# PoseDiff: Pose-conditioned Multimodal Diffusion Model for Unbounded Scene Synthesis from Sparse Inputs

Seoyoung Lee[1*] Joonseok Lee[2,3†]
[1]The University of Texas at Austin [2]Seoul National University [3]Google Research
seoyounglee@utexas.edu, joonseok@snu.ac.kr

## Abstract

*Novel view synthesis has been heavily driven by NeRF-based models, but these models often hold limitations with the requirement of dense coverage of input views and expensive computations. NeRF models designed for scenarios with a few sparse input views face difficulty in being generalizable to complex or unbounded scenes, where multiple scene content can be at any distance from a multi-directional camera, and thus generate unnatural and low quality images with blurry or floating artifacts. To accommodate the lack of dense information in sparse view scenarios and the computational burden of NeRF-based models in novel view synthesis, our approach adopts diffusion models. In this paper, we present PoseDiff, which combines the fast and plausible generation ability of diffusion models and 3D-aware view consistency of pose parameters from NeRF-based models. Specifically, PoseDiff is a multimodal pose-conditioned diffusion model applicable for novel view synthesis of unbounded scenes as well as bounded or forward-facing scenes with sparse views. PoseDiff renders plausible novel views for given pose parameters while maintaining high-frequency geometric details in significantly less time than conventional NeRF-based methods.*

## 1. Introduction

The synthesis of photorealistic images is a popular research topic in computer vision and graphics. The objective of novel view synthesis is to render a scene from unseen viewpoints when a certain set of observed viewpoints are given. Recently, this task has increasingly gained spotlight in the community [3, 14, 21] along with the success of coordinate-based neural representations [34, 43, 36, 7], such as Neural Radiance Fields (NeRF) [38]. NeRFs learn to effectively represent objects and scenes in a 3D space, by parameterizing the per-coordinate volumetric density and color of a scene with the weights of a multilayer perceptron (MLP). With this simple yet effective architecture, NeRF models have emerged as powerful representations for novel view synthesis, demonstrating state-of-the-art performance.

However, most existing NeRF-based models [38, 65, 32, 1, 42, 53, 6, 53, 18, 2] require a dense and large-scale coverage of the scene as input to achieve the reportedly high quality performance. This causes practical issues in various applications, such as robotics, VR, and autonomous driving, where input is often very sparse with only one to few views available per object or scene of interest. It can also be a problem as large-scale real datasets often entail issues related to human or societal biases, copyright, and privacy.

In order to circumvent the need for dense scene coverage, various approaches [5, 8, 20, 29, 22, 64, 48, 60, 56, 31, 62, 12, 53, 35, 22] have been proposed. Many of these models are first expensively pretrained for the same task on a large-scale multi-view dataset with many scenes, then finetuned for a sparse set of images for a specific scene. While these models demonstrate relatively superior results, they involve challenges including obtaining a large enough pretraining dataset and reaching generalizability across various novel domains at test time.

Opposed to the pretraining-finetuning approach, *test-time optimization* approaches [10, 63, 49, 33, 19, 41, 26, 28, 52, 9] optimize their networks from scratch, solely using the given images of a particular scene. Often with extra supervision (*e.g.*, depth) and regularization techniques, these approaches extend generalizability of the models to various viewpoints. Yet, they are limited as they rely heavily on external supervision [10, 63, 49] which is not always available, or are viable only for rendering in low-resolution or simple scenes (*e.g.*, with single objects in the center of the scene, with uniform backgrounds, or synthetic scenes) [63, 26], contrary to realistic in-the-wild scenes.

Particularly, previous sparse-view-based models struggle to generate photorealistic novel views for complex or *unbounded* scenes, where the camera may point at any direction of the scene with more than one scene content located at an arbitrary distance from the camera. As shown

*Work done at Seoul National University.
†Corresponding author

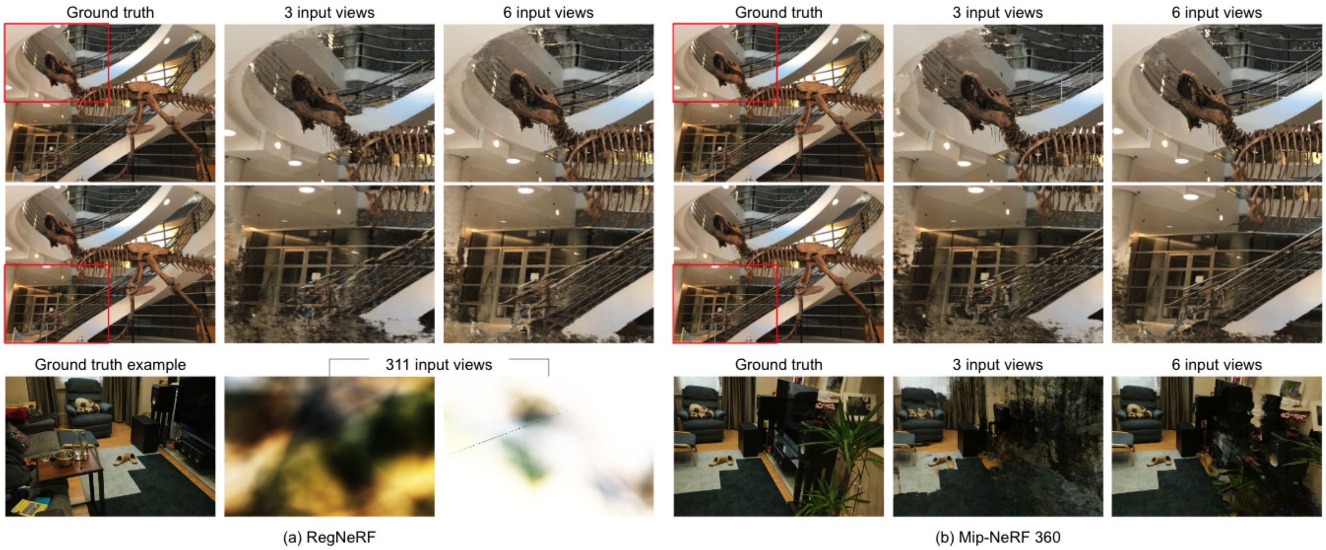


Figure 1. **Failure examples of previous models:** (a) Models for sparse input views (*e.g.*, RegNeRF [41]) are unable to model high-frequency details, especially in the peripheral areas (top example), and fail to render any meaningful views for unbounded scenes, even with a large number of input views (bottom example). (b) Models designed to handle unbounded scenes (*e.g.*, Mip-NeRF 360 [2]) also struggle similarly with significantly reduced input views.

