses renderings of the generated 3D shapes from seven different viewpoints, verifying that the 2D diffusion process yields a valid 3D shape (not just plausible projections in the views where the diffusion was performed).

Our model demonstrates significant improvements over both baselines according to FID_D (Tab. 2a). Notably, our 3D generated scenes also look plausible from different viewpoints than those in which the model performed the denoising, as shown by the comparable values of FID_H and FID_D . Concurrent Viewset Diffusion (Szymanowicz et al., 2023) performs worse on CO3D and MVImgNet, due to its use of a finite grid of features to represent the scene, meaning it must trade off detail for scene size; however it is the top-performing method on ShapeNet (which is simpler since objects and cameras are placed in a canonical frame of reference). Qualitatively (Fig. 2), our model not only generates visually coherent 3D scenes due to its explicit 3D representation, but also exhibits convincing 3D geometry, as seen in the crisp depth maps. We attribute this in part to our lack of restrictive regularisers, and in part to our expressive 3D representation and multi-view U-Net architecture, which together ensure the latent pixel noise in image space is integrated into a coherent 3D scene during the diffusion process. Further qualitative results (including from the baselines) and ablations are given in App. A.

4.3 ABLATION EXPERIMENTS

We performed five ablation experiments to quantify the benefit of our key technical contributions and design decisions, showing decreased performance of our model (i) without dropout of representation described in Sec. 3.3; (ii) replacing our IB-planes representation (Sec. 3.1) with triplanes; (iii) without cross-view attention; (iv) replacing volumetric rendering with a black-box 2D CNN; (v) without polar features. We report results on CO3D hydrant in Tab. 3 and discuss them in detail in App. A.4.

	Generation		Single-view reconstruction				Multi-view reconstruction			
	$FID_D \downarrow$	$FID_H \downarrow$	PSNR $_D \uparrow$	SSIM $_D \uparrow$	LPIPS $_D \uparrow$	DRC $_D \uparrow$	PSNR $_D \uparrow$	SSIM $_D \uparrow$	LPIPS $_D \uparrow$	DRC $_D \uparrow$
No repr. drop.	58.9	266.5	15.47	0.279	0.450	0.586	19.73	0.497	0.311	0.700
No IBR	176.4	177.9	14.55	0.273	0.631	0.782	17.39	0.349	0.569	0.839
No cross-view attn.	98.0	126.1	14.91	0.288	0.482	0.808	19.50	0.545	0.307	0.871
No 3D	36.3	–	13.51	0.186	0.509	–	14.04	0.208	0.472	–
No polar features	113.2	126.5	16.27	0.345	0.482	0.747	20.46	0.587	0.292	0.854
Full model	91.9	118.1	16.07	0.329	0.456	0.821	20.22	0.571	0.283	0.882

Table 3: Ablation results for variants of our method on CO3D hydrant. See App. A.4 for more details.

5 CONCLUSION

We have introduced a new approach to 3D scene generation and reconstruction, that can be trained from multi-view images without 3D supervision. Our denoising diffusion model *GIBR* incorporates an explicit 3D representation of the latent scene at each denoising step, ensuring that the resulting multi-view images always depict a single consistent 3D scene. To enable this, we introduced a powerful new scene representation based on image features lifted into 3D space, that can adapt its capacity according to the parts of the scene that are imaged, ensuring details are captured faithfully.

Limitations. While this work makes progress towards unsupervised learning of 3D generative models from in-the-wild images, it still assumes each scene is static. Also, even with approximation of loss (3), our model is slower to train than 2D diffusion models as it requires volumetric rendering.

ACKNOWLEDGEMENTS

TA thanks Hakan Bilen, Christopher K. I. Williams, Oisín Mac Aodha, Zhengqi Li and Ben Poole for valuable feedback and fruitful discussions throughout the project. The authors also thank Michael Niemeyer and Michael Oechsle for proof-reading the paper. PH was supported in part by the Royal Society (RGS\R2\222045). TA was supported in part by an EPSRC Doctoral Training Partnership.

REFERENCES

- Titus Anciukevicius, Patrick Fox-Roberts, Edward Rosten, and Paul Henderson. Unsupervised causal generative understanding of images. *Advances in Neural Information Processing Systems*, 35: 37037–37054, 2022.
- Titus Anciukevičius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J Mitra, and Paul Guerrero. Renderdiffusion: Image diffusion for 3d reconstruction, inpainting and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12608–12618, June 2023.
- Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.
- 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*, pp. 5855–5864, 2021.
- Miguel Angel Bautista, Pengsheng Guo, Samira Abnar, Walter Talbott, Alexander Toshev, Zhuoyuan Chen, Laurent Dinh, Shuangfei Zhai, Hanlin Goh, Daniel Ulbricht, et al. Gaudi: A neural architect for immersive 3d scene generation. *arXiv preprint arXiv:2207.13751*, 2022.
- Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- Shengqu Cai, Eric Ryan Chan, Songyou Peng, Mohamad Shahbazi, Anton Obukhov, Luc Van Gool, and Gordon Wetzstein. Diffdreamer: Consistent single-view perpetual view generation with conditional diffusion models. *arXiv preprint arXiv:2211.12131*, 2022.
- 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 *CVPR*, 2022.
- 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. Generative novel view synthesis with 3d-aware diffusion models, 2023.
- Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.
- Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14124–14133, 2021.
- Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. *arXiv preprint arXiv:2203.09517*, 2022.
- Hansheng Chen, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su. Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. *arXiv preprint arXiv:2304.06714*, 2023.
- Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alex Schwing, and Liangyan Gui. SDFusion: Multimodal 3d shape completion, reconstruction, and generation. In *CVPR*, 2023.

- Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. Gram: Generative radiance manifolds for 3d-aware image generation. In *IEEE Computer Vision and Pattern Recognition*, 2022.
- Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1174–1183. PMLR, 10–15 Jul 2018. URL <https://proceedings.mlr.press/v80/denton18a.html>.
- Terrance Devries, Miguel Ángel Bautista, Nitish Srivastava, Graham W. Taylor, and Joshua M. Susskind. Unconstrained scene generation with locally conditioned radiance fields. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14284–14293, 2021.
- Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.
- Yilun Du, Cameron Smith, Ayush Tewari, and Vincent Sitzmann. Learning to render novel views from wide-baseline stereo pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- SM Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S Morcos, Marta Garnelo, Avraham Ruderman, Andrei A Rusu, Ivo Danihelka, Karol Gregor, et al. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018.
- Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. Scenescape: Text-driven consistent scene generation, 2023.
- 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*, pp. 12479–12488, 2023a.
- Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *CVPR*, 2023b.
- 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.
- Jiatao Gu, Qingzhe Gao, Shuangfei Zhai, Baoquan Chen, Lingjie Liu, and Josh Susskind. Learning controllable 3d diffusion models from single-view images, 2023.
- Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. 3dgen: Triplane latent diffusion for textured mesh generation. *arXiv preprint arXiv:2303.05371*, 2023.
- Paul Henderson and Christoph H. Lampert. Unsupervised object-centric video generation and decomposition in 3D. In *Advances in Neural Information Processing Systems (NeurIPS) 33*, 2020.
- Paul Henderson, Vagia Tsiminaki, and Christoph Lampert. Leveraging 2D data to learn textured 3D mesh generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- Philipp Henzler, Jeremy Reizenstein, Patrick Labatut, Roman Shapovalov, Tobias Ritschel, Andrea Vedaldi, and David Novotny. Unsupervised learning of 3d object categories from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4700–4709, 2021.
- 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. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30, 2017.
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

- Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23:47–1, 2022.
- Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models, 2023.
- Qingqing Huang, Daniel S. Park, Tao Wang, Timo I. Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, Jesse Engel, Quoc V. Le, William Chan, Zhifeng Chen, and Wei Han. Noise2music: Text-conditioned music generation with diffusion models, 2023.
- Ka-Hei Hui, Ruihui Li, Jingyu Hu, and Chi-Wing Fu. Neural wavelet-domain diffusion for 3d shape generation. In *SIGGRAPH Asia 2022 Conference Papers*, pp. 1–9, 2022.
- Christian Häne, Christopher Zach, Andrea Cohen, Roland Angst, and Marc Pollefeys. Joint 3d scene reconstruction and class segmentation. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 97–104, 2013. doi: 10.1109/CVPR.2013.20.
- Allan Jabri, David J. Fleet, and Ting Chen. Scalable adaptive computation for iterative generation. *arXiv preprint arXiv:2212.11972*, 2022.
- Ajay Jain, Ben Mildenhall, Jonathan T. Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. 2022.
- Animesh Karnewar, Andrea Vedaldi, David Novotny, and Niloy Mitra. Holodiffusion: Training a 3d diffusion model using 2d images, 2023.
- 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*, pp. 4401–4410, 2019.
- Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and Francois Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning, ICML’20, 2020*.
- Seung Wook Kim, Bradley Brown, Kangxue Yin, Karsten Kreis, Katja Schwarz, Daiqing Li, Robin Rombach, Antonio Torralba, and Sanja Fidler. Neuralfield-ldm: Scene generation with hierarchical latent diffusion models, 2023.
- Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL <http://arxiv.org/abs/1412.6980>.
- Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL <http://arxiv.org/abs/1312.6114>.
- Adam R. Kosiorok, Heiko Strathmann, Daniel Zoran, Pol Moreno, Rosalia Schneider, Sona Mokrá, and Danilo Jimenez Rezende. Nerf-vae: A geometry aware 3d scene generative model. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5742–5752. PMLR, 2021. URL <http://proceedings.mlr.press/v139/kosiorok21a.html>.
- Jonáš Kulháněk, Erik Derner, Torsten Sattler, and Robert Babuška. Viewformer: Nerf-free neural rendering from few images using transformers. In *European Conference on Computer Vision (ECCV)*, 2022.

- Hendrik P. A. Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Trans. Graph.*, 22(2):234–257, apr 2003. ISSN 0730-0301. doi: 10.1145/636886.636891. URL <https://doi.org/10.1145/636886.636891>.
- Muheng Li, Yueqi Duan, Jie Zhou, and Jiwen Lu. Diffusion-sdf: Text-to-shape via voxelized diffusion, 2022.
- 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.
- 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- Minghua Liu, Chao Xu, Haiyan Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization, 2023a.
- Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object, 2023b.
- Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7824–7833, 2022.
- Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11461–11471, 2022.
- Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2021. doi: 10.1109/cvpr46437.2021.00286. URL <http://dx.doi.org/10.1109/CVPR46437.2021.00286>.
- Nelson Max. Optical models for direct volume rendering. *IEEE Trans. on Visualization and Computer Graphics*, 1995.
- Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Realfusion: 360° reconstruction of any object from a single image. In *Arxiv*, 2023.
- Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. *arXiv preprint arXiv:2211.07600*, 2022.
- 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 *ECCV*, 2020.
- Norman Müller, , Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Peter Kotschieder, and Matthias Nießner. Diffrr: Rendering-guided 3d radiance field diffusion. *arxiv*, 2022.
- Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022.
- 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*, pp. 7588–7597, 2019.
- Thu Nguyen-Phuoc, Christian Richardt, Long Mai, Yong-Liang Yang, and Niloy Mitra. Blockgan: Learning 3d object-aware scene representations from unlabelled images. *arXiv preprint arXiv:2002.08988*, 2020.

