

Viewset Diffusion: (0-)Image-Conditioned 3D Generative Models from 2D Data

Stanislaw Szymanowicz Christian Rupprecht Andrea Vedaldi

Visual Geometry Group — University of Oxford

{stan,chrisr,vedaldi}@robots.ox.ac.uk

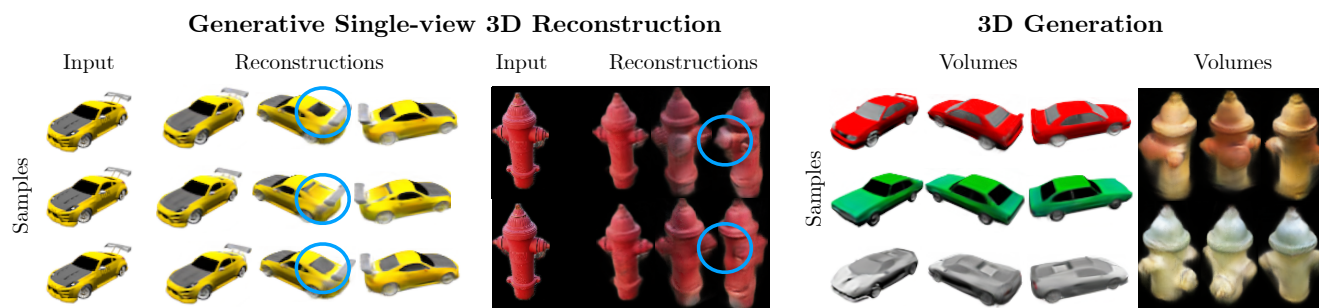


Figure 1: Viewset Diffusion. Our category-specific models perform both ‘generative’ 3D reconstruction and unconditional 3D generation. In single-view 3D reconstruction (left) our models generate plausible explanations of occluded regions (car’s back, hydrant’s occluded side). The same models are able to generate varied 3D objects (right) in a feed-forward manner while being trained only on 2D data.

Abstract

We present *Viewset Diffusion*: a framework for training image-conditioned 3D generative models from 2D data. Image-conditioned 3D generative models allow us to address the inherent ambiguity in single-view 3D reconstruction. Given one image of an object, there is often more than one possible 3D volume that matches the input image, because a single image never captures all sides of an object. Deterministic models are inherently limited to producing one possible reconstruction and therefore make mistakes in ambiguous settings. Modelling distributions of 3D shapes is challenging because 3D ground truth data is often not available. We propose to solve the issue of data availability by training a diffusion model which jointly denoises a multi-view image set. We constrain the output of *Viewset Diffusion* models to a single 3D volume per image set, guaranteeing consistent geometry. Training is done through reconstruction losses on renderings, allowing training with only three images per object. Our design of architecture and training scheme allows our model to perform 3D generation and generative, ambiguity-aware single-view reconstruction in a feed-forward manner. Project page: szymanowicz.github.io/viewset-diffusion.

1. Introduction

3D reconstruction—recovering the 3D shape of the world around us from 2D observations—is a fundamental problem in Computer Vision. In this work, we study single-object few-view reconstruction: given as few as one image of an object we aim to recover a representation of its 3D shape and colour. We also extend our considerations to 3D generation: producing plausible 3D shapes given 0 images as input (see Fig. 1).

Single-view 3D reconstruction is an inherently ambiguous task: during the projection of the 3D scene onto a 2D image plane, the depth dimension is lost. The goal of this task is thus not to recover the exact shape and texture of the invisible regions but to generate a plausible reconstruction. While this is extremely challenging for arbitrary scenes, on the level of objects, this can be achieved by learning a prior over the shape and texture variations across instances.

Learning global or local object priors has been a topic of study of several works [9, 43] which deal with the task of 3D reconstruction in a deterministic manner: they output one reconstruction per object or scene. However, a consequence of this limitation is that in presence of ambiguity the networks can either predict (1) a single most likely guess in the shape space of the model, yielding a plausible but incorrect guess or (2) the expected value in the volume space,

yielding an implausible guess encapsulating all possible reconstructions (see Fig. 2).

In this work, we tackle the problem of modelling ambiguity in few-view 3D reconstruction: the goal is to learn a conditional generation method which enables us to sample a set of plausible reconstructions consistent with a given image of an object from a given viewpoint. Recently, Denoising Diffusion Probabilistic Models (DDPM) [12] have been shown as capable learners of conditional distributions over images [6]. However, as these models are trained by predicting a reverse of the diffusion process, they need to be trained on data in the domain they are operating in. Specifically, a direct application of diffusion models for predicting 3D reconstructions would require training with 3D data. The amount of 3D data is currently much more limited than the availability of image collections. We thus seek to learn a 3D reconstruction model on multi-view 2D data alone.

Concretely, we aim at learning a model that can predict a full 3D radiance field from a single or even no (*i.e.*, unconditional generation) input image for a given object category. To enable this setting we build on a key insight: a sufficiently large number of 2D views of an object contains all information that is encoded in its radiance field. The process of training a neural field from a collection of images (Viewset) and generating images through volume rendering can be seen as a bi-directional mapping between the space of neural fields and Viewsets. It is thus valid to model diffusion in Viewset space, where a set of views is denoised together. Another key insight is that not all views need to be at the same noise level during the diffusion process. We can thus treat the input image as an already denoised image in a Viewset, thus conditioning the set on the input view.

Geometry-free diffusion processes over different views of the same object [20, 42] need to learn the geometry constraints between views, often leading to 3D-inconsistent results. We thus go one step further and explicitly integrate the mapping between Viewsets and neural fields into the network. Our architecture receives a partially noisy Viewset as input and directly predicts a 3D volume that can then be used with a differentiable renderer to generate views of the object (see Fig. 3). This setup can be trained only using Viewsets and requires no 3D supervision. In our experiments, we show that even a dataset with three views per instance is enough to train the model. Our formulation of the input as a set allows for any number of conditioning images, *i.e.* also zero conditioning images, seamlessly extending to unconditional 3D generation.