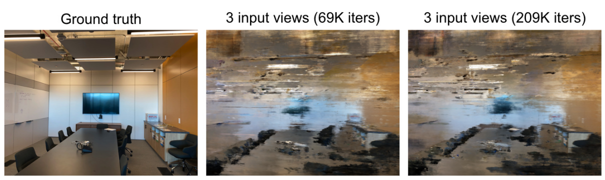


Figure 2. Few-view based models often result in low quality renderings without purposefully injecting a deterministic inductive bias for object centeredness.

in Fig. 1(a), models for sparse input views (RegNeRF [41]) find difficulty in complex scenes with high-frequency details and content in the periphery of the scene (top) and unbounded scenes (bottom), resulting in unclear and inconsistent floating artifacts. Similar observations can be found in Fig. 1(b) with models that reconstruct unbounded scenes with dense input views (*e.g.*, Mip-NeRF 360 [2]), with drastically reduced number of input views. Moreover, without the intentionally designed supervision with a deterministic inductive bias based on the main object being in the center of the scene, few-view based models often fail to converge to a level of photorealistic rendering. This phenomenon was especially prominent in extremely sparse scenarios (*e.g.*, 3 or 6 input views), even for very simple scenes regardless of the number of training iterations, as shown in Fig. 2. Overall, it can be summarized that models still struggle to learn from the sparse information of images and poses.

Another issue with NeRF-based models is the painfully expensive and long computation times necessary to train and inference the models. Although the results may not be favorable, as demonstrated above, it may still take up to several days to train a NeRF model for a single scene.

Therefore, in order to fill in the sparse information and reduce computation times in conventional approaches to sparse view based novel view synthesis, we propose to utilize the ability of diffusion models in generation based on common sense and prior knowledge. Specifically, we plan to take advantages of the generative powers in formulating plausible views and relatively shorter computation times of diffusion models, while maintaining the strength of NeRFs in modeling with 3D global view consistency by leveraging pose parameters from NeRF.

In this paper, we present *PoseDiff*, a novel method to generate realistic novel views for unbounded scenes from sparse inputs, and our main contributions:

- a *3D-aware diffusion model* conditioned on camera pose parameters, that can augment information on unseen views in sparse input scenarios.
- a notable *reduction in training and inference time* for novel view synthesis, especially of *unbounded scenes*.
- a resultant *reduction in unnatural rendering outcomes with floating artifacts*, with the synthesis of plausible and realistic novel views.

## 2. Related Work

**Sparse View Based Novel View Synthesis.** One way to handle the lack of dense input information is to take advantage of prior knowledge accumulated with models pretrained for similar tasks on larger datasets with dense multiple views of scenes [5, 8, 20, 29, 22, 48, 60, 56, 31, 62]. These approaches involve scene priors learned via an array of methods, such as self-supervision for equivariance [12] and cycle-consistency [35], 3D cost volume from image warping as input to a 3D CNN [5, 22], and extraction of local CNN features of images [8, 64]. While these models show impressive results, they hold limitations including the difficulties of collecting data for pretraining, curbs on the generalizability to test scene classes not seen in the pretraining, as well as additional costs in fine-tuning for each scene. In contrast, test-time optimization approaches only train on the given test scene, while using additional supervision (*e.g.*, depth) [10, 63, 49] or regularization techniques [33, 19, 41, 26, 52, 9] to generalize for the highly specific optimization space incurred by sparse information. However, these approaches are often highly dependent on external supervision data and models [44, 11, 30] that may not always be available. Moreover, some models also rely on intentional inductive biases for object-centric scenes, thereby limiting the generalizability of models to various scenes, including multi-object or unbounded scenes.

**Novel View Synthesis for Unbounded Scenes.** While initial NeRF [38] models rendered relatively simple scenes with plain backgrounds or forward-facing scenes with single centered objects, they have been extended to larger and unbounded scenes.With some approaches based on training decomposed visual components of NeRF [32, 55, 57], others focus on reparameterizing the 3D scene unbounded in all directions by concentrating on nearby content more heavily than content distant from the camera [65, 40, 2]. As these models tend to address large scenes (*e.g.*, city-scale or outdoor scenes), they need large-scale input data that densely cover a scene for high performance view synthesis.

**Diffusion Models for Image Generation.** A recently popular stream of research in computer vision is image generation with diffusion models [51, 47, 50, 25, 23]. Along with the development of large-scale language models [24, 45, 46, 4] and CLIP models [44], diffusion-based architectures have shown spectacular performances in multimodal conditional image generation and manipulation tasks, especially leveraging on the text modality. While these models have shown impressive results in 2D space and datasets, they have mostly been constrained to a single camera parameter and thus have not been able to understand or learn 3D concepts in the given datasets [17]. Attempts to apply diffusion models to 3D space have also heavily exploited larger multi-view datasets for pretraining for sparse views of very simple objects [62, 15, 66, 59], leading to questions of whether the model truly is a few-view based model and understands the 3D configurations of a given test scene.

## 3. Preliminaries

**Problem Formulation.** The task of *novel view synthesis* aims to render a scene from viewpoints previously unobserved in training. In this paper, we further narrow down our focus in two ways: 1) the number of available views $n_s$ in the training set is extremely small (*e.g.*, 3 and 6), thus sparsely covering the scene, and 2) the target scene is *unbounded*, indicating that scene contents may be at any distance from the camera, which may point at any direction, as opposed to a single object located at the center. Formally, the task takes two inputs: 1) a set $\mathcal{X} = \{\mathbf{x}^{(i)} \in \mathbb{R}^{h\times w\times 3} \,|\, i = 1, ..., n_s\}$ of observed views $\mathbf{x}^{(i)} \in \mathbb{R}^{h\times w\times 3}$, where $h$ and $w$ are the height and width of views, and 2) a set $\mathcal{P} = \{\mathbf{p}^{(i)} \in \mathbb{R}^{3\times 4} \,|\, i = 1, ..., n_s\}$ of camera pose parameters $\mathbf{p}^{(i)} = [\mathbf{r}^{(i)} \mid \mathbf{t}^{(i)}] \in SE(3)$ in the 3D Cartesian space corresponding to each $\mathbf{x}^{(i)}$, where $\mathbf{r}^{(i)} \in \mathbb{R}^{3\times 3}$ is the rotation matrix and $\mathbf{t}^{(i)} \in \mathbb{R}^{3\times 1}$ is the translation vector. The output of inference is an image $\mathbf{y} \in \mathbb{R}^{h\times w\times 3}$, a view from an unseen viewpoint or camera pose $\mathbf{p}_{\text{test}} = [\mathbf{r}_{\text{test}} \mid \mathbf{t}_{\text{test}}] \in SE(3)$ that may have not been included in $\mathcal{P}$.