- Michael Niemeyer, Jonathan T. Barron, Ben Mildenhall, Mehdi S. M. Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022. doi: 10.1109/cvpr52688.2022.00540. URL <http://dx.doi.org/10.1109/CVPR52688.2022.00540>.
- Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 165–174, 2019.
- Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pp. 523–540. Springer, 2020.
- Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022.
- Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *International Conference on Computer Vision*, 2021.
- Xuanchi Ren and Xiaolong Wang. Look outside the room: Synthesizing a consistent long-term 3d scene video from a single image, 2022.
- Chris Rockwell, David F Fouhey, and Justin Johnson. Pixelsynth: Generating a 3d-consistent experience from a single image. *arXiv preprint arXiv:2108.05892*, 2021.
- Robin Rombach, Patrick Esser, and Björn Ommer. Geometry-free view synthesis: Transformers and no 3d priors, 2021.
- Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.
- Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.
- Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. GRAF: generative radiance fields for 3d-aware image synthesis. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL <https://proceedings.neurips.cc/paper/2020/hash/e92e1b476bb5262d793fd40931e0ed53-Abstract.html>.
- S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*, volume 1, pp. 519–528, 2006a. doi: 10.1109/CVPR.2006.19.
- Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR’06)*, volume 1, pp. 519–528. IEEE, 2006b.
- Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation, 2023.

- Zifan Shi, Yujun Shen, Jiapeng Zhu, Dit-Yan Yeung, and Qifeng Chen. 3d-aware indoor scene synthesis with depth priors. 2022.
- J Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. *arXiv preprint arXiv:2211.16677*, 2022.
- Ivan Skorokhodov, Aliaksandr Siarohin, Yinghao Xu, Jian Ren, Hsin-Ying Lee, Peter Wonka, and Sergey Tulyakov. 3d generation on imagenet, 2023.
- Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *CoRR*, abs/1503.03585, 2015. URL <http://arxiv.org/abs/1503.03585>.
- Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv:2010.02502*, October 2020. URL <https://arxiv.org/abs/2010.02502>.
- Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Viewset diffusion: (0-)image-conditioned 3d generative models from 2d data. *arXiv*, 2023.
- Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. *arXiv*, 2023.
- 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. In *arXiv*, 2023.
- Hung-Yu Tseng, Qinbo Li, Changil Kim, Suhub Alsisan, Jia-Bin Huang, and Johannes Kopf. Consistent view synthesis with pose-guided diffusion models, 2023.
- Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. Lion: Latent point diffusion models for 3d shape generation. *Advances in Neural Information Processing Systems*, 35:10021–10039, 2022.
- Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International conference on machine learning*, pp. 1747–1756. PMLR, 2016.
- Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation, 2022a.
- Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *CVPR*, 2021.
- 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, 2022b.
- Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolific-dreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023.
- Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022.

- Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7467–7477, 2020.
- Chao-Yuan Wu, Justin Johnson, Jitendra Malik, Christoph Feichtenhofer, and Georgia Gkioxari. Multiview compressive coding for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9065–9075, 2023.
- Jamie Wynn and Daniyar Turmukhambetov. Diffusionerf: Regularizing neural radiance fields with denoising diffusion models. In *arxiv*, 2023.
- Jianfeng Xiang, Jiaolong Yang, Binbin Huang, and Xin Tong. 3d-aware image generation using 2d diffusion models, 2023.
- 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. *Computer Graphics Forum*, 2022. ISSN 1467-8659. doi: 10.1111/cgf.14505.
- Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5438–5448, 2022.
- Paul Yoo, Jiaxian Guo, Yutaka Matsuo, and Shixiang Shane Gu. Dreamsparse: Escaping from plato’s cave with 2d frozen diffusion model given sparse views. *CoRR*, 2023.
- Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4578–4587, 2021.
- Jason J. Yu, Fereshteh Forghani, Konstantinos G. Derpanis, and Marcus A. Brubaker. Long-term photometric consistent novel view synthesis with diffusion models, 2023a.
- Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Tianyou Liang, Guanying Chen, Shuguang Cui, and Xiaoguang Han. Mvimgnet: A large-scale dataset of multi-view images. In *CVPR*, 2023b.
- Xiaoming Zhao, Fangchang Ma, David Güera, Zhile Ren, Alexander G. Schwing, and Alex Colburn. Generative multiplane images: Making a 2d gan 3d-aware. In *Proc. ECCV*, 2022.
- Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5826–5835, 2021.
- Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *CVPR*, 2023.
- Zi-Xin Zou, Weihao Cheng, Yan-Pei Cao, Shi-Sheng Huang, Ying Shan, and Song-Hai Zhang. Sparse3d: Distilling multiview-consistent diffusion for object reconstruction from sparse views. *arXiv preprint arXiv:2308.14078*, 2023.

