Sp^{2}360: Sparse-view 360° Scene Reconstruction using Cascaded 2D Diffusion Priors

Soumava Paul, Christopher Wewer, Bernt Schiele, and Jan Eric Lenssen

Max Planck Institute for Informatics, Germany
soumava2016@gmail.com, {cwewer, schiele, jlenssen}@mpi-inf.mpg.de

Abstract. We aim to tackle sparse-view reconstruction of a 360° 3D scene using priors from latent diffusion models (LDM). The sparse-view setting is ill-posed and underconstrained, especially for scenes where the camera rotates 360 degrees around a point, as no visual information is available beyond some frontal views focused on the central object(s) of interest. In this work, we show that pretrained 2D diffusion models can strongly improve the reconstruction of a scene with low-cost fine-tuning, alleviating reliance on large-scale 3D datasets. Specifically, we present *SparseSplat360 (Sp^{2}360)*, a method that employs a cascade of in-painting and artifact removal models to fill in missing details and clean novel views. Due to superior training and rendering speeds, we use an explicit scene representation in the form of 3D Gaussians over NeRF-based implicit representations. We propose an iterative update strategy to fuse generated pseudo novel views with existing 3D Gaussians fitted to the initial sparse inputs. As a result, we obtain a multi-view consistent scene representation with details coherent with the observed inputs. Our evaluation on the challenging Mip-NeRF360 dataset shows that our proposed 2D to 3D distillation algorithm considerably improves the performance of a regularized version of 3DGS adapted to a sparse-view setting and outperforms existing sparse-view reconstruction methods in 360° scene reconstruction on traditional metrics. Qualitatively, our method generates entire 360° scenes from as few as 9 input views, with a high degree of foreground and background detail.

1 Introduction

Obtaining high-quality 3D reconstructions or novel views from a set of images has been a long-standing goal in computer vision and has received increased interest recently. Recent 3D reconstruction methods, such as those based on Neural Radiance Fields (NeRF) [37], Signed Distance Functions (SDFs) [62], or the explicit 3D Gaussian Splatting (3DGS) [29], are now able to produce photorealistic novel views of 360° scenes. However, in doing so, they rely on hundreds of input images that densely capture the underlying scene. This requirement is both time-consuming and often an unrealistic assumption for complex, large-scale scenes. Ideally, one would like a 3D reconstruction pipeline to offer generalization to unobserved parts of the scene and be able to successfully reconstruct areas that

are only observed a few times. In this work, we present a method to efficiently obtain high-quality 3D Gaussian representations from just a few views, moving towards this goal.

Standard 3DGS, much like NeRF, is crippled in a sparse observation setting. In the absence of sufficient observations and global geometric cues, 3DGS invariably overfits to training views. This leads to severe artifacts and background collapse already in nearby novel views due to the inherent depth ambiguities associated with inferring 3D structure from few-view 2D images. There exists a long line of work to improve performance of both NeRF and 3DGS in sparse novel view synthesis [11,14,26,31,41,46,60,68,69,76]. While these works can reduce artifacts in sparsely observed regions, they are not able to fill in missing details due to the use of simple regularizations or weak priors.

In the setting of large-scale 360° scenes, the problem is even more ill-posed and under-constrained. Much stronger priors are needed, such as those from large pre-trained 2D diffusion models [40,44,47,49], capturing knowledge about typical structures in the 3D world. Recent approaches [34,50,67] add view-conditioning to these image generators by fine-tuning them on a large mixture of real-world and synthetic multi-view datasets. Leveraging these strong priors for optimization of radiance fields yields realistic reconstructions in unobserved areas of challenging 360° scenes. In this work, we propose to forego augmenting a 2D diffusion model with additional channels for pose or context to make it 3D-aware. Instead, we perform low-cost fine-tuning of pretrained models to adapt them to specific sub-tasks of few-view reconstruction. This weakens the assumption of large-scale 3D training data, which is expensive to obtain.