**Diffusion Models.** Diffusion probabilistic models [16] are a family of generative models that aims to learn how to recover the actual distribution of the given data by reversing the forward diffusion process, where noise is gradually added to the data. In essence, the model learns a reverse Markov chain of length $T$, which can be translated into a series of $T$ denoising autoencoders [54] for $t \in \{0, ..., T\}$. Given a ground truth image $\mathbf{x}$, diffusion models are constructed as a framework where a model is first initialized as random noise $\mathbf{z}_T \sim \mathcal{N}(0, \mathbf{I})$. Then, $\mathbf{z}_T$ is iteratively denoised under a predefined diffusion schedule. This gradual learning process continues until the model is able to reconstruct $\mathbf{x}$, which is the completely denoised original image. At each intermediate optimization step $t \in \{0, ..., T\}$, an intermediate noised image $\mathbf{z}_t$ can be formulated as

$$\mathbf{z}_t = \sqrt{\alpha_t}\mathbf{x} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t, \tag{1}$$

where $1 = \alpha_0 > \alpha_1 > \cdots > \alpha_{T-1} > \alpha_T = 0$ are hyperparameters according to the diffusion noise schedule, and $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$. At each step, a denoising objective [16] guides the network $f_\theta$, which can be conditioned on an additional conditioning input $\mathbf{p} \in \mathbb{R}^d$:

$$\mathbb{E}_{\mathbf{z},\mathbf{p},t,\boldsymbol{\epsilon}_t}\left[w_t\|f_\theta(\mathbf{z}_t, t, \mathbf{p}) - \boldsymbol{\epsilon}_t\|_2^2\right], \tag{2}$$

where $w_t$ is determined by the diffusion schedule. By conditioning the network $f_\theta$ on $\mathbf{p}$, the diffusion model is able to learn the latent distribution conditioned on $\mathbf{p}$.

**Neural Radiance Fields (NeRF).** NeRF [38] captures the implicit and continuous 3D representations of static objects or scenes. The mapping from a 3D spatial coordinate $\mathbf{q} \in \mathbb{R}^3$ in the scene and viewing direction $\mathbf{d} = (\theta, \phi)$ to its corresponding volumetric density $\boldsymbol{\sigma} \in [0, \infty)$ and emitted color $\mathbf{c} = (r, g, b) \in [0, 1]^3$ is encoded into the weights of an MLP. The color of the pixel $C(\mathbf{r})$ along a camera ray $\mathbf{r}$ is estimated with the weighted sum of the color values of N sampled points along the ray, weighted by the density and accumulated transmittance as

$$\hat{C}(\mathbf{r}) = \sum_{i=0}^{N-1} T_i \left(1 - \exp(-\boldsymbol{\sigma}_i \boldsymbol{\delta}_i)\right) \mathbf{c}_i, \quad T_i = \exp\left( - \sum_{j=0}^{i-1} \boldsymbol{\sigma}_j \boldsymbol{\delta}_j \right) \tag{3}$$

where $\boldsymbol{\delta}_i$ is the distance between consecutive sampled points. NeRF is optimized by the $L_2$ loss $\mathcal{L} = \sum_{\mathbf{r}\in\mathcal{R}} \|C(\mathbf{r}) - \hat{C}(\mathbf{r})\|_2^2$ between the estimated colors $\hat{C}(\mathbf{r})$ for a random batch of rays $\mathcal{R}$ and their ground truth values.

**Mip-NeRF 360 [2].** Vanilla NeRF representations are often aliased, due to the lack of understanding in multiple scales. Mip-NeRF [1] improves NeRF to reason about scales by casting a 3D cone and introducing integrated positional encoding (IPE) to represent a volume of conical frustum or Gaussian region, as opposed to casting a point-wise ray and using positional encoding that represents an infinitesimal point. Mip-NeRF 360 [2] further extends Mip-NeRF to cover unbounded scenes with non-linear scene parametrization, online distillation, and a distortion-based regularizer. As Mip-NeRF 360 is a more appropriate NeRF representation for scenes with varying camera parameters and various objects, we develop our idea based on pose parameters used in this Mip-NeRF 360 model.

## 4. The Proposed Method: PoseDiff

We propose a novel method to generate realistic novel views with a few sparse inputs for a given unbounded scene. A diffusion model is used to augment the lack of information due to large proportions of unseen viewpoints in sparse view scenarios, and to accelerate computation times. By conditioning a diffusion model with the corresponding 3D-aware camera pose parameters and text description for the few input views, we train a pose-conditioned multimodal diffusion model that generates realistic views from certain viewpoints (Section 4.1). Then, we render a plausible set of views for a set of unseen camera poses, by inferring from the pose-conditional multimodal diffusion model trained on the original few sparse seen views (Section 4.2). The overall architecture is illustrated in Figs. 3 and 4.

### 4.1. Pose-conditioned Diffusion

The conditional diffusion model focuses on learning the relationship between 3D camera configurations and the corresponding views in a localized latent subspace relevant to our scene of interest. Thus, we are able to supplement the lack of information on unseen views in the current training set. As shown in Fig. 3, this module takes three inputs.