A ADDITIONAL EXPERIMENTS

A.1 ABLATION EXPERIMENTS

We performed five ablation experiments to quantify the benefit of our key technical contributions and design decisions. We report results on the CO3D hydrant dataset in Tab. 3.

No representation dropout. We first remove the neural representation dropout (No repr. drop.), i.e. when rendering from each view, we also include features from that view even during training. For generation, this results in a large performance degradation wrt FID_H (266.5 vs 118.1), but an improvement wrt FID_D . This is because the ablated model learns trivial solutions, typically a plane placed directly in front of each camera, textured with a projection of the scene. Thus, each diffused viewpoint yields a visually-pleasing image, but these do not depict a single 3D-consistent scene; when we move to held-out viewpoints realism drops dramatically (i.e. higher FID_H). Depth accuracy for 3D reconstruction is also much poorer (e.g. 0.586 vs 0.821), since the predicted depths are not meaningful.

No image-based representation. We next experiment with parameterizing the 3D scene using triplanes, instead of using image-aligned features. In this case the U-Net outputs three planes of features, which are mapped to a single volume of 3D space placed relative to the first camera (Chan et al., 2022; Chen et al., 2022); thus there is no longer a direct geometric correspondence between pixel features and the part of the scene they parameterize. This leads to a substantial drop in model performance across all metrics, indicating that image-based representation is better able to model the details in the 3D scene. We attribute this to its ability to allocate capacity efficiently in the 3D volume, and to allow information flow directly from the noise at pixels to the underlying regions of the 3D scene.

No cross-view attention. We remove our encoder’s ability to attend across jointly across multiple views, by replacing the cross-view attention operation with a traditional attention operation that operates within each image independently. This results in a small drop in performance across all metrics, since only the local (per-3D-point) decoder MLP is available to integrate information across views.

No polar features. We remove the polar unprojection of features, so the scene is defined only within the union of the view frustra of the cameras where diffusion was performed. This results in a performance drop wrt FID_H , since it is now possible that parts of the scene visible in held-out viewpoints are undefined. Interestingly, the performance on 3D reconstruction improves slightly; this may be because more modelling capacity is available for the central foreground region of the scene.

No 3D. We next experiment with generating multi-view images using only 2D multi-view diffusion, still with cross-view attention so the model is expressive enough to learn 3D consistency, but without any explicit 3D scene representation. This model uses the same U-Net architecture, but the last layer of the U-Net directly outputs RGB images, instead of features for rendering. While the resulting images score highly in terms of realism according to the FID metric, they fail to maintain a coherent 3D scene. In other words, each image in the multi-view set appears to represent a different scene, even though each individual image looked convincing. This is quantitatively reflected in the low performance scores for novel view synthesis tasks, demonstrating the model’s inability to accurately create new viewpoints of the scene, as confirmed by low PSNR metrics.

A.2 VARYING THE NUMBER OF VIEWS

We now experiment with training our model using different numbers of views per scene (3–6) in each minibatch, instead of the default eight (Tab. 4). We use the same number of views for diffusion during testing as training, and therefore only report metrics on eight held-out views, to ensure the metrics are comparable across runs. We observe a significant improvement in unconditional generation performance (FID_H) as the number of views increases; performance is not saturating at eight views, so we hypothesise that further improvements would be possible simply by training with more views per scene (at the cost of increased computational expense). Reconstruction performance however appears to saturate around 5–6 views.

#views	Generation	Single-image reconstruction			
	FID _H ↓	PSNR _H ↑	SSIM _H ↑	LPIPS _H ↑	DRC _H ↑
3	224.0	16.84	0.384	0.546	0.732
4	185.2	17.02	0.391	0.516	0.794
5	165.5	17.45	0.418	0.486	0.820
6	138.1	17.47	0.420	0.460	0.830
8	118.1	17.12	0.403	0.449	0.829

Table 4: Results on generation and single-view reconstruction with varying numbers of views during training, on CO3D hydrant. To allow a like-for-like comparison, we report FID and reconstruction metrics on eight held-out views, regardless of the (varying) number of views in which diffusion is performed. We see that generation uniformly improves with more training views, while reconstruction performance saturates.