We present SparseSplat360 (Sp^{2}360), an efficient method that addresses the given task of sparse 3D reconstruction by iteratively adding synthesized views to the training set of the 3D representation. The generation of new training views is divided and conquered as the 2D sub-tasks of (1) *in-painting missing areas* and (2) *artifact elimination*. The in-painting model for the first stage is pre-trained on large 2D datasets and efficiently fine-tuned on the sparse views of the given scene. The artifact elimination network for the second stage is an image-to-image diffusion model, fine-tuned to specialize on removing typical artifacts appearing in sparse 3DGS. Thus, both stages utilize 2D diffusion models that are fine-tuned on small amounts of data. In each iteration, the models are conditioned on rendered novel views of the already existing scene, thus using 2D images as input and output, avoiding the requirement of training on large datasets of 3D scenes. In contrast to previous works, our method leverages stronger priors than simple regularizers and does not rely on million-scale multi-view data or huge compute resources to train a 3D-aware diffusion model.

In summary, our contributions are:

- We present a novel systematic approach to perform sparse 3D reconstruction of 360° scenes by autoregressively adding generated novel views to the training set.

– We introduce a two-step approach for generating novel training views by performing *in-painting* and *artifact removal* with 2D diffusion models, which avoids fine-tuning on large-scale 3D data.
– We show that Sp^{2}360 outperforms recent works based on regularization and generative priors in reconstructing large 3D scenes.

2 Related Work

2.1 Sparse-View Radiance Fields

Following the breakthroughs of NeRF [37] and 3D Gaussian Splatting [29] for inverse rendering of radiance fields, there have been many approaches to weaken the requirement of dense scene captures to sparse input views only. These methods can be categorized into regularization techniques and generalizable reconstruction priors.

Regularization Techniques Fitting a 3D representation from sparse observations only is an ill-posed problem and very prone to local minima. In the case of radiance fields, this is typically visible as 'floaters' during rendering of novel views. A classical technique for training with limited data is regularization. Many existing methods leverage depth from Structure-from-Motion [14, 46], monocular estimation [11, 31, 69, 76], or RGB-D sensors [60]. DietNeRF [26] proposes a semantic consistency loss based on CLIP [43] features. FreeNeRF [70] regularizes frequency range of NeRF inputs by increasing the frequencies of positional-encoding features in a coarse-to-fine manner. Moving closer to generative priors, RegNeRF [41] and DiffusioNeRF [68] maximize likelihoods of rendered patches under a trained normalizing flow or diffusion model, respectively.

Generalizable Reconstruction In the case of very few or even a single view only, regularization techniques are usually not strong enough to account for the ambiguity in reconstruction. Therefore, another line of research focuses on training priors for novel view synthesis across many scenes. pixelNeRF [71] extracts pixel-aligned CNN features from input images at projected sample points during volume rendering as conditioning for a shared NeRF MLP. Similarly, many approaches [7, 20, 33, 59] define different NeRF conditionings on 2D or fused 3D features. Following the trend of leveraging explicit data structures for accelerating NeRFs, further priors have been learned on triplanes [25], voxel grids [18], and neural points [64]. Building upon the success of 3D Gaussian Splatting [29] and its broad applications to, e.g., surface [17, 24] or non-rigid reconstruction [12, 35, 66], recent methods like pixelSplat [6] and MVSplat [10] achieve state-of-the-art performance in stereo view interpolation while enabling real-time rendering. However, all of these works train regression models that infer blurry novel views in case of high uncertainty. Bridging the gap of generalizable and generative priors, GeNVS [5] and latentSplat [65] render view-conditioned feature fields followed by a 2D generative decoding to obtain a novel view.

2.2 Generative Priors

In case of ambiguous novel views, the expectation over all possible reconstructions might itself not be a reasonable prediction. Therefore, regression-based approaches fail. Generative methods, on the other hand, try to sample from this possibly multi-modal distribution.