In summary, our contributions are: (i) An ambiguity-aware 3D reconstruction model which is able to sample different plausible reconstructions given a single input image. Our model is based on recent advances in diffusion models and only requires three images per object. (ii) A training method that unifies different variants of 3D generation: as

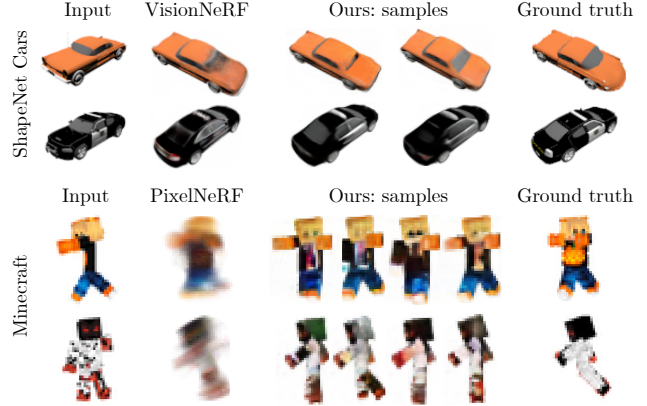


Figure 2: Ambiguities. Under occlusion, deterministic methods blur possible shapes (orange car’s back, Minecraft characters’ poses) and colours (black car’s back, occluded sides of Minecraft characters). Our method samples plausible 3D reconstructions.

a by-product of our training scheme, our model is able to perform 3D unconditional generation and single-view reconstruction while being trained only on 2D data. (iii) An architecture which enables our reconstructions to match the conditioning images, aggregate information from an arbitrary number of views in an occlusion-aware manner and exhibit plausible 3D geometry. (iv) A synthetic benchmark dataset, designed for evaluating the performance of single-image reconstruction techniques in ambiguous settings.

2. Related Work

The task of single-view (or few-view) object-centric 3D reconstruction requires predicting the 3D shape of an object that matches the input images and also a plausible 3D shape for unseen regions. Our method falls into the unsupervised setting of this task, where only 2D data is available, but not 3D ground truth. In this case, the problem is under-constrained and is typically tackled by learning a prior over the space of possible shapes in a category.

Reconstructing Neural Fields. A common representation used in recent single-view reconstruction approaches are Neural Radiance Fields [23]: Sin-NeRF [46] and Diet-NeRF [13] use semantic pseudo-labels in unseen views to provide multi-view pseudo-constraints from pre-trained 2D networks, effectively leveraging a 2D prior. Other works learn an intrinsically 3D prior and represent individual shapes by conditioning on a global latent code [14, 25, 28, 32] which can be optimised at test-time via auto-decoding, akin to latent space inversion in priors learned with 3D GANs [2]. The latent codes can also be local [11, 19, 48] or a combination of global and local features [19], therefore improving high-frequency details near the conditioning viewpoint. Our method builds on the success of these

works: we also learn a 3D prior and employ local conditioning. However, one key feature that makes our approach stand out is that for a single image, we can **sample different plausible reconstructions**, while all deterministic works output a **single reconstruction** (either directly regressed or auto-decoded). In presence of ambiguity, *i.e.* in unseen regions, deterministic approaches learn the conditional average of all possible outcomes which minimises the expected error but usually falls outside of the actual data distribution. As a result, predictions tend to be blurry and often contain multiple modes in one. In contrast, our method samples multiple sharp reconstructions, each of which is different yet plausible (Fig. 2).

Reconstruction beyond Neural Fields. Many other possible 3D representations have been explored, including geometry-free representations [27, 36, 35, 40], occupancy voxel grids [4, 38, 47], texturised meshes [15, 44] or hybrid implicit-explicit representations [29, 45]. While our work is currently based on a discretised Neural Radiance Field, it is compatible with any differentiable formulation.

Ambiguity in 3D Reconstruction. Single-view 3D reconstruction is an ill-posed problem because an input only partially constrains the output. There are two levels of ambiguity in monocular 3D perception [43]: given a 2D observation there (1) exist many possible 3D shapes that explain the observation well, but only a small subset corresponds to plausible shapes within a given category and even then, (2) there are many plausible shapes of a given category explaining the observation, for example, due to occlusion. As such, 3D reconstruction can be tackled as a constrained optimisation problem [9] or with an adversarial loss during training time [43]. Ignoring ambiguity leads to blurry shapes without fine details [5] unless post-processing is used.

Our work also embraces the ambiguity in 3D reconstruction: we train our network to (1) output a shape that matches an observation and (2) only output plausible shapes from a given category. This setting is similar to Wu *et al.* [43], however, our generative approach allows sampling different plausible reconstructions via a conditional diffusion model, rather than a constrained optimisation approach.

3D Modelling with Diffusion Models Recently, Denoising Diffusion Probabilistic Models [12] (DDPM) have been applied to modelling 3D shape distributions by diffusing directly in the space of 3D representations, including point clouds [21], triplanes [30] and discrete radiance fields [24]. Shortcomings of these approaches include (1) assuming an available 3D dataset and (2) requiring heuristics for dealing with ‘floater’ artefacts [24, 30] common in volumes reconstructed from multi-view data in an optimisation setting (see Sec. 3.2 for more detailed discussion).

To alleviate this issue, other methods use pre-trained 2D diffusion models to employ the Score Distillation Loss for

text-to-3D generation [26, 39] in an optimisation pipeline. Extension of this technique to image-to-3D (few-view reconstruction) is often done by textual inversion for conditioning the diffusion model [18, 22]. Concurrently to our work, several others [3, 20, 42] learn image-conditioned diffusion models. The outputs are 2D, and their 3D consistency is only approximate, with frequent flickers [3, 20, 42]. 3D consistency can be enforced via costly (sometimes 1 hour [50]) test-time optimization of a 3D representation [7, 50]. In contrast to prior and concurrent works, our method (1) can be trained on **2D data**, requiring only 3 views of an object (2) is guaranteed to be 3D consistent and (3) is feed-forward, and therefore much faster than test-time distillation methods [7, 22, 50]. Concurrent HoloDiffusion [16] also learns 3D generative models from 2D data, but it only considers unconditional generation, while we propose a principled, unified framework for conditional and unconditional generation.