Firstly, a small number $n_s$ (*e.g.*, 3, 6) of images $\mathcal{X} = \{\mathbf{x}^{(i)} \in \mathbb{R}^{h\times w\times 3} \,|\, i = 1, ..., n_s\}$ showing differing views of a single target scene, where $h, w$ are the height and width values of the images, are encoded with a Variational Autoencoder (VAE) model with KL loss [27] into the mean and log variance values of a diagonal Gaussian distribution. We then sample latents of the images from each respective diagonal Gaussian distribution and apply random noise to form noised images $\mathbf{z}^{(i)}$ ($i \in \{1, ..., n_s\}$). Secondly, a representative text prompt $\mathbf{t}_r$ with a customized token `[S*]` (*e.g.*, `"zwx"`) inspired by [13] to describe the scene in $\mathcal{X}$ (*e.g.*, `"a zwx room"`) is converted into a tokenized text embedding $\mathbf{e}_r \in \mathbb{R}^{l\times d}$, where $l$ is the number of tokens in the text prompt and $d$ is the embedding dimension per token, with a pre-trained CLIP text encoder [44]. The special token helps to localize the latent subspace relevant to our specific test scene among other similar class instances, while leveraging prior knowledge of text embedding models. Lastly, $n_s$ pairs of camera poses $\mathbf{p}^{(i)} = [\mathbf{r}^{(i)} \mid \mathbf{t}^{(i)}] \in SE(3)$ for each $\mathbf{x}^{(i)}$ are processed into rays per pixel of each image, represented as a vector with an origin and direction. The origin and direction values are concatenated to form a set $\mathcal{P}' = \{\gamma(\mathbf{p}^{(i)}) \,|\, i = 1, ..., n_s\}$ of camera poses.

The noised images $\mathbf{z}^{(i)}$ ($i \in \{1, ..., n_s\}$) and camera pose parameters $\gamma(\mathbf{p}^{(i)})$ ($i \in \{1, ..., n_s\}$) are concatenated to form the input of a conditional 2D UNet. The UNet and the conditional preprocessed text embeddings $\mathbf{e}_r$ are used to train a generative latent diffusion model $f_\theta$. Along the progress of the diffusion process with $T$ optimization steps, each initial noised image $\mathbf{z}_T^{(i)}$ is iteratively refined via $T$ time steps into $\mathbf{z}_t^{(i)} (t \in \{0, ..., T\})$ until the ground truth image $\mathbf{z}_0^{(i)} = \mathbf{x}^{(i)}$ is realized. The diffusion model is optimized with a $L_2$ reconstruction loss [16] for each $i \in \{1, ..., n_s\}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$:

$$\mathcal{L}(\mathbf{z}^{(i)}, \mathbf{e}_r, \gamma(\mathbf{p}^{(i)}), \theta) = \mathbb{E}_{\mathbf{z}^{(i)}, \mathbf{e}_r, \gamma(\mathbf{p}^{(i)}), t, \boldsymbol{\epsilon}} \left[ \left\| f_\theta(\mathbf{z}_t^{(i)}, \mathbf{e}_r, \gamma(\mathbf{p}^{(i)})) - \boldsymbol{\epsilon} \right\|_2^2 \right].$$

### 4.2. Inference of Unseen Views

By using the pose-conditioned multimodal generative diffusion model $f_\theta$ trained in Section 4.1, this step aims to create realistic novel views from previously unseen camera poses. As shown in Fig. 4, the previously trained $f_\theta$ is used to infer $n_u$ plausible views, where each view $\mathbf{y} \in \mathbb{R}^{h\times w\times 3}$ is inferenced from a camera configuration $\mathbf{p}_{\text{test}} = [\mathbf{r}_{\text{test}} \mid \mathbf{t}_{\text{test}}] \in SE(3)$ that may have not been included in $\mathcal{P}'$ for the $n_s$ given views of the scene. Unlike $n_s$, which was a small number, $n_u$ can be any number selected by the user. Various tactics can be used to sample previously unobserved viewpoints. In our experiments, we follow the random sampling technique used in Mip-NeRF 360 [2] to select unobserved viewpoints in various trajectories.

As a result of inferring $n_u$ unseen views with the diffusion model $f_\theta$, trained specifically with our test scene, we are able to construct a larger dataset of size $n = n_s + n_u$ that densely covers the scene. While the $n_u$ inferred views for unseen viewpoints may not be identical to the actual ground truth, the resulting images will still show plausible views rather than foggy or floating artifacts, due to the plausible generation capabilities of diffusion models.

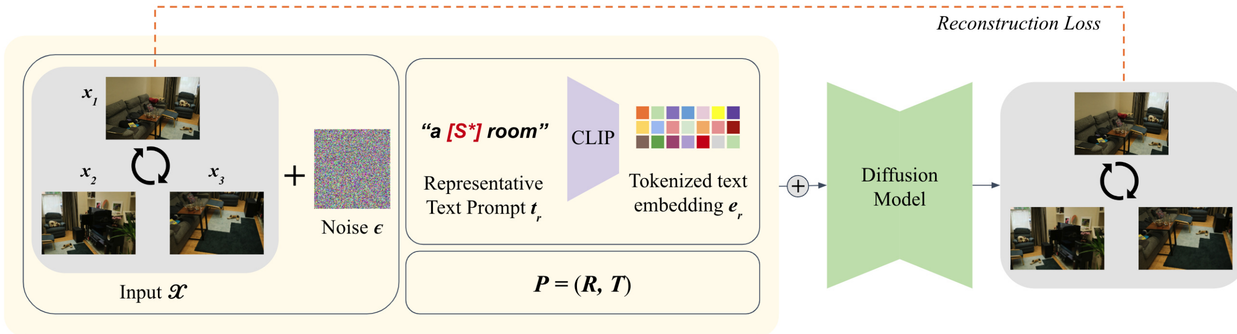


Figure 3. **Overview of our pose-conditioned multimodal diffusion model.** Given a few sparse views of a scene, respective camera pose parameters for each view, and a text description per scene, a diffusion model is trained to reconstruct the given views.