Diffusion Models In recent years, diffusion models [15, 22] emerged as the state-of-the-art for image synthesis. They are characterized by a pre-defined forward noising process that gradually destroys data by adding random (typically Gaussian) noise. The objective is to learn a reverse denoising process with a neural network that, after training, can sample from the data distribution given pure noise. Important improvements include refined sampling procedures [28, 56] and the more efficient application in a spatially compressed latent space compared to the high-resolution pixel space [47]. Their stable optimization, in contrast to GANs, enabled today's text-to-image generators [40, 44, 47, 49] trained on billions of images [54].

2D Diffusion for 3D While diffusion models have been applied directly on 3D representations like triplanes [8, 55], voxel grids [38], or (neural) point clouds [36, 53, 73], 3D data is scarce. Given the success of large-scale diffusion models for image synthesis, there is a great research interest in leveraging them as priors for 3D reconstruction and generation. DreamFusion [42] and follow-ups [9, 13, 32, 57, 61] employ score distillation sampling (SDS) to iteratively maximize the likelihood of radiance field renderings under a conditional 2D diffusion prior. For sparse-view reconstruction, existing approaches incorporate view-conditioning via epipolar feature transform [75], cross-attention to encoded relative poses [34, 50], or pixelNeRF [71] feature renderings [67]. However, this fine-tuning is expensive and requires large-scale multi-view data, which we circumvent with Sp^{2}360.

3 Method

In this section, we describe our method in detail. The section begins with a general overview of Sp^{2}360 in Sec. 3.1, outlining the autoregressive algorithm for training view generation. In the following sections, the individual parts of the system are introduced: the in-painting module in Sec. 3.2, the artifact removal procedure in Sec. 3.3, and the sparse 3DGS baseline method in Sec. 3.5.

3.1 Sp^{2}360 for Sparse-View 3D Reconstruction

Given a sparse set of input images $\mathcal{I} = \{\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_M\}$ with camera poses $\{\pi_1, \pi_2, ..., \pi_M\}$, and a sparse point cloud $\mathbf{P} \in \mathbb{R}^{S\times 3}$, estimated by Structure-from-Motion (SfM) [51, 52], our goal is to obtain a 3D Gaussian representation of the scene, which enables the rendering of novel views from camera perspectives that are largely different from given views in $\mathcal{I}$. The tackled scenario of sparse-view inputs is extremely challenging, as the given optimization problem is heavily under-constrained and leads to severe artifacts if done naively.

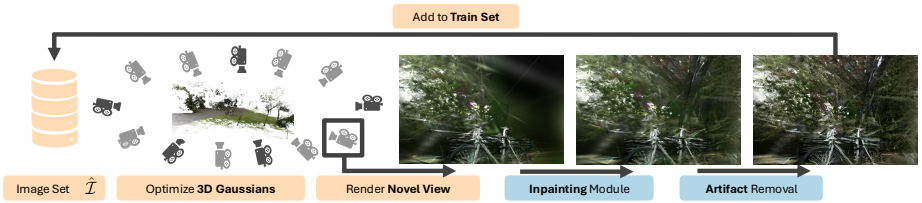

Fig. 1: Overview of Sp^{2}360. We render 3D Gaussians fitted to our sparse set of M views from a novel viewpoint. The image has missing regions and Gaussian artifacts, which are fixed by a combination of in-painting and denoising diffusion models. This then acts as pseudo ground truth to spawn and update 3D Gaussians and satisfy the new view constraints. This process is repeated for several novel views spanning the 360° scene until the representation becomes multi-view consistent.

An overview of Sp^{2}360 is given in Fig. 1 and Alg. 1. The approach begins by optimizing a set of 3D Gaussians [29] to reconstruct the initial sparse set of input images $\hat{\mathcal{I}} = \mathcal{I}$. For this, we introduce a *Sparse 3DGS* baseline (c.f. Sec. 3.5), which combines best practices from previous works on NeRFs. The obtained representation serves as initial prior for 360° reconstruction. Next, we autoregressively add new generated views to our training set: (1) we sample novel cameras and render novel views with artifacts and missing areas, make them look plausible by (2) performing in-painting (c.f. 3.2) and then (3) artifact removal (c.f. 3.3), before (4) adding them to set $\hat{\mathcal{I}}$ and continuing optimizing our 3D representation for k iterations.