The work most close to ours is RenderDiffusion [1] where a single noisy image is taken as an input to a network which outputs a 3D representation and is thus able to learn an unconditional generation model from 2D data alone. The difference in our work is introducing the notion of a Viewset: we jointly diffuse over a **set** of images, allowing for probabilistic sampling of 3D reconstructions (RenderDiffusion reconstructs deterministically). In addition, we propose architectural advancements in form of local conditioning and multi-view attention-based aggregation, leading to superior performance in unconditional generation. See 3.3 for more discussion.

3. Method

We consider the problem of learning a distribution over 3D volumes, supporting both unconditional sampling and sampling conditioned on one or more views of the object, for the purposes of single or few-view 3D reconstruction. Given their exploding success in image and video generation, we seek to extend diffusion approaches to this task. The key challenge is the lack of 3D data for supervising such a model, which requires rethinking the training setup.

3.1. Diffusion: notation and preliminaries

Consider first the problem of learning a distribution $p(\mathbf{x})$ over 2D images $\mathbf{x} \in \mathbb{R}^{3 \times H \times W}$ (or, with little changes, a distribution $p(\mathbf{x}|y)$ conditioned on additional information y such as a text description). The diffusion approach generates a sequence of increasingly noisier versions of the data. This sequence starts from $\mathbf{x}_0 = \mathbf{x}$ and adds progressively more Gaussian noise $\mathbf{x}_t = \sqrt{1 - \beta_t^2} \mathbf{x}_{t-1} + \beta_t^2 \bar{\epsilon}_t$, where $\beta_t \in (0, 1)$ is the noise standard deviation and $\bar{\epsilon}_t \sim \mathcal{N}(0, I)$ are i.i.d. normal samples.

The marginal distribution $p(\mathbf{x}_t)$ at step t can be characterised by rewriting $\mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \epsilon_t$, where $\sigma_1 = \beta_1$,

$\sigma_t^2 = (1 - \beta_t^2)\sigma_{t-1}^2 + \beta_t^2$, $\alpha_t^2 = 1 - \sigma_t^2$ and ε_t are also normally distributed (but not independent).

In order to draw a sample x_0 , one starts backward, drawing first a sample x_T from the marginal $p(x_T)$, and then taking samples x_t from $p(x_{t-1}|x_t)$, until x_0 is obtained. The key observation is that these are comparatively simple distributions to learn. Various, slightly different, formulations are possible; here, we learn a denoising network $\hat{x}_0(x_t)$ that tries to estimate the “clean” sample x_0 from its noisy version x_t .

This is trained by minimizing the loss

$$\mathcal{L}(\hat{x}_0) = \frac{1}{|\mathcal{X}|} \sum_{x_0 \in \mathcal{X}} w_t \mathbb{E}_{p(x_t|x_0)}$$

3.2. The challenge of a 3D extension

Now consider the problem of learning a distribution $p(v)$ where v is a model of a 3D object or scene. In this paper, we assume that v is a *radiance field*, although most of our considerations apply to any kind of 3D model. Hence, $v = (\sigma, c)$ is a pair of functions σ and c mapping a 3D point $p \in \mathbb{R}^3$ to an opacity value $\sigma(p) \geq 0$ and an RGB color $c(p) \in [0, 1]^3$. For simplicity, we discretize the radiance field over a 3D grid, expressing it as a tensor $v \in \mathbb{R}^{4 \times H \times W \times D}$, and evaluate $(\sigma(p), c(p))$ using bilinear interpolation and padding as needed. Given a camera π (specified by rotation, translation and intrinsic parameters such as the focal length), one can then *render* an image $x = \Psi(v, \pi)$ in a differentiable manner by projecting the volume via ray casting.

A direct application of diffusion to radiance fields v (or of any other 3D models) assumes the availability of a dataset \mathcal{V} of such fields for training. Differently from images, however, 3D models are seldom directly available. One possible approach is to construct a dataset of 3D models from multi-view data. Specifically, a *Viewset* (x, Π) is a set $x \in \mathbb{R}^{N \times 3 \times H \times W}$ of N views of given 3D scene with known camera poses $\Pi = (\pi_i)_{i=1}^N$. Given a sufficiently large Viewset x , one can use NeRF, or a similar method, to reconstruct a corresponding 3D model v ; given enough Viewsets, one can then build a collection \mathcal{V} of corresponding 3D models to train diffusion methods.

There are a number of undesirable aspects to this approach. First, it requires reconstructing a large number of 3D models v before even starting to learn their distribution $p(v)$ (recall that NeRF can take minutes if not hours for a single reconstruction). Second, the number of 2D images required for reconstructing a 3D model v is usually large (≥ 50) and not always available. Third, even when there are enough images, reconstructing a 3D model v from its views x is still ambiguous as there are generally many different models v that result in (nearly) exactly the same views. The ambiguity is caused by visibility (the interior of

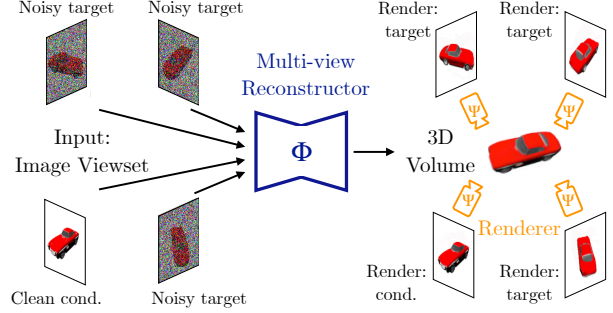


Figure 3: Method. Viewset Diffusion takes in **any** number of clean conditioning images and target images with Gaussian noise (Sec. 3.3). The denoising function is defined as reconstructing (Sec. 3.4) and rendering (Sec. 3.5) a 3D volume. When there is at least one clean conditioning view, Viewset Diffusion samples plausible 3D reconstructions. When all input views are noisy, Viewset Diffusion generates 3D volumes.

an object does not matter to its appearance) and other less obvious reasons (e.g., a compact 3D radiance field representation using triplanes or TensorRF is defined only up to a feature rotation). These ambiguities may cause different 3D models to be reconstructed inconsistently, and inconsistencies must be eliminated before learning the distribution $p(v)$ (see Rodin [41]). Finally, the reconstructed radiance fields are often not immediately ready for training diffusion models and one needs to employ heuristics, such as opacity regularisers, to minimise ‘floater’ artefacts [30].