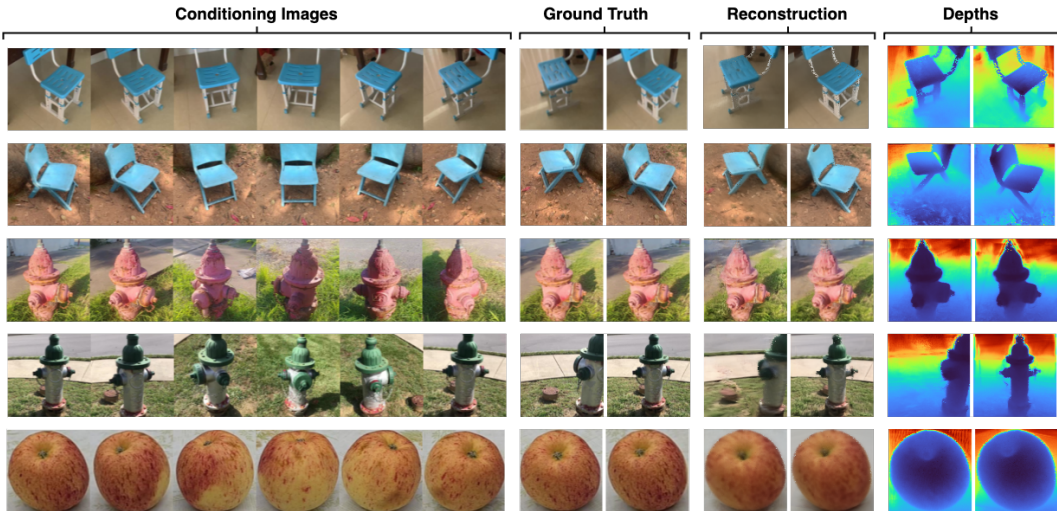


Figure 4: Results from our model on 3D reconstruction from six images. The leftmost columns are the input (conditioning) views; the next two columns show ground-truth images at novel viewpoints. The remaining columns show our model’s 3D reconstruction rendered from those viewpoints, as well as the predicted depth-maps. Note how the model faithfully reconstructs the geometric and textural details visible in its input images.

A.3 FURTHER QUALITATIVE RESULTS

In Fig. 4 we show qualitative results on reconstruction from six input views. We see that the model accurately incorporates all details of the input images. This is further shown by quantitative evaluation (Tab. 1, right four columns), showing that our model achieves high PSNR score for all datasets. In the supplementary material, we additionally include random (not cherry-picked) videos showing results from our model, to allow visualising the coherent and detailed 3D scenes it produces.

A.4 QUALITATIVE COMPARISON WITH BASELINES

In Fig. 5 and Fig. 6, we present a qualitative comparison between our model and the preceding works PixelNeRF++ (Yu et al., 2021), Viewset Diffusion (Szymanowicz et al., 2023) and RenderDiffusion++ (Anciukevičius et al., 2023) on 3D reconstruction. Fig. 5 shows reconstruction results when the model is provided with just a single image at test time; conversely, Fig. 6 shows results when conditioned on six images. We also show unconditional generation results (without any input image) in Fig. 7. It can be seen that the prior generative approaches, RenderDiffusion++ and Viewset Diffusion, struggle to capture intricate details within scenes. On the other hand, the discriminative model PixelNeRF++ tends to render scenes that look detailed from viewpoints near the input image, but become blurry at more distant viewpoints. In contrast, our model demonstrates a superior ability to sample high-fidelity 3D scenes with greater detail and realism, especially at far-away poses, as further corroborated by the quantitative assessments in Tab. 1. We also include 2 samples (using different random seeds) of novel views produced by 2D multi-view diffusion (i.e. “no 3D” ablation of our model described in). We