Algorithm 1 Sp^{2}360 Algorithm

Require: Sparse input image set $\mathcal{I}$, camera poses $\{\pi_1, \pi_2, ..., \pi_M\}$, sparse point cloud $\mathbf{P} \in \mathbb{R}^{S\times 3}$

Ensure: Set of 3D Gaussians $\mathcal{G}$

$\hat{\mathcal{I}} \leftarrow \mathcal{I}$
$\mathcal{G} \leftarrow$ Optimize *Sparse 3DGS* for k iterations
for N iterations **do**
  $\pi \leftarrow$ Sample novel camera pose.
  $\mathbf{I} \leftarrow R_\pi(\mathcal{G})$ - Render from camera π
  $\mathbf{I} \leftarrow$ In-paint($\mathbf{I}$)
  $\mathbf{I} \leftarrow$ ArtifactRemoval($\mathbf{I}$)
  $\hat{\mathbf{I}} \leftarrow \hat{\mathbf{I}} \cup \{\mathbf{I}\}$
  $\mathcal{G} \leftarrow$ Optimize *Sparse 3DGS* for k iterations
end for

For training 3DGS in an iterative fashion, as outlined above, special precaution has to be taken to prevent overfitting on the initial views. An optimal schedule involves finding the number of iterations per cycle k and the hyperparameters for the original 3DGS [29]. We provide a detailed evaluation in **A.4**. In the following, we will detail the in-painting and artifact removal stages of our pipeline.

3.2 In-Painting Novel Views

When using only a sparse observation set, many areas remain unobserved, leading to areas of no Gaussians in the 3D representation and zero opacity regions in

some novel views (c.f. Fig.1). Regularization techniques cannot help with inferring details. Instead, we incorporate a generative in-painting diffusion model to fill such regions indicated by a binary mask ϕ, which is obtained by rendering opacity from our current Gaussian representation $\mathcal{G}$ and binarizing it with a threshold τ.

We fine-tune Stable Diffusion 2 [47] to perform in-painting on our novel view renderings. The current training images $\hat{\mathcal{I}}$ are used as training data to adapt it to the current scene. For the fine-tuning technique, we get inspired by recent work [58] that uses LoRA [23] adapters for the UNet $\boldsymbol{\epsilon_\theta}$ and text encoder $c_\theta(y)$. Given images $\hat{\mathcal{I}} = \{\mathbf{I}_1, ..., \mathbf{I}_M\}$, we create artificial masks $\{\phi_1, ..., \phi_M\}$ by creating random rectangular masks over each image and taking either their union or the complement of the union. Then, the adapter weights are fine-tuned using the following objective:

$$\mathcal{L} = \mathbb{E}_{i\sim\mathcal{U}(M),y,\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{1}),t}\Big[\|\epsilon_t - \boldsymbol{\epsilon_\theta}(\mathbf{z}_t; t, \phi_i, c_\theta(y))\|_2^2\Big], \tag{1}$$

where $\mathbf{z}_t$ is the diffused latent encoding of image $\mathbf{I}_i$ at step t. Here, y is a simple text prompt - "A photo of [V]", where [V] is a rare token like in DreamBooth [48] whose embedding is optimized for in-painting. For views $\mathbf{I} \notin \mathcal{I}_i$ (not from the original set), Eq 1 is only evaluated for regions that have a rendered opacity $> \tau$. This in-painting objective for fine-tuning enables our model to in-paint missing regions in a novel view rendering with details faithful to the observed M views. At inference, $\boldsymbol{\epsilon_\theta}$ predicts the noise in $\mathbf{z}_t$ as:

$$\hat{\epsilon_t} = \boldsymbol{\epsilon_\theta}(\mathbf{z}_t; t, \phi_i, c_\theta(y)), \tag{2}$$

which is used to progressively obtain less noisy latents in s DDIM [56] sampling steps starting from t. After passing the denoised latent $\hat{z}_0$ through a VAE decoder $\mathcal{D}$. we obtain the in-painted image x_0.