Because of these reasons, we explore developing a diffusion model that can learn a (unconditional or conditional) distribution over 3D volume v given only *2D images* x for training. Because one can think of the 3D volume v as a *latent variable*, which must be inferred during training, it is tempting to just adopt *latent diffusion*, which has been spectacularly successful also for 2D images. Unfortunately, latent diffusion is inapplicable in this case. The reason is that latent diffusion assumes apriori knowledge of the encoder $x \mapsto v$ mapping input data x to latents v and back. In our case, this mapping, which amounts to image-based 3D reconstruction, is *unknown and ambiguous*, and would need to be learned simultaneously.

In summary, our challenge is to (1) learn a conditional or unconditional generator of 3D volumes (radiance fields); (2) using only 2D images for training; (3) while retaining key benefits of diffusion, including stable training and avoiding mode collapse.

3.3. Viewset diffusion

Our approach is centred around a few simple but powerful observations. First, we note that diffusion is of a simple application in image space, because, differently from the 3D volumes v , its views x are observable. Furthermore,

while the views \mathbf{x} may identify the 3D volume \mathbf{v} only up to an ambiguity class, this ambiguity is eliminated by projecting the model back to its views. Finally, we note that latent variables *do* already exist in any diffusion model: they are the intermediate activations in the network that is used to implement the denoising function — one simply does not apply diffusion to these latents.

Our idea is to *apply diffusion to the views \mathbf{x}* instead of the 3D model \mathbf{v} and recover the 3D model as a latent variable in the denoising network. Specifically, we learn an encoder network

$$\mathbf{v} = \Phi(\mathbf{x}_t, \Pi, \sigma_t)$$

which, given a *noised Viewset* (\mathbf{x}_t, Π) , produces as output a radiance field \mathbf{v} . This radiance field is then *decoded* into an estimate

$$\hat{\mathbf{x}}_0(\mathbf{x}_t) = \Psi(\mathbf{v}, \Pi) = \Psi(\Phi(\mathbf{x}_t, \Pi, \sigma_t), \Pi)$$

of the *clean Viewset* by the decoder Ψ that implements the radiance field rendering equations. This is the same formulation as standard image-based diffusion, except that (1) one generates a set of views in parallel instead of a single image and (2) the denoiser network has a particular geometric interpretation. The training loss is the same as for standard diffusion:

$$\mathcal{L}(\Phi, \mathbf{x}_0, \mathbf{x}_t, \Pi) = w_t \|\Psi(\Phi(\mathbf{x}_t, \Pi, \sigma_t), \Pi) - \mathbf{x}_0\|^2$$

where $\mathbf{x}_t = \sqrt{1 - \sigma_t^2} \mathbf{x}_0 + \sigma_t \epsilon_t$ is a noised version of the (clean) input Viewset \mathbf{x}_0 and w_t is timestep-dependent loss term weighting.

Single and few-view reconstruction. With the model above, we can learn *simultaneously* unconditional 3D generation as well as single and few-view reconstruction with little modifications. Given a conditioning Viewset (\mathbf{y}, Π') , in fact, we can sample $p(\mathbf{x}|\Pi, \mathbf{y}, \Pi')$ by feeding into the network Φ a mixture of noised and clean views:

$$\mathbf{v} = \Phi(\mathbf{x}_t \oplus \mathbf{y}, \Pi \oplus \Pi', \sigma_t \oplus \mathbf{0})$$

where \oplus denotes concatenation along the view dimension. Here $\sigma_t \oplus \mathbf{0}$ means that we treat σ_t as a vector of noise variances, one for each view in the Viewset, and append zeros to denote the fact that the conditioning views \mathbf{y} are “clean”.

Discussion. The approach above defines a distribution over views \mathbf{x} , but *does not* define explicitly a distribution $p(\mathbf{v})$ on the 3D models themselves. Still, the formulation allows for readily sampling/generating such 3D models by sampling the corresponding views. Ultimately, we can see the Viewset \mathbf{x} as a *representation* of the 3D object \mathbf{v} . While the Viewset \mathbf{x} may not contain sufficient information to recover \mathbf{v} uniquely, given a sufficient number of such views,

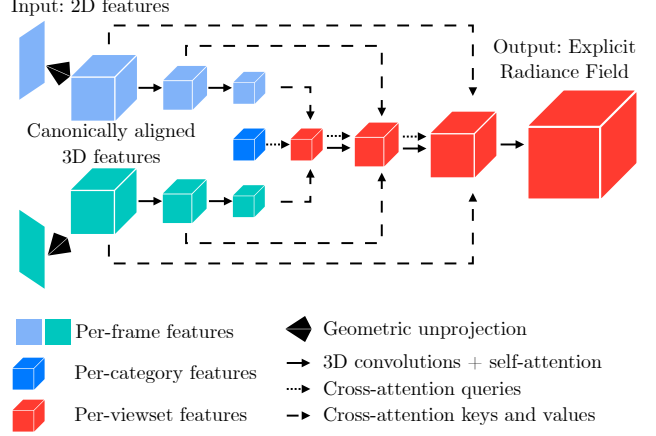


Figure 4: Architecture. 2D inputs are unprojected along camera rays to canonical feature volumes. Multi-scale features are extracted and aggregated with an attention mechanism to output a single radiance field. Number of input frames can be variable.

uncertainty in the reconstruction of the object can be relegated to nuisance ambiguities, discussed above. In practice, defining a distribution over views \mathbf{x} allows training with few (in our case 3) views per object, which would not have been enough to optimise a radiance field. At inference time the number of jointly denoised images can be larger, fully describing the volume \mathbf{v} .

Our approach is related to the recent RenderDiffusion by [1], but with substantial theoretical and practical differences. First, using our notation, their approach amounts to reducing the size of the Viewset to a *single view*. Different from using a non-trivial Viewset, a single view is insufficient to adequately represent a 3D object \mathbf{v} . By using a Viewset, the generation of successive denoised samples ensures coherent and plausible appearance and shape from **all** viewpoints, which is not guaranteed in RenderDiffusion, which only denoises a **single** viewpoint. In fact, our network Φ performs (noisy) multi-view reconstruction to draw samples and supports any number of views for reconstruction.