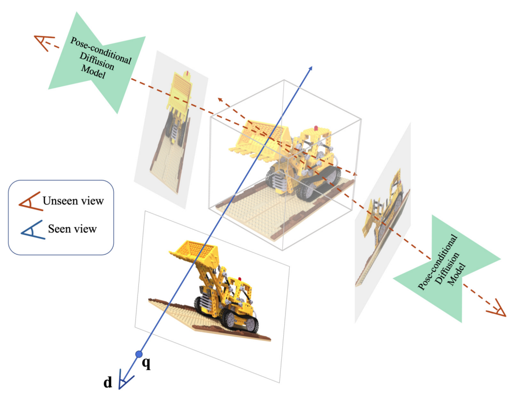


Figure 4. **Inference of unseen views**. Based on the pose-conditioned multimodal diffusion model trained on seen views, we inference plausible views from unseen viewpoints.

| # of Views | Model | Training Epochs | Training Time | Inference Time per Image |
|---|---|---|---|---|
| 3 | RegNeRF | 69,768 | 9 hrs | 13.84 secs |
| | Mip-NeRF 360 | 500,000 | 58 hrs | 6.62 secs |
| | **Ours** | **800** | **6 mins** | **4 secs** |
| 6 | RegNeRF | 139,535 | 16 hrs | 12.09 secs |
| | Mip-NeRF 360 | 500,000 | 58 hrs | 6.05 secs |
| | **Ours** | **1000** | **10 mins** | **4 secs** |

Table 1. **Computation times of models for each number of input views used for training.** All computation times were measured when using one A6000 GPU for one experiment configuration.

# 5. Experiments

## 5.1. Experimental Settings

**Datasets.** We verify our method on two datasets: LLFF [37] for forward-facing scenes and 360 dataset [2] for unbounded scenes. The 360 dataset consists of unbounded scenes with complex objects and a detailed background, taken from various angles and distances.

**Baselines.** We perform quantitative and qualitative comparisons with various experiment configurations against baselines including RegNeRF [41] and Mip-NeRF 360 [2]. We first compare our results to RegNeRF, a sparse-view model, with varying numbers of training input, with and without the inductive bias for objects being in the center of the scene. Next, we compare our method with Mip-NeRF 360, a dense-view model intended for unbounded scenes, in terms of results from different training epochs and time, with varying size of input data.

**Evaluation Metrics.** Our method is evaluated both quantitatively and qualitatively. Quantitatively, we use PSNR and SSIM [61] metrics to assess the quality of our generated results against the ground truth views and baseline results. For qualitative evaluation, we demonstrate the degree of resolution and realism of the synthesized views.

**Implementation Details.** Our model is built upon the text-to-image latent diffusion models [58] and Mip-NeRF 360 [39] for extracting the camera pose parameters of each image. For training the latent diffusion model, we increase the input channel size of the UNet to 10 to accommodate the additional camera pose parameters. The images are not randomly transformed, but rather only converted to tensors and normalized for better alignment with image-wise camera poses. We set the learning rate to $1e^{-4}$ for training our pose-conditioned diffusion model, with 800 to 1000 training epochs used for 3 and 6 input views. Additional details on training and inference can be found in Table 1. We use a single NVIDIA RTX A6000 GPU for training and inference. All other hyperparameters related to the latent diffusion model follow the same setting from the original paper.

## 5.2. Qualitative Evaluations

We demonstrate some novel view synthesis results obtained by our model and baselines for qualitative comparison. As shown in Fig. 5, our model is able to perform novel view synthesis in both unbounded scenes and forward-facing scenes. Compared to the state-of-the-art few-view based model (RegNeRF), our model is able to ren-

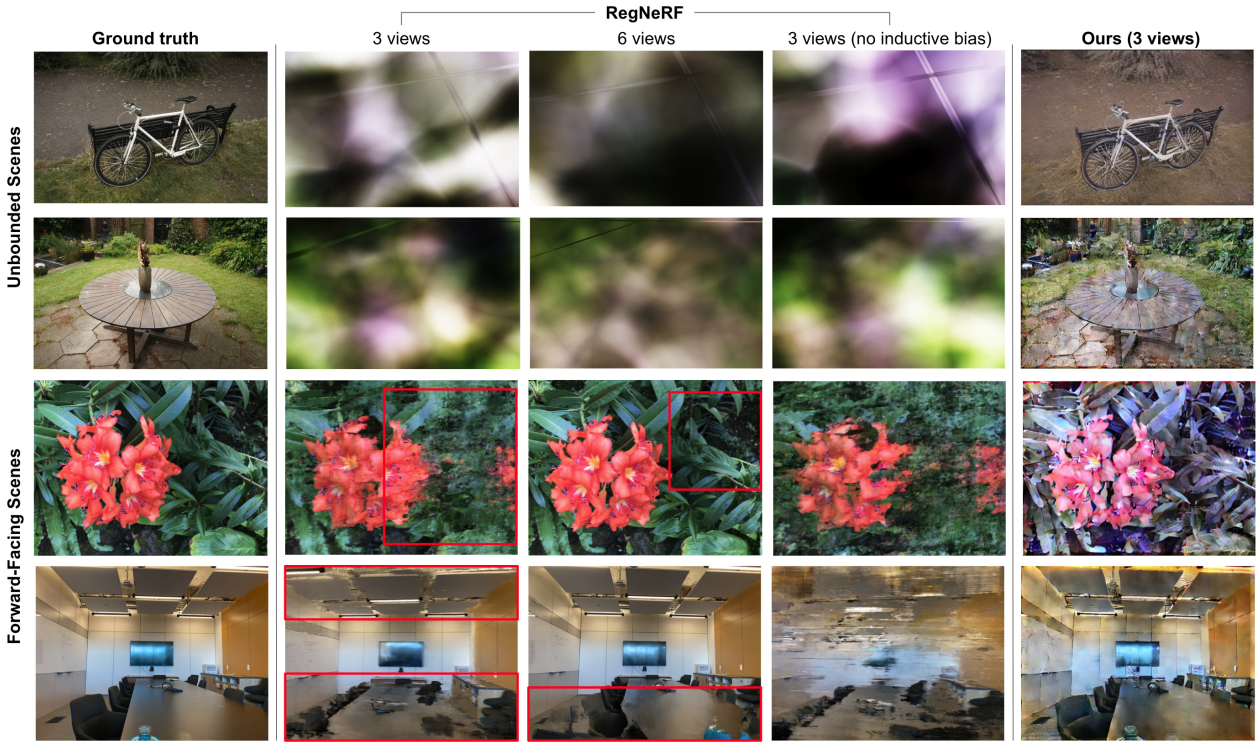

Figure 5. **Comparison with sparse-view based NeRF model results**. Our model is able to render plausible novel views for all scene types tested with a few input views. While RegNeRF failed to render unbounded scenes or without inductive bias injection in general, it showed better renderings with forward-facing scenes, albeit with blurry geometric details.