3.3 Removing Sparse-View Artifacts

The in-painting technique can fill in plausible details in low-opacity areas of novel view renders. However, it cannot deal with the dominance of blur, floaters, and color artifacts, which are not detected by the in-painting mask ϕ. As such, we resort to a diffusion-based approach to learn how typical artifacts in 3D Gaussian representations from sparse views look like and how to remove them.

We fine-tune an image-conditioned diffusion model [4] to edit images based on short, user-friendly edit instructions. Here, the UNet $\boldsymbol{\epsilon_\theta}$ is trained to predict noise in z_t conditioned on a given image c, in addition to the usual text conditioning $c_\theta(y)$. To enable this, additional input channels are added to the first convolutional layer of $\boldsymbol{\epsilon_\theta}$ so that z_t and $\mathcal{E}(c)$ can be concatenated. Weights of these channels are initialized to zero, whereas rest of the model is initialized from the pre-trained Stable Diffusion v1.5 checkpoint. For architectural design choices, we resort to those made by Instruct-Pix2Pix [4].

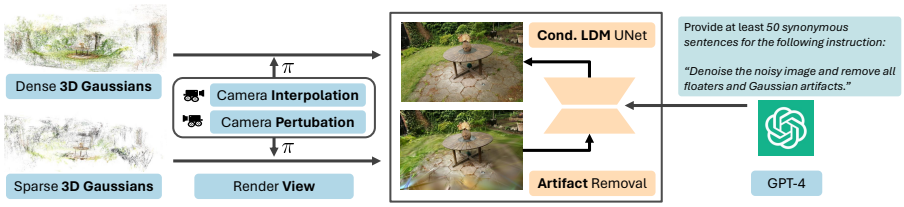

Fig. 2: Artifact removal fine-tuning. Pairs of clean images and images with artifacts are obtained from 3DGS fitted to sparse and dense observations, respectively, across 36 scenes. These are combined with one of 51 synonymous prompts generated by GPT-4 [1] from a base instruction. SD v1.5 [47] is then fine-tuned with a dataset of $10.5K$ samples for the Gaussian artifact removal task.

Dataset Creation For training, we rely on a set $\mathcal{X} = \{(\mathbf{x}^i, \mathbf{c}^i, y^i)_{i=1}^N\}$ of data triplets, each containing a clean image $\mathbf{x}^i$, an image with artifacts $\mathbf{c}^i$, and the corresponding edit instruction for fine-tuning y^i, to "teach" a diffusion model how to detect Gaussian artifacts and generate a clean version of the conditioning image. For this, we build an *artifact simulation engine* comprising a 3DGS model fitted to dense views, *Sparse 3DGS* fitted to few views, and camera interpolation and perturbation modules to use supervision of the dense model at viewpoints beyond ground truth camera poses. The fine-tuning setup is illustrated in Fig. 2. For a given scene, we fit sparse models for $M \in \{3, 6, 9, 18\}$ number of views. For larger M, we observe that there are very few artifacts beyond standard Gaussian blur. To have diversity in edit instructions, we start with a base instruction - "Denoise the noisy image and remove all floaters and Gaussian artifacts." and ask GPT-4 to generate 50 synonymous instructions. During training, each clean, artifact image pair is randomly combined with one of these 51 instructions.