3.4. Network architecture

The first step in our architecture is feature extraction via a small 5-layer 2D convolutional net. The second step is image unprojection: for each image I , associated camera pose Π and the camera intrinsic matrix K we form a volume $V \in \mathbb{R}^{3 \times H \times W \times D}$ where each voxel holds the RGB value of the pixel it would be projected to with the camera Π . Each volume is aligned with the world coordinate centre and aligned with the global world axes. Unprojected RGB volumes from N frames are then passed through a variant of a 3D U-Net with attention-based cross-frame aggregation. Specifically, the encoder is a series of 3D ResNet

Convolutional and self-attention blocks, forming a set of feature maps $\{W_i\} \in \mathbb{R}^{C_i \times H_i \times W_i \times D_i}$ independently for each frame. Next, the coarsest feature maps are aggregated with an attention mechanism. We aggregate the features at each voxel location independently to minimise computational complexity, which is $\mathcal{O}(N^2)$ for each voxel element. Afterwards, the feature map is upsampled, passed through a sequence of 3D ResNet Convolutional blocks and acts as a query for the next N finer feature maps. See Fig. 4 for illustration and Supplementary Material for more details. The output of the network is one volume for the N frames, which is then upsampled in 3D with a series of 5 3D convolutional layers. Unprojection allows the volume to easily match the conditioning image, while aggregation allows a learnt combination of features across views which can depend on, for example, occlusion. The impact of both of these components is analysed in Table 4.

3.5. Radiance Field, Rendering and Training

We implement radiance fields v as explicit voxel grids holding non-activated opacity and colour values. Rendering is done via raymarching along rays sent from the camera origin through each pixel, obtaining the values of colour and opacity via trilinear sampling and post-interpolation activation, similar to DVGO [37]. Colour values are obtained with the volumetric rendering equations – see DVGO [37] for details. At training time, we use view dropout: we randomly give our network one of three possible input types: (1) two noisy views, (2) one noisy or (3) one clean and one noisy. We optimise the network Φ by minimising the photometric L2 loss, as discussed in 3.3, of the renders from both viewpoints. We use Min-SNR-5 weighting [10] of different timesteps. In addition, we penalise a third, ‘holdout’ view, not available to the network, as a regularisation strategy to encourage 3D consistency. Please see Supplementary Material for more details.

4. Experiments

We perform extensive experiments to evaluate our method and its components.

Data. We use benchmarks from three datasets for validating the effectiveness of our method: two versions of a synthetic dataset of articulated *Minecraft* characters, ShapeNet-SRN Cars [33] dataset and the Hydrant class from CO3D [27] dataset.

We build a synthetic dataset of *Minecraft* characters (see Fig. 5). Each character consists of a torso with randomly articulated arms, legs and head and is textured with one of 3,000 skins. We render 40,000 training and 5,000 validation examples with OpenGL at 256x256 resolution and downsample to 48x48 with nearest neighbour interpolation. Each example consists of 3 images and associated camera



Figure 5: Minecraft dataset. Textured meshes are articulated and rendered from random camera viewpoints, allowing for procedural generation of a large number of instances.

poses. We form two test sets with the *Minecraft* dataset: one with randomly sampled skins (unseen at training time), poses and camera viewpoints and one with characters whose 3D pose is ambiguous when seen from a single viewpoint, e.g. due to one arm being occluded by the torso (also with skins unseen at training time). Each example in each test set (100 in total) consists of one conditioning image and camera pose and one image of the same character from a different camera viewpoint. To facilitate future research we will release the dataset and code.

We also test on ShapeNet-SRN Cars benchmark and Hydrant class from CO3D dataset. In ShapeNet-SRN we use the standard train/val/test split.

For pre-processing details in CO3D, please see the supplementary material. Each training example is formed from 3 random images and associated camera poses Π taken from a single video sequence. Different examples may come from the same scene due to data limitations but they are treated as independent scenes at training and test time.

We use the default train/validation split from the *fewview dev* subset and resize images to 128×128 resolution. Our test set examples consist of one conditioning image available to the network (with the associated camera pose) and one testing image of the same object from a different viewpoint. The test set consists of 100 randomly chosen examples from the test sequences.

Baselines. In this work our primary aim is to show the importance of modelling ambiguity *i.e.* that stochastic methods result in sharper and possibly more accurate reconstructions than deterministic methods. Hence, we compare against methods that can be trained with 2D data: a fully deterministic method, PixelNeRF [48], our reimplementation of RenderDiffusion [1] and our improvement over it, RenderDiffusion++. We train PixelNeRF using the publicly available code with a tuned learning rate and Softplus activation for improved training stability. Details are available in the supplementary material. We reimplement RenderDiffusion using publicly available details. RenderDiffusion originally uses only single images, but for a fair comparison, we also compare to ‘RenderDiffusion++’, a variant of RenderDiffusion where the network still receives one im-

age, but is supervised by 3 images per object and uses our architecture (*i.e.*, utilises the unprojection mechanism and outputs a 3D grid instead of a triplane). For RD++ we use the same hyperparameters as for our method, except change the nature and number of input views. Finally, to evaluate the importance of a probabilistic model, we include a baseline of our method without diffusion (*i.e.*, one that receives a clean image and directly regresses a 3D volume). On ShapeNet-SRN we compare to single-view reconstruction deterministic works which report scores on this standard benchmark [8, 14, 19, 33, 32, 42, 48].

Evaluation protocol. We follow the standard evaluation protocol in ShapeNet-SRN. In Minecraft and CO3D we form a small testing set with 100 examples of image pairs and associated camera poses. In each pair, one image is used as the conditioning image available at the input and the other one is a target view of the object. We render the reconstructed volume from the target viewpoint and measure the Peak Signal-to-Noise Ratio (PSNR) and Learned Perceptual Image Patch Similarity [49] (LPIPS), measured with a pre-trained VGG Net [31], when compared to the ground truth image. The baselines we use are deterministic in single-view reconstruction, hence we render the reconstruction once for each object. Our method is stochastic, which allows us to generate different samples. Deterministic methods learn to predict the conditional average, which in turn maximises the expected value of mean-based metrics such as PSNR under ambiguities. In high-dimensional space (e.g. images), the distance between two samples (prediction and target) will almost always be larger than the distance between the mean and a sample. Thus, we take multiple samples of reconstructions of every object and report the **best** PSNR from these samples (as some reconstructions deviate from ground truth despite matching the conditioning image and being plausible). In Minecraft and CO3D we take 100 samples per testing instance. In ShapeNet-SRN we take 20 samples due to the computational burden of the dataset. As LPIPS is not as strongly affected by this property since it measures perceived visual similarity, here we report the **average** across all 100 / 20 samples. For completeness, we also report the best LPIPS across all samples and the average PSNR, but we report them in brackets ‘(x)’ as we do not deem them to be comparable metrics.