der plausible novel views of the unbounded scenes, while RegNeRF fails to render a tangible scene in all experiment scenarios tested for unbounded scenes. On the other hand, RegNeRF manages to create much concrete views for simpler forward-facing scenes. However, although RegNeRF catches the colors of the scene better, it fails to capture the high-frequency details of the scene, such as the leaves surrounding the flowers. This observation is aggravated when it comes to objects in the periphery of the scene. While scene content towards the center of the scene are well managed by RegNeRF with more input views, it is unable to perform at the same level when the intentional inductive bias to enforce object centeredness is removed. On the contrary, our model is able to clearly capture the geometric details of the scene even without any inductive biases injected in the training process. Unlike most NeRF-based models, our model does not render any floating or unnatural artifacts in the output images. A downside of our model is the slightly inaccurate color and texture observed in some of the output.

Fig. 6 compares the results on unbounded scenes from our model and Mip-NeRF 360, a model for unbounded scenes based on dense views, at various training steps. When fully trained on extremely sparse inputs, our model is able to render relatively realistic novel views of the given test scene, while Mip-NeRF 360 fails to converge on a clear image that preserves the geometric structures of the scene. Whereas Mip-NeRF 360 struggles to capture the high-frequency details of the scene overall, our model finds difficulty in precisely capturing the colors, for some of the scenes where geometric details are maintained well.

It is noteworthy, however, that Mip-NeRF 360 needs drastically longer training times than ours. With comparatively much shorter training times, our model is able to achieve superior visual performances, compared to the conventional NeRF-based model, even with just 3 views. As shown in Fig. 6, even after 2 hours (25,000 epochs) of training, Mip-NeRF 360 is still unable to render high-definition images for all input views and experiment configurations tested. On the other hand, our model renders much clearer scenes only after 6 minutes of training on 3 views.

### 5.3. Quantitative Evaluations

Tables 3 and 4 compare the scores of PSNR and SSIM [61] of baselines and our model for few-input scenarios on both unbounded scenes and simpler forward-facing scenes. We evaluate renderings from models trained on 3 and 6 input images, which are significantly less than the usual number of images used to train dense-view models, as shown in Table 2. Throughout most experiments, our model greatly outperforms baselines in terms of PSNR scores, proving the high quality of our renderings compared to the baselines as shown in Figs. 5 and 6. However, our models

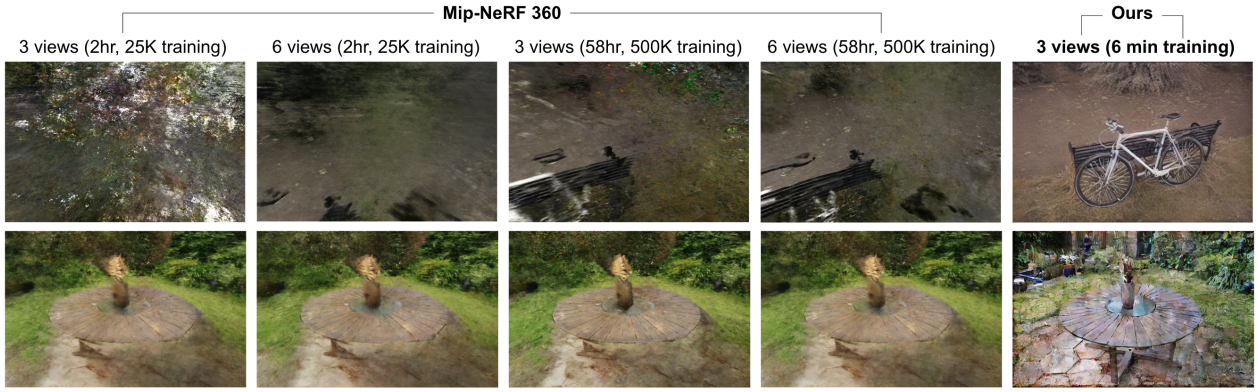


Figure 6. **Comparison of results with dense-view based NeRF model designed for unbounded scenes**. Our model shows plausible renderings that capture high-frequency details from a few sparse input views, at a much shorter training time compared to Mip-NeRF 360.

| Dataset | Scene | Train Set | Test Set | Total |
|---|---|---|---|---|
| 360 Dataset (Unbounded Scenes) | Bicycle | 170 | 24 | 194 |
| | Garden | 162 | 23 | 185 |
| | Kitchen | 245 | 34 | 279 |
| LLFF (Forward-facing Scenes) | Room | 36 | 5 | 41 |
| | Flower | 30 | 4 | 34 |
| | T-rex | 49 | 6 | 55 |

Table 2. **Common dataset sizes used in dense coverage models.** By default, most models typically take every 8th image in the whole dataset as a test image.

do not particularly excel in SSIM scores. As SSIM considers contrast in images as a major part of the metric, it seems to weigh down on the difference in color that our rendered images show in contrast to the ground truth images.

Moreover, we compare the training and inference costs of the baselines and our model in Table 1. With the same computational resources, our model takes several orders of magnitude less in time for training, and only takes around 2/3 or 1/4 of the time for inference, compared to baselines.

### 5.4. Ablation Studies

**Specialized Text Tokens.** In this section, we demonstrate the benefits of using a specialized text token as opposed to text tokens consisting of pure ordinary natural language words. Fig. 7 shows images generated from text-driven latent diffusion models. When simply given a general noun to describe the object or scene (*e.g.* `"trex"`), the generated images show a wide variability in the resulting content of the scene. On the other hand, when we use a randomly generated special text token `[T*]` (*e.g.*, `"zwx"`) to describe the instance scene of interest, the generated results are consistent with the test scene image used for generation.