Training the Artifact Removal Module Using our synthetically curated dataset $\mathcal{X}$, we fine-tune SD v1.5 as follows:

$$\mathcal{L} = \mathbb{E}_{i\sim\mathcal{U}(N),\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I}),t}\Big[\|\epsilon_t - \boldsymbol{\epsilon_\theta}(\mathbf{z}_t^i; t, \mathcal{E}(\mathbf{c}^i), c_\theta(y^i))\|_2^2\Big] \tag{3}$$

where $\mathbf{z}_t^i$ is the encoded image $\mathbf{x}^i$ diffused with sampled noise ϵ at time step t. Thus, the model is fine-tuned to generate clean images $\mathbf{x}^i$, conditioned on artifact images $\mathbf{c}^i$ and text prompts y^i.

Generating Clean Renders Given an in-painted rendering $\mathbf{x}_0$ and the base prompt $y =$"Denoise the noisy image and remove all floaters and Gaussian artifacts" at inference time, the fine-tuned artifact removal UNet $\boldsymbol{\epsilon_\theta}$ predicts the noise in latent $\mathbf{z}_t$ according to $t \sim \mathcal{U}[t_{min}, t_{max}]$ as:

$$\begin{aligned}\hat{\epsilon}_t =& \boldsymbol{\epsilon_\theta}(\mathbf{z}_t; t, \varnothing, \varnothing) \\ &+ s_I \cdot (\boldsymbol{\epsilon_\theta}(\mathbf{z}_t; t, \mathcal{E}(\mathbf{x}_0), \varnothing) - \boldsymbol{\epsilon_\theta}(\mathbf{z}_t; t, \varnothing, \varnothing)) \\ &+ s_T \cdot (\boldsymbol{\epsilon_\theta}(\mathbf{z}_t; t, \varnothing, c_\theta(y)) - \boldsymbol{\epsilon_\theta}(\mathbf{z}_t; t, \mathcal{E}(\mathbf{x}_0), \varnothing))\end{aligned} \tag{4}$$

where s_I and s_T are the image and prompt guidance scales, dictating how strongly the final multistep reconstruction agrees with the in-painted render $\mathbf{x}_0$ and the edit prompt y, respectively. After s DDIM [56] sampling steps, we obtain our final image by decoding the denoised latent.

3.4 Distilling 2D priors to 3D

Our diffusion priors infer plausible detail in unobserved regions and an iterative update algorithm (**A.4**) incrementally grows and updates scene Gaussians by fusing information at novel viewpoints. We initialize the scene with 3D Gaussians fitted by *Sparse 3DGS* to M input views, autoregressively sample closest novel viewpoints (based on SE3 distance), obtain pseudo ground truths for novel views using our combination of diffusion priors, add them to the training stack and optimize for certain iterations (determined by the schedule in **A.4**). At every iteration, we sample either an observed or unobserved viewpoint from the current training stack. We optimize Gaussian attributes using the 3DGS objective for known poses and the SparseFusion [75] objective for novel views:

$$\mathcal{L}_{sample}(\psi) = \mathbb{E}_{\pi,t}\Big[w(t)(\|I_\pi - \hat{I}_\pi\|_1 + \mathcal{L}_p(I_\pi, \hat{I}_\pi))\Big] \tag{5}$$

where $\mathcal{L}_p$ is the perceptual loss [72], $w(t)$ a noise-dependent weighting function, I_π is the 3DGS render at novel viewpoint π, and $\hat{I}_\pi$ is the inpainted, clean version of I_π obtained with our cascaded diffusion priors.

3.5 Sparse 3DGS

The sparse 3DGS baseline serves as our starting point that already improves reconstruction quality over standard 3DGS. It is inspired by several recent works on sparse neural fields [14, 41] and 3DGS [31, 69, 76]. The baseline unifies depth regularization from monocular depth estimators and depth priors from pseudo views while using specific hyperparameter settings, such as densification thresholds and opacity reset configurations for Gaussian splatting. It is outlined in detail in **A.2** and evaluated individually in the ablation studies in Sec. 4.4.

4 Experiments

This section compares *Sp*^{2}*360* with state-of-the-art sparse-view reconstruction techniques. We also provide detailed ablation studies motivating specific design choices across different components of our approach. The setup for *Iterative 3DGS* (c.f. **A.4**) translates seamlessly to the iterative distillation procedure (c.f. 3.4) with diffusion priors, and hence, we refrain from further analysis here.