Technical details. We optimise the parameters of our network with Adam [17] optimizer and learning rate 2×10^{-5} . We use batch size 16 and optimise our diffusion networks for 100k (ShapeNet) / 200k (Minecraft, CO3D) iterations and the networks without diffusion for 40k iterations. We use a diffusion schedule with 1000 diffusion steps and a cosine noise schedule. At inference, we use 250 steps of DDIM [34] sampling.

Method	Random		Ambiguous	
	PSNR \uparrow	LPIPS \downarrow	PSNR \uparrow	LPIPS \downarrow
RenderDiffusion	19.85	0.213	16.33	0.236
PixelNeRF	21.55	0.220	17.86	0.250
RenderDiffusion++	24.18	0.157	19.92	0.210
Ours w/o \mathcal{D}	24.63	0.115	20.26	0.156
Ours w \mathcal{D} - best	24.82	(0.072)	21.50	(0.081)
Ours w \mathcal{D} - mean	(22.81)	0.107	(18.62)	0.130

Table 1: Single view reconstruction - Minecraft. Ours achieves larger gains in the Ambiguous subset, showcasing the strength of probabilistic modelling.

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
3DiM	21.01	0.57	-
LFN	22.42	0.89	-
SRN	22.25	0.88	0.129
CodeNeRF	22.73	0.89	0.128
FE-NVS	22.83	(0.91*)	(0.099*)
VisionNeRF	22.88	<u>0.90</u>	0.084
PixelNeRF	23.17	0.89	0.146
Ours w/o \mathcal{D}	<u>23.21</u>	<u>0.90</u>	0.116
Ours w \mathcal{D} - best	23.29	0.91	(0.094)
Ours w \mathcal{D} - mean	(22.72)	(0.90)	<u>0.099</u>

Table 2: Single view reconstruction – ShapeNet Cars. Ours achieves best PSNR, with the additional benefit of probabilistic treatment. *FE-NVS optimises SSIM in training, affecting perceptual sharpness.

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
RenderDiffusion	17.43	0.70	0.263
PixelNeRF	18.07	0.67	0.297
RenderDiffusion++	21.61	0.69	0.282
Ours w/o \mathcal{D}	22.06	0.78	0.217
Ours w \mathcal{D} - best	22.36	0.80	(0.176)
Ours w \mathcal{D} - mean	(20.52)	(0.77)	0.199

Table 3: Single view reconstruction - CO3D. Our method improves over baselines in CO3D hydrant dataset across all metrics.

4.1. Single-view reconstruction: Minecraft dataset

We first illustrate the ability of our method to resolve ambiguity on our synthetic Minecraft dataset. We evaluate two subsets. The ‘Ambiguous’ subset consists of 100 examples with ambiguous poses when reconstructed from a single image due to occlusion or projective ambiguity. The ‘Random’

subset consists of 100 randomly chosen examples to illustrate that our method can both resolve ambiguity as well as reconstruct unambiguous examples correctly.

In Table 1 we compare the reconstruction quality, as measured by PSNR and LPIPS. Sampling multiple plausible reconstructions via view-set diffusion leads to an improved test-view PSNR in the best sample in both the random and ambiguous datasets. The significant gain in PSNR in the ‘Ambiguous’ dataset illustrates that diffusion can effectively resolve ambiguity present in single-view reconstruction. Renders of samples of the reconstructed volumes are shown in Fig. 2 and show how diverse poses and textures are sampled under the presence of ambiguity.

In Table 1 it is also seen that using view-set diffusion leads to a decrease in **average** LPIPS (lower is better), suggesting that **all** samples from our method are more perceptually plausible than the results from the baselines. Finally, the improvement in the metrics is further accompanied by qualitative comparison in Figs. 2 and 6 where the samples from our method are seen to be much sharper than the baseline results.

4.2. Single view reconstruction: ShapeNet & CO3D

We further validate the effectiveness of our method on the standard benchmark ShapeNet-SRN Cars and the challenging real-world data from the CO3D dataset. Quantitative results can be seen in Tables 2 and 3 where our method with diffusion outperforms the baselines. Similarly to the experiment on synthetic data, the best sample from our model also achieves higher PSNR than the baselines, suggesting the model is able to sample from a correct distribution. The samples from our model obtained with Viewset diffusion achieve lower (better) LPIPS than all baselines in CO3D (Table 3) and almost all baselines in ShapeNet (Table 2), suggesting they are perceptually closer to the ground truth. VisionNeRF [19] outperforms our method on LPIPS and achieves its performance due to a ViT-based 2D feature extraction. Our 2D feature extractor is much smaller (5 convolutional layers) and perhaps performance of our method can be boosted with more powerful feature extraction.

4.3. Unconditional generation

In our framework, unconditional generation is a special case where the number of clean input views is 0. To generate 3D volumes prior works [24, 41, 30] require 3D ground truth at training time, while we require only 3 views per object. At test time, our view-set can be arbitrarily big, therefore constraining the 3D geometry not only via a learnt prior but also via multi-view consistency of the generated images. In Fig. 7 we show samples of our method and of RenderDiffusion. The examples generated by our method using Viewset diffusion show the best sharpness from all viewpoints, both in texture and in shape, suggesting that the