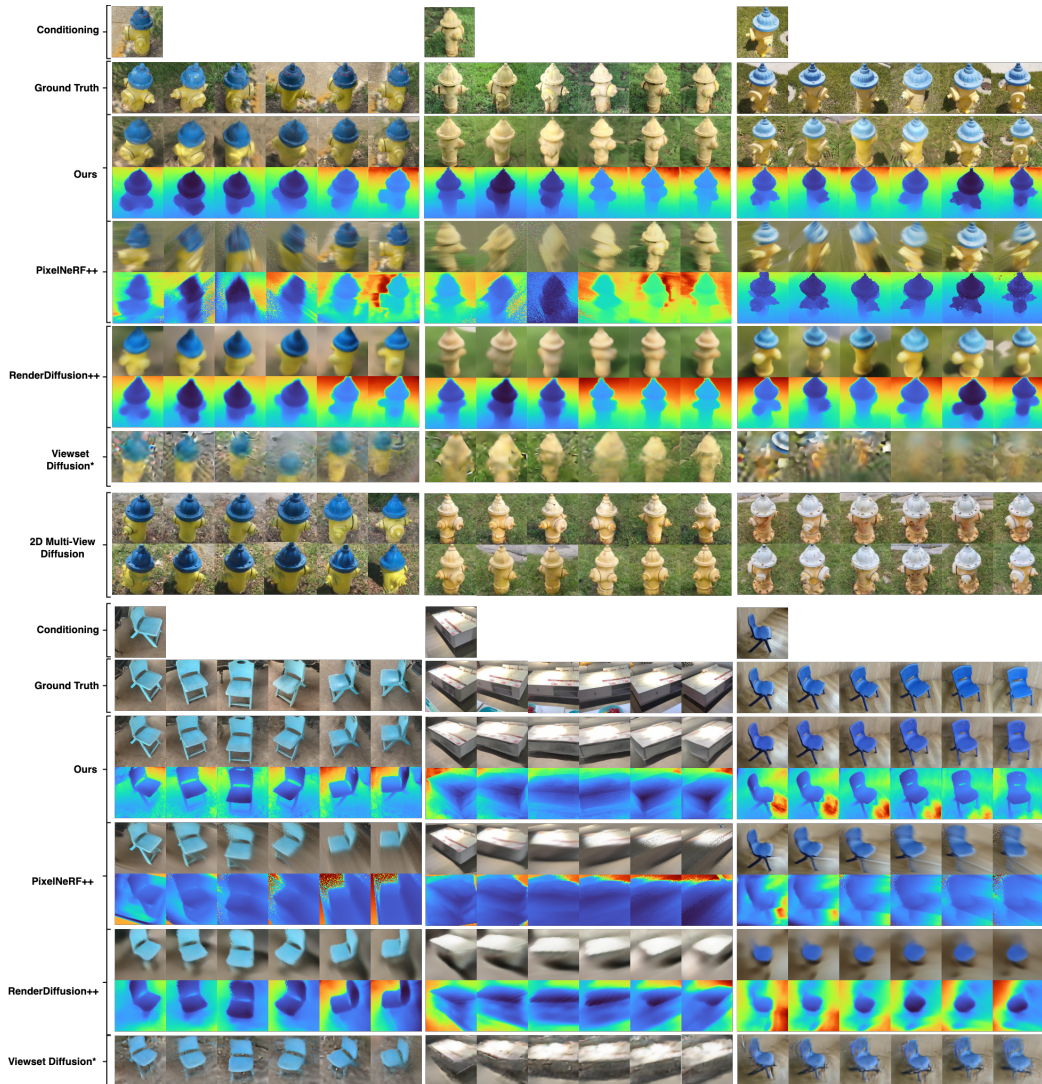


Figure 5: **Single-view 3D reconstruction**. The first row shows the conditioning (input) image, and the second row show the ground-truth for novel views. The subsequent two rows show 3D scenes sampled by our model, showing predicted views and depth maps. Corresponding results from baseline models and the 2D multi-view diffusion ablation study are also shown. Note that the 2D multi-view diffusion ablation does not generate depth maps; in this case, two rows show multi-view image samples generated using different random seeds. Our model demonstrates high-fidelity reconstruction of 3D scenes with plausible reconstructions of unseen regions. In comparison, RenderDiffusion++ samples 3D of low fidelity, while PixelNeRF++ fails render plausible details in unobserved areas. Viewset Diffusion performs well on MVImageNet, but for the larger outdoor scenes in CO3D it often renders floaters or foggy surfaces. We also see that 2D multi-view diffusion (ablation of our model) produces images that are realistic in isolation; however, they are 3D inconsistent and often do not match the ground-truth pose of the object.

see that this ablation of our model produces images that are realistic in isolation; however they are not 3D consistent and often do not match the ground-truth pose of the object.

B ARCHITECTURE

In the following subsections we describe the details of several components, including the multi-view U-Net, feature fusion, and camera pose conditioning.

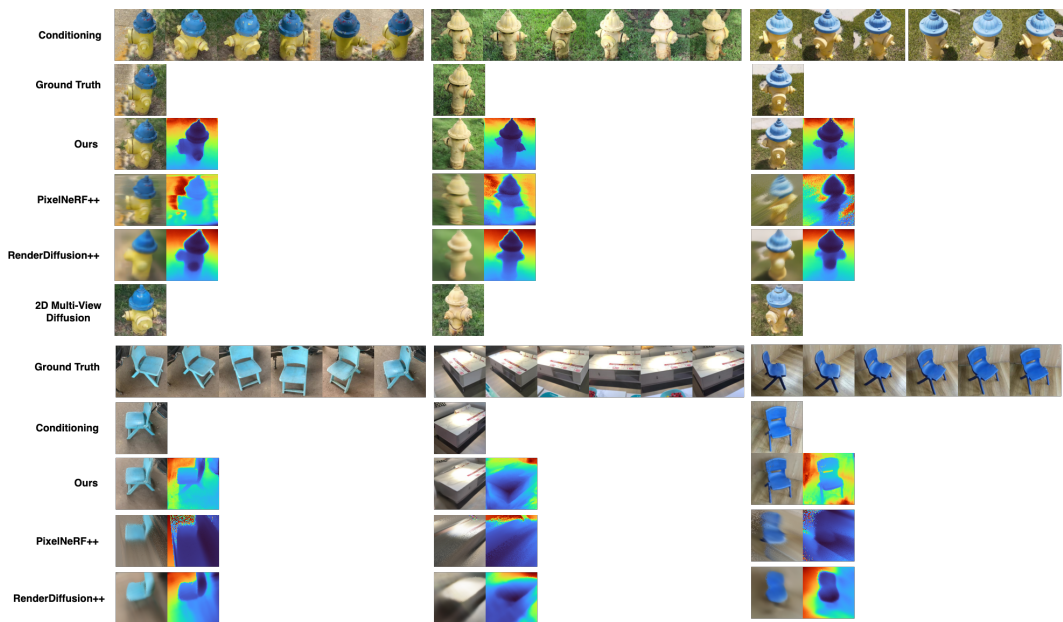


Figure 6: **Sparse-view 3D reconstruction.** The first row displays the six conditioning (input) images, and the second row presents the ground-truth for a novel view. The subsequent row depict sampled 3D scenes by our model, showing the predicted view (left) and its depth map (right). Corresponding results from baseline models and the 2D multi-view diffusion ablation study are also shown. Note that the 2D multi-view diffusion ablation does not generate depth maps; in this case, two rows show multi-view image samples generated using different random seeds. Only our model demonstrates high-fidelity reconstruction of 3D scenes.