This difference in image generation capability may be interpreted as the difference in the latent spaces that the diffusion model lives on when making generations. As shown in Fig. 8, our diffusion model is guided specifically towards


Figure 7. **Effect of using a specialized text token.** A particular descriptive text token (*e.g.*, `"zwx"`) to describe our instance can be used to overfit to our scene of interest and only generate scenes similar to the input image.

the green latent space relevant to a particular instance, noted with a special token `[T*]` as `"A [T*] trex"`. This would make the learning take place on or near the latent space specifically related to our particular instance trex. This allows the model to understand relevant semantic embeddings that do not lie far from the given instance. In contrast, general diffusion models [23] generate a fine-tuned model for a general class of instances (*e.g.*, `"A trex"`) and make inferences in that larger and more general latent space, such as the yellow space in Fig. 8. Thus, the generated results may not necessarily be related to the specific source instance scene we wish to generate, as long as they are relevant to any instance from the 'trex' class.

**Pose-conditioning.** Fig. 9 shows the effect of utilizing pose to condition the text-to-image latent diffusion models. When only given the text conditioning with the specialized text tokens (above 2 rows), the model only generates images that are seen from a relatively uniform camera direction or pose. However, when pose is additionally used to condition the diffusion model as done in our model (bottom row), the model generates novel views that consider the various viewpoints, from which the scene can be viewed. The rendered

| Dataset | Scene | RegNeRF | | | | | | Ours | |
|---|---|---|---|---|---|---|---|---|---|
| | | 3 views | | 6 views | | No inductive bias (3 views) | | 3 views | |
| | | PSNR(↑) | SSIM(↑) | PSNR(↑) | SSIM(↑) | PSNR(↑) | SSIM(↑) | PSNR(↑) | SSIM(↑) |
| 360 Dataset | Bicycle | 6.84 | 0.306 | 12.62 | 0.396 | 6.96 | 0.309 | **27.88** | 0.258 |
| | Garden | 8.47 | 0.376 | 12.76 | 0.429 | 8.16 | 0.352 | **28.04** | 0.206 |
| LLFF | Flower | 19.72 | 0.677 | 23.81 | 0.849 | 15.26 | 0.440 | **27.96** | 0.410 |
| | Room | 21.04 | 0.860 | **29.21** | 0.951 | 15.52 | 0.630 | 28.81 | 0.695 |

Table 3. **Comparison with RegNeRF results.** The top PSNR score for each experiment configuration is emphasized in bold.

| Dataset | Scene | Mip-NeRF 360 | | | | Ours | |
|---|---|---|---|---|---|---|---|
| | | 3 views | | 6 views | | 3 views | |
| | | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| 360 Dataset | Bicycle | 13.40 | 0.134 | 14.58 | 0.182 | **27.88** | 0.258 |
| | Garden | 17.88 | 0.374 | 17.87 | 0.378 | **28.04** | 0.206 |

Table 4. **Comparison with Mip-NeRF 360 results on unbounded scenes.** Top scores for PSNR are emphasized in bold.

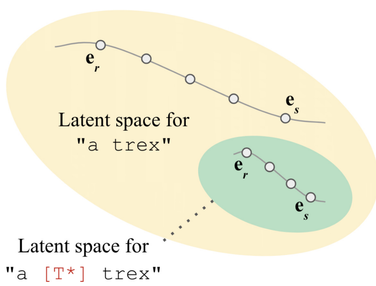


Figure 8. **A diagram of the latent training in diffusion models.** Shown in a rough diagram, our model learns a targeted latent space (green) particular for the instance scene, while general latent diffusion models [23] often live on a larger and more general latent space (yellow) for the class that the instance scene belongs to.


Figure 9. **Effect of pose-conditioning**. By using pose as an additional condition, our diffusion model is able to generate novel views from various camera pose configurations.

images are also much more realistic, without any strange objects (*e.g.*, an object that is a fusion of a bench and a bicycle, a bicycle in a bench, or half a bicycle) as in the text-to-image diffusion models conditioned only on text. Thus, we conclude that using camera pose to condition a diffusion model does help the model to understand and reason about 3D-aware camera viewpoints.

## 6. Limitations & Future Work

Although our model has achieved to render realistic scenes with high-frequency details by using significantly less training costs, renderings from our model sometimes show inaccurate and rather artistic results. This seems to be due to our leveraging of the generative powers inherent in diffusion models. Moreover, our model requires a careful design of hyperparameters for each experiment condition. We leave methods for color and appearance regularization and potentially learnable methods to determine the necessary hyperparameters for scalable experimentation settings as promising future work.

## 7. Conclusion

We have presented *PoseDiff*, a method to generate novel views for unbounded scenes with a few sparse inputs. In order to supplement the sparse information from few input images in sparse view scenarios and the long computation times of conventional methods for novel view synthesis, we utilize latent diffusion models conditioned on pose parameters from NeRF and text descriptions. In this process, the proposed model was able to show synergy between the fast computations and generative capabilities of diffusion models and the ability of pose parameters from NeRF to maintain global view-consistency. As a result, we were able to synthesize novel viewpoints of a scene that preserve high-frequency geometric details with computation times that were several orders of magnitude less than baselines. We identify methods for color contrast improvement and learnable hyperparameter tuning for scalability as potential future research directions.

**Acknowledgement.** This work was supported by the New Faculty Startup Fund from Seoul National University and by National Research Foundation (NRF) grant (No. 2021H1D3A2A03038607/50%, 2022R1C1C1010627/20%, RS-2023-00222663/10%) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant (No. 2022-0-00264/20%) funded by the government of Korea.

## References


[1] 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 *ICCV*, 2021.

[2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022.

[3] Alexander Bergman, Petr Kellnhofer, and Gordon Wetzstein. Fast training of neural lumigraph representations using meta learning. *NeurIPS*, 34:172–186, 2021.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 33:1877–1901, 2020.

[5] 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 *ICCV*, 2021.

[6] Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *CVPR*, 2022.

[7] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, 2019.

[8] Julian Chibane, Aayush Bansal, Verica Lazova, and Gerard Pons-Moll. Stereo Radiance Fields (SRF): Learning view synthesis for sparse views of novel scenes. In *CVPR*, 2021.

[9] Congyue Deng, Chiyu Jiang, Charles R Qi, Xinchen Yan, Yin Zhou, Leonidas Guibas, Dragomir Anguelov, et al. NeRDi: Single-view nerf synthesis with language-guided diffusion as general image priors. In *CVPR*, 2023.