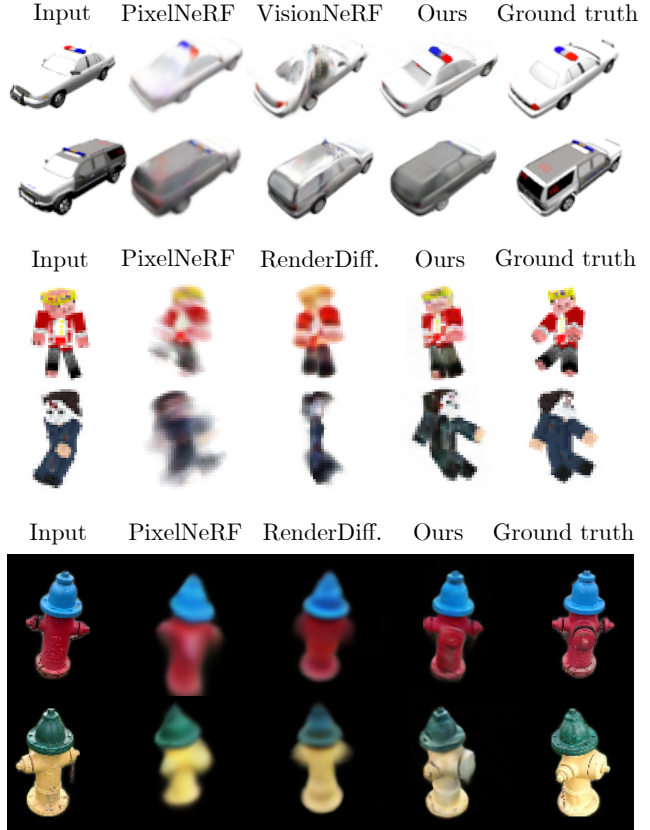


Figure 6: Single view reconstruction – examples. Our method outputs sharper shapes than prior work. The solutions are ambiguous, therefore our samples do not match the ground truth exactly but are more plausible than deterministic baselines.

view-set is important for the generation of shapes that have a well-defined geometry when observed from 360°. RenderDiffusion does not have local conditioning, hindering its capability to produce high-frequency details. In addition, using a Viewset, as opposed to RenderDiffusion’s single view enables good detail from all viewpoints, not just the diffused one. The reason for the superior performance of a Viewset is the availability of all views in the final step of generation, where the network performs nearly-clean reconstruction. In RenderDiffusion’s or ‘without (w/o) Viewset’ paradigm, this final step is, in fact, single-view reconstruction, which in the previous section we showed has inferior performance when done with only one input view. Availability of all viewpoints in a Viewset allows for generating sharp textures in all 360°.

4.4. Ablations

We assess the importance of different components of our method: input image unprojection, attention-based aggregation of features from different views and Viewset diffusion \mathcal{D} . We test on the ‘Ambiguous’ Minecraft dataset and

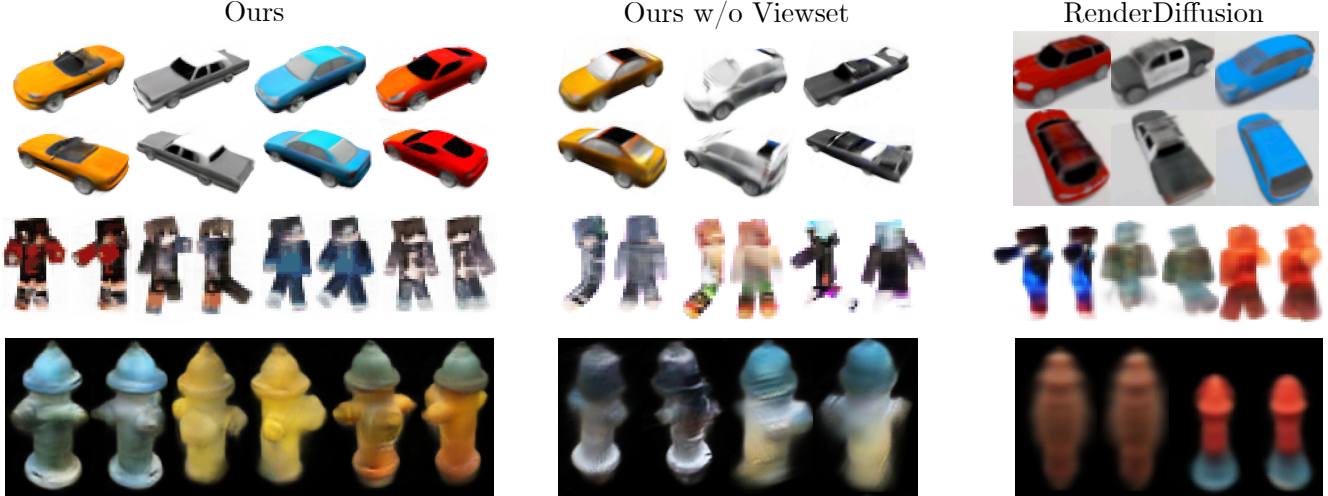


Figure 7: Unconditional generation - Cars, Minecraft and Hydrants. Samples from our method show higher visual detail than RenderDiffusion [1]. Viewset diffusion further improves the samples over single view generation and comes for free with our method.

	PSNR \uparrow	LPIPS \downarrow
Full model	20.36	0.075
\ominus diffusion \mathcal{D}	18.85	0.101
\ominus attention in aggregation	19.54	0.100
\ominus unprojection	18.26	0.164

Table 4: Ablations. Impact of removing component from our method on reconstruction quality.

evaluate best PSNR and average LPIPS in the unseen novel view. We train smaller models (half the number of U-Net convolutional layers and no self-attention layers) for fewer iterations (60k) due to computational cost. Results are reported in Table 4. Not using diffusion leads to a drop in PSNR and worse perceptual quality due to the reconstructions being blurry in the presence of ambiguity. Removing attention-based feature aggregation across frames and aggregating them with a simple mean prohibits the network from reasoning about viewpoints and occlusion when pooling the features from different views. Finally, removing unprojection hinders the learning process due to the removal of local conditioning which is known to improve the learning process [48].

5. Conclusions

In this paper, we have presented a method to learn a model for probabilistic single-view reconstruction using a diffusion model. While the commonly used diffusion framework needs a dataset of samples from the diffusion space (in this case 3D volumetric fields), we show a for-

mulation that enables learning from a dataset without 3D ground truth with as few as three views per object. This idea is fairly general and can drastically expand the application space of diffusion models, especially in the domain of 3D computer vision. We also showed a unified perspective on 3D reconstruction and 3D generation, enabling feed-forward probabilistic 3D reconstruction with diffusion models. In our experiments, we show that probabilistic modelling of the single view reconstruction problem leads to higher quality results and avoids blurry solutions.