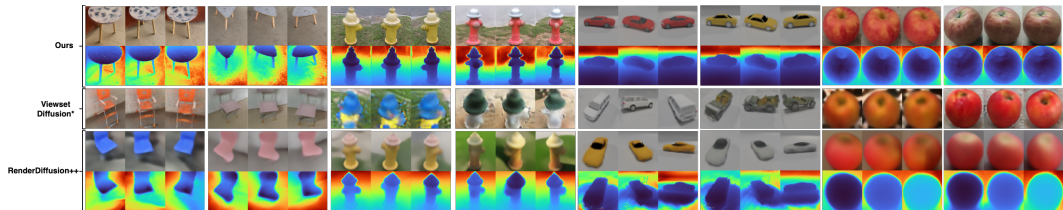


Figure 7: **Unconditional Generation Results.** Each group of 3 columns shows 3 views of a scene sampled unconditionally from the relevant model (we show 8 scenes per model). We see that our model samples the highest-fidelity scenes, especially when trained on challenging in-the-wild MVIImgNet and CO3D datasets. In contrast, prior approaches (RenderDiffusion++ and Viewset Diffusion*) perform well on the simple object-centric ShapeNet dataset, but fail to synthesise fine details on MVIImgNet and CO3D.

Multi-view denoising U-Net. We extend the denoiser’s architecture to support jointly encoding varying numbers of images, to increase the quality of 3D reconstruction. Specifically, we adopt the architecture of [Ho et al. \(2020\)](#), which is a U-Net ([Ronneberger et al., 2015](#)) architecture with 8 ResNet blocks, each block additionally conditioned with a Fourier embedding of diffusion timestep. To support explicit 3D scene representation, we additionally condition each block on the camera pose embedding, which we get by passing the camera extrinsics and intrinsics (relative to some arbitrarily-chosen view) through a small MLP. Similarly, to support class-conditioned generation for MVImgNet and multi-class CO3D, we add a class embedding at each ResNet block. To allow the model to integrate information from arbitrarily many images, we use cross-view attention layers after each ResNet block; these are identical to typical multi-headed attention ([Vaswani et al., 2017](#)), but operate across all points in all views, by flattening these together before the attention operation.

Multi-view feature fusion. As described in [Sec. 3.2](#), our multi-view encoder ingests a set of V images and generates a corresponding set of V feature representations. During the test phase, these features are queried at a 3D point $p \in \mathbb{R}^3$ along a some camera ray. This query is conducted via a combination of camera and equirectangular projections, mapping the 3D point p into feature planes at specific camera positions. Bilinear interpolation is subsequently employed to obtain the final feature set. These features, along with the distances from the 3D points to the camera, are amalgamated using a feature fuser.

After extensive experimentation to identify the optimal architecture for this fuser, we found that multi-head attention (attending across views, independently for each 3D point p) out-performed other architectures, likely due to its ability to selectively focus on relevant features based on a point’s distance to the camera. Despite this, our experiments also revealed that max-fusion of these features offers both faster convergence in terms of wall-clock time and robust performance with far smaller memory requirement; therefore, we report results using max-fusion in our studies. The fused feature is then passed through a 2-layer MLP to output color and density at a 3D point p .

Camera pose conditioning. To facilitate model generalization across an arbitrary number of views, we developed a conditioning approach that pools camera poses. Specifically, we employ a multi-layer perceptron (MLP) to obtain embeddings of each camera pose, each consisting of 16 extrinsic and 9 intrinsic parameters. Then to get the embedding for the i -th view, we concatenated with the max-pooling of all other viewpoint embeddings. This encapsulates not just the unique characteristics of the camera pose for the current view but also their relational context with other views.

Through various experiments, we explored multiple methods for incorporating camera pose information into our model. These approaches ranged from appending embedding as additional channels at the inception of the UNet, to including ray origin and direction as additional channels. However, our experiments indicated that introducing these camera pose embeddings at each ResNet block along with diffusion timestep conditioning yielded the best performance.

C TRAINING

Optimization. We employed the Adam ([Kingma & Ba, 2015](#)) optimizer with a learning rate of 8×10^{-5} and beta values of $\beta_1 = 0.9$ and $\beta_2 = 0.999$ for model training. Norm-based gradient clipping was applied with a value of 1.0. We used a batch size of 8. For evaluation, we used an Exponential Moving Average (EMA) model with a decay factor of `ema_decay = 0.995`.

Minibatch sampling. To support generalization to varying numbers of images, during each training step, we randomly sample 6, 7 or 8 images from a given scene. To reduce GPU memory consumption during training, we render 12% or 5% of pixels depending on resolution.

Volumetric rendering. In our volumetric rendering process, each pixel was rendered by sampling 64 depths along the ray with stratified sampling, followed by 64 importance samples. We sample the background radiance from a uniform distribution when rendering each pixel.

Denosing diffusion. We adopt a sigmoid noise schedule ([Jabri et al., 2022](#)) with 1000 timesteps for our denosing diffusion process. To generate samples, we use 250 DDIM steps ([Song et al., 2020](#)) for unconditional generation and 50 DDIM steps for conditional generation (single-image and sparse-view reconstruction).

D BASELINES

We compare to the most related diffusion method RenderDiffusion (Anciukevičius et al., 2023) and the most related non-generative method PixelNeRF (Yu et al., 2021); we also compare hydrant and apple generation performance with the concurrent Viewset Diffusion (Szymanowicz et al., 2023). Like ours, RenderDiffusion performs diffusion in image space, but uses a triplane representation of latent 3D shapes and requires objects and cameras to be placed in a canonical frame of reference. PixelNeRF is not generative, but performs reconstruction from few views by unprojecting CNN features into 3D space. Viewset Diffusion is a diffusion model over masked multi-view images; however unlike ours, it represents scenes by features in a fixed-size voxel grid and so is unable to adapt its capacity nor model very large scenes.