[10] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised NeRF: Fewer views and faster training for free. In *CVPR*, 2022.

[11] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. *arXiv:1605.08803*, 2016.

[12] Emilien Dupont, Miguel Bautista Martin, Alex Colburn, Aditya Sankar, Josh Susskind, and Qi Shan. Equivariant neural rendering. In *ICML*, 2020.

[13] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv:2208.01618*, 2022.

[14] Chen Gao, Yichang Shih, Wei-Sheng Lai, Chia-Kai Liang, and Jia-Bin Huang. Portrait neural radiance fields from a single image. *arXiv:2012.05903*, 2020.

[15] Jiatao Gu, Alex Trevithick, Kai-En Lin, Joshua M Susskind, Christian Theobalt, Lingjie Liu, and Ravi Ramamoorthi. NerfDiff: Single-image view synthesis with nerf-guided distillation from 3D-aware diffusion. In *ICML*. PMLR, 2023.

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020.

[17] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022.

[18] Tao Hu, Shu Liu, Yilun Chen, Tiancheng Shen, and Jiaya Jia. EfficientNeRF: efficient neural radiance fields. In *CVPR*, 2022.

[19] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting NeRF on a diet: Semantically consistent few-shot view synthesis. In *ICCV*, 2021.

[20] Wonbong Jang and Lourdes Agapito. CodeNeRF: Disentangled neural radiance fields for object categories. In *ICCV*, 2021.

[21] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker. SDFDiff: Differentiable rendering of signed distance fields for 3D shape optimization. In *CVPR*, 2020.

[22] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. GeoNeRF: Generalizing nerf with geometry priors. In *CVPR*, 2022.

[23] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *arXiv:2210.09276*, 2022.

[24] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

[25] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. DiffusionCLIP: Text-guided diffusion models for robust image manipulation. In *CVPR*, 2022.

[26] Mijeong Kim, Seonguk Seo, and Bohyung Han. InfoNeRF: Ray entropy minimization for few-shot neural volume rendering. In *CVPR*, 2022.

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013.

[28] Seoyoung Lee, Seongsu Ha, and Joonseok Lee. Disentangled audio-driven NeRF: Talking head generation with detailed identity-specific microexpressions. In *CVPRW*, 2023.

[29] Jiaxin Li, Zijian Feng, Qi She, Henghui Ding, Changhu Wang, and Gim Hee Lee. Mine: Towards continuous depth MPI with NeRF for novel view synthesis. In *ICCV*, 2021.

[30] Kai-En Lin, Yen-Chen Lin, 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.

[31] 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 *CVPR*, 2022.

[32] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the wild: Neural radiance fields for unconstrained photo collections. In *CVPR*, 2021.

[33] Quan Meng, Anpei Chen, Haimin Luo, Minye Wu, Hao Su, Lan Xu, Xuming He, and Jingyi Yu. GNeRF: GAN-based neural radiance field without posed camera. In *ICCV*, 2021.

[34] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *CVPR*, 2019.

[35] Lu Mi, Abhijit Kundu, David Ross, Frank Dellaert, Noah Snavely, and Alireza Fathi. im2NeRF: Image to neural radiance field in the wild. *arXiv:2209.04061*, 2022.

[36] Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *ICCV*, 2019.

[37] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019.

[38] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[39] Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, Peter Hedman, Ricardo Martin-Brualla, and Jonathan T. Barron. MultiNeRF: A Code Release for Mip-NeRF 360, Ref-NeRF, and RawNeRF, 2022.

[40] Thomas Neff, Pascal Stadlbauer, Mathias Parger, Andreas Kurz, Joerg H Mueller, Chakravarty R Alla Chaitanya, Anton Kaplanyan, and Markus Steinberger. DONeRF: Towards real-time rendering of compact neural radiance fields using depth oracle networks. In *Computer Graphics Forum*, volume 40, pages 45–59. Wiley Online Library, 2021.

[41] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. RegNeRF: Regularizing neural radiance fields for view synthesis from sparse inputs. In *CVPR*, 2022.

[42] Julian Ost, Fahim Mannan, Nils Thuerey, Julian Knodt, and Felix Heide. Neural scene graphs for dynamic scenes. In *CVPR*, 2021.

[43] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019.

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[45] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[46] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research (JMLR)*, 21(1):5485–5551, 2020.

[47] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv:2204.06125*, 2022.

[48] Konstantinos Rematas, Ricardo Martin-Brualla, and Vittorio Ferrari. ShaRF: Shape-conditioned radiance fields from a single view. *arXiv:2102.08860*, 2021.

[49] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *CVPR*, 2022.

[50] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[51] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv:2205.11487*, 2022.

[52] Seunghyeon Seo, Donghoon Han, Yeonjin Chang, and Nojun Kwak. MixNeRF: Modeling a ray with mixture density for novel view synthesis from sparse inputs. In *CVPR*, 2023.

[53] Yeji Song, Chaerin Kong, Seoyoung Lee, Nojun Kwak, and Joonseok Lee. Towards efficient neural scene graphs by learning consistency fields. In *BMVC*, 2022.

[54] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv:2011.13456*, 2020.

[55] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-NeRF: Scalable large scene neural view synthesis. In *CVPR*, 2022.

[56] Alex Trevithick and Bo Yang. GRF: Learning a general radiance field for 3D representation and rendering. In *ICCV*, 2021.

[57] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-NeRF: Scalable construction of large-scale nerfs for virtual fly-throughs. In *CVPR*, 2022.

[58] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

[59] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score Jacobian chaining: Lifting pretrained 2D diffusion models for 3D generation. In *CVPR*, 2023.

[60] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. IBRNet: Learning multi-view image-based rendering. In *CVPR*, 2021.

[61] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[62] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv:2210.04628*, 2022.

[63] Dejia Xu, Yifan Jiang, Peihao Wang, Zhiwen Fan, Humphrey Shi, and Zhangyang Wang. SinNeRF: Training neural radi-

ance fields on complex scenes from a single image. In *ECCV*, 2022.

[64] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. PixelNeRF: Neural radiance fields from one or few images. In *CVPR*, 2021.

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

[66] Zhizhuo Zhou and Shubham Tulsiani. SparseFusion: Distilling view-conditioned diffusion for 3D reconstruction. In *CVPR*, 2023.