Ethics. We use the CO3D dataset in a manner compatible with their terms. The images used in this research do *not* contain personal information (*e.g.*, faces). For further details on ethics, data protection, and copyright please see <https://www.robots.ox.ac.uk/~vedaldi/research/union/ethics.html>.

Acknowledgements. S. Szymanowicz is supported by an EPSRC Doctoral Training Partnerships (DTP) EP/R513295/1. A. Vedaldi and C. Rupprecht are supported by ERC-CoG UNION 101001212. C. Rupprecht is also supported by VisualAI EP/T028572/1.

References

- [1] Titas Anciukevicius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J. Mitra, and Paul Guerrero. RenderDiffusion: Image diffusion for 3D reconstruction, inpainting and generation. *arXiv*, 2022. 3, 5, 6, 9
- [2] 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. 2

- [3] 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 *arXiv*, 2023. 3
- [4] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. 3
- [5] Angela Dai, Charles Ruizhongtai Qi, and Matthias Nießner. Shape completion using 3d-encoder-predictor cnns and shape synthesis. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2017. 3
- [6] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In *Proc. NeurIPS*, 2021. 2
- [7] Jiatao Gu, Alex Trevithick, Kai-En Lin, Josh Susskind, Christian Theobalt, Lingjie Liu, and Ravi Ramamoorthi. Nerfdiff: Single-image view synthesis with nerf-guided distillation from 3d-aware diffusion. In *2302.10109*, 2023. 3
- [8] Pengsheng Guo, Miguel Angel Bautista, Alex Colburn, Liang Yang, Daniel Ulbricht, Joshua M. Susskind, and Qi Shan. Fast and explicit neural view synthesis. In *WACV*, 2022. 7
- [9] JunYoung Gwak, Christopher B Choy, Manmohan Chandraker, Animesh Garg, and Silvio Savarese. Weakly supervised 3d reconstruction with adversarial constraint. In *3D Vision (3DV), 2017 Fifth International Conference on 3D Vision*, 2017. 1, 3
- [10] Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-snr weighting strategy. In *arXiv*, 2023. 6
- [11] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2
- [12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. 2020. 2, 3
- [13] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5885–5894, October 2021. 2
- [14] Wonbong Jang and Lourdes Agapito. CodeNeRF: Disentangled neural radiance fields for object categories. In *Proc. ICCV*, 2021. 2, 7
- [15] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 3
- [16] Animesh Karnewar, Andrea Vedaldi, David Novotny, and Niloy Mitra. Holodiffusion: Training a 3D diffusion model using 2D images. In *CVPR*, 2023. 3
- [17] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7
- [18] Gang Li, Heliang Zheng, Chaoyue Wang, Chang Li, Changwen Zheng, and Dacheng Tao. 3ddesigner: Towards photorealistic 3d object generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2211.14108*, 2022. 3
- [19] Kai-En Lin, Lin Yen-Chen, Wei-Sheng Lai, Tsung-Yi Lin, Yi-Chang Shih, and Ravi Ramamoorthi. Vision transformer for nerf-based view synthesis from a single input image. In *WACV*, 2023. 2, 7, 8
- [20] 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, 2023. 2, 3
- [21] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *CVPR*, June 2021. 3
- [22] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Realfusion: 360° reconstruction of any object from a single image. *arXiv preprint arXiv:2302.10663*, 2023. 3
- [23] 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 *Proc. ECCV*, 2020. 2
- [24] Norman Müller, , Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Peter Kotschieder, and Matthias Nießner. Diffrr: Rendering-guided 3d radiance field diffusion. *arxiv*, 2022. 3, 8
- [25] Norman Müller, Andrea Simonelli, Lorenzo Porzi, Samuel Rota Bulò, Matthias Nießner, and Peter Kotschieder. Autorf: Learning 3d object radiance fields from single view observations. *CoRR*, abs/2204.03593, arXiv.cs. 2
- [26] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022. 3
- [27] 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. 3, 6
- [28] Konstantinos Rematas, Ricardo Martin-Brualla, and Vittorio Ferrari. ShaRF: Shape-conditioned radiance fields from a single view. In *Proc. ICML*, 2021. 2
- [29] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *Proc. NeurIPS*, 2021. 3
- [30] J. Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. *arxiv*, 2022. 3, 4, 8
- [31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. ICLR*, 2015. 7
- [32] Vincent Sitzmann, Semon Rezhikov, William T. Freeman, Joshua B. Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. In *Proc. NeurIPS*, 2021. 2, 7

- [33] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *Advances in Neural Information Processing Systems*, 2019. 6, 7
- [34] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv:2010.02502*, October 2020. 7
- [35] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Generalizable patch-based neural rendering. In *European Conference on Computer Vision*. Springer, 2022. 3
- [36] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Light field neural rendering. In *CVPR*, 2022. 3
- [37] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proc. CVPR*, 2022. 6
- [38] Shubham Tulsiani, Tinghui Zhou, Alexei A. Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3
- [39] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. *arXiv*, 2022. 3
- [40] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snively, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *CVPR*, 2021. 3
- [41] 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. *arXiv.cs, abs/2212.06135*, 2022. 4, 8
- [42] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arxiv*, 2022. 2, 3, 7
- [43] Jiajun Wu, Chengkai Zhang, Xiuming Zhang, Zhoutong Zhang, William T. Freeman, and Joshua B. Tenenbaum. Learning shape priors for single-view 3D completion and reconstruction. *ECCV*, 2018. 1, 3
- [44] Shangzhe Wu, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. DOVE: Learning deformable 3d objects by watching videos. *arXiv*, 2021. 3
- [45] Shangzhe Wu, Ruining Li, Tomas Jakab, Christian Rupprecht, and Andrea Vedaldi. MagicPony: Learning articulated 3d animals in the wild. 2023. 3
- [46] Dejia Xu, Yifan Jiang, Peihao Wang, Zhiwen Fan, Humphrey Shi, and Zhangyang Wang. SinNeRF: Training neural radiance fields on complex scenes from a single image. In *Proc. ECCV*, 2022. 2
- [47] Xinchun Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 3
- [48] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. PixelNeRF: Neural radiance fields from one or few images. In *Proc. CVPR*, 2021. 2, 6, 7, 9
- [49] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 7
- [50] Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *arXiv*, 2022. 3