For a fair comparison with RenderDiffusion and PixelNeRF, we adapt them to use the same feature-extraction architecture as our own work (i.e. an attentive U-Net, modified to include cross-view attention), and thus denote them as RenderDiffusion++ and PixelNeRF++. Without these modifications, both baselines fail catastrophically due to the non-canonical alignment and scales of objects and camera poses.

For Viewset Diffusion we modify their data-loader to remove masking (so the entire scene is visible, not just a single foreground object). For CO3D we retain their scene normalization so the focal object is centered in world space, upright, and of unit size – making their task considerably easier than ours. For MVImgNet, we perform similar normalization, but since the foreground object scale is unknown, we instead rescale based on the bounding box of the camera centers. For ShapeNet, we use the original world-space object and camera poses, giving Viewset Diffusion a strong advantage since all scenes have identical scale and orientation (in contrast our method only has access to the relative camera poses, and no global or object-centric frame of reference). We sweep over different values for the feature volume size, and keep other hyperparameters as in the public code. When evaluating Viewset Diffusion, we use four input views to the U-Net (as in their public code), rather than our six, and calculate heldout-view metrics on four views spaced equally between these.

E DATASETS

We partition each dataset into training, validation, and test sets, following a 90-5-5% split based on lexicographic ordering of provided scene names. The validation set was used for model development and hyperparameter tuning, while the test set was reserved solely for final model evaluation to mitigate any overfitting risks.

Real-world datasets. For the MVImgNet and CO3D datasets, the images are resized to 96×96 or 256×256 resolution. For CO3D, prior to resizing, we take a square crop centered on the ground-truth object mask, to ensure that the images are more focused on the object of interest. For MVImgNet, we take a centre crop with size equal to the smaller dimension of the image. Camera poses from COLMAP for both datasets are scaled for consistency, utilizing either the bounding box around the camera trajectory (MVImgNet) or the object point-cloud (CO3D) as a reference. Unlike Viewset Diffusion (Szymanowicz et al., 2023), we only normalize the scene scale; we do *not* modify the translation and rotation to center the object in world space, since this is not necessary with our view-space scene representation. However, when retraining Viewset Diffusion on our data, we do retain their canonicalisation of translation and rotation.

ShapeNet. To accelerate evaluation, we constrain the number of test scenes in the ShapeNet dataset to 150. Images in this dataset are at their original resolution of 64×64 pixels.

F EXPERIMENT HISTORY AND DESIGN DECISIONS

While developing the presented multi-view diffusion architecture, we have performed various failed experiments. We list these here, describing reasons for failure and our improvement, in case they are of value to other researchers.

3D-aware single image diffusion. Initially, we extended RenderDiffusion (Anciukevičius et al., 2023) to accommodate camera-pose-free training using an image-based rendering. We employed depth supervision to facilitate 3D reconstruction from single-image training, similar to Xiang et al. (2023). However, such design struggled to generate or reconstruct 3D assets of sufficient quality. We believe the limitations stemmed from the model not being trained to create 3D assets that are visually

compelling from multiple viewpoints. Furthermore, the 3D scene is bounded by such approach. To overcome these hurdles, we instead used multi-view image supervision and an unbounded neural scene representation, which can effectively leverage available video datasets. This enabled us to generate 3D assets that appear consistent and high-quality from various perspectives.

Recurrent architecture. In another line of experimentation, we explored the use of a recurrent neural network (RNN) architecture. Here, the model conditions on the prior rendering \mathbf{u}_{i-1} at camera \mathbf{c}_i of the scene \mathbf{z}_{i-1} in a recurrent state i and the noisy image $\mathbf{x}_i^{(t)}$. It is then tasked to reconstruct the image \mathbf{x}_i . Such a “refine, fuse and render” approach has been standard in the literature (Wiles et al., 2020; Rockwell et al., 2021). However, like our earlier single-image model, the RNN-based architecture also failed to produce high-quality 3D assets. We hypothesize that this failure is due to each recurrent state i only receiving a partial observation of the scene \mathbf{z}_{i-1} through its conditioning \mathbf{c}_i (as it only observes camera view frustum). Hence, this again limits the model’s ability to make accurate predictions for denoising the scene. Moreover, even if complete information were available, updating the scene at each 3D point would be a complex task for the model to learn. To address these issues, we transitioned to a multi-view architecture and introduced drop-out technique to enable its training. This change significantly enhanced performance, enabling the model to accurately generate and reconstruct the 3D scene with an arbitrary number of views.

Score-distillation for scene generation. We experimented with the score-distillation approach (Poole et al., 2022), inspired by its effectiveness in generating object-centric scenes (Wang et al., 2023). However, we found several limitations when applying it to our setting. Firstly, the method led to saturated, toy-looking scenes owing to its mode-seeking, optimization-based sampling. While this issue might not critically impact 3D generation, it becomes problematic for 3D reconstruction, which requires an accurate depiction of the conditioning image. Out-of-the-box score-distillation often fails when the conditioning image does not align with the dominant mode. Secondly, the technique necessitates careful selection of camera poses. This requirement is manageable for object-centric scenes but becomes increasingly challenging for larger, more complex scenes. Due to these limitations, we opted for classical denoising diffusion sampling of 3D scenes, similar to Anciukevičius et al. (2023). This approach is not only faster but also less memory-intensive, taking mere seconds as opposed to hours required by score-distillation.