4.1 Experimental Setup

Evaluation Dataset We evaluate *Sp*^{2}*360* on the 9 scenes of the MipNeRF360 dataset [2], comprising 5 outdoor and 4 indoor scenes. Each scene has a central

object or area of complex geometry with an equally intricate background. This makes it the most challenging 360° dataset compared to CO3D [45], RealEstate 10K [74], DTU [27], etc. We retain the train/test split of the MipNeRF360 dataset, where every $8th$ image is kept aside for evaluation. To create M-view subsets, we sample from the train set of each scene using a geodesic distance-based heuristic to encourage maximum possible scene coverage (see supplement for details).

Fine-tuning dataset We fine-tune the in-painting module only on the M input views. For the artifact removal module, we train 3DGS on sparse and dense subsets of 360° scenes from MipNeRF360 [2], Tanks and Temples [30] and Deep Blending [19] across a total of 36 scenes to obtain $\sim 10.5K$ data triplets. We train 9 separate artifact removal modules holding out the MipNeRF360 scene we want to reconstruct. On a single A100 GPU, fine-tuning the in-painting and artifact removal modules takes roughly 2h and 1h, respectively.

Baselines We compare our approach against 7 baselines. FreeNeRF [70], RegNeRF [41], DiffusioNeRF [68], and DNGaussian [31] are few-view regularization methods based on NeRFs or 3D Gaussians. ZeroNVS [50] is a recent generative approach for reconstructing a complete 3D scene from a single image. We use the ZeroNVS* baseline introduced in ReconFusion [67], designed to adapt ZeroNVS to multi-view inputs. Conditioning the diffusion model on the input view closest to the sampled random view enables scene reconstruction for a general M-view setting. We also compare against 3DGS, the reconstruction pipeline for Sp^{2}360, and against Sparse 3DGS, our self-created baseline.

Metrics Due to the generative nature of our approach, we employ FID [21] and KID [3] to measure similarity of distribution of reconstructed novel views and ground truth images. We also compute two perceptual metrics - LPIPS [72] and DISTS [16] - to measure similarity in image structure and texture in the feature space. Despite their known drawbacks as evaluators of generative techniques [5,50], we additionally provide PSNR and SSIM scores for completeness. Both favor pixel-aligned blurry estimates over high-frequency details, making them ill-suited to our setting.

4.2 Implementation Details

We implement our entire framework in Pytorch 1.12.1 and run all experiments on single $A100$ or $A40$ GPUs. We work with image resolutions in the 400-600 pixel range as this is closest to the output resolution of 512 for both diffusion models. We set $\lambda_1 = 0.2$ (same as 3DGS) and vary $\lambda_{depth}, \lambda_{pseudo}$ in $\{0.0, 0.05, 0.1\}$ for *Sparse 3DGS*. For in-painting, we set $\tau = 0.8$ and fine-tune the in-painting module at 512×512 resolution for 3000-5000 steps with LoRA modules of rank in $\{8, 16, 32\}$ (depending on M). We use a batch size of 16 and learning rates of 2e-4 for the UNet and 4e-5 for the text encoder. The artifact removal module is trained at 256×256 resolution for 2500 iterations with batch size 16 and learning rate 1e-4. To enable classifier-free guidance for both models, we randomly dropout conditioning inputs (text, mask, image, etc.) with probability of 0.1 during

training. The classifier-free guidance scales are set to $s_I = 2.5$ and $s_T = 7.0$. We use $t_{max} = 0.99$ for both in-painting and artifact removal and linearly decrease t_{min} for the in-painting module from 0.98 to 0.90, and from 0.98 to 0.70 for the artifact removal module. We sample both the in-painted and clean renders for $s = 20$ DDIM sampling steps. We also linearly decay the weight of $\mathcal{L}_{sample}$ (c.f. 3.4) from 1 to 0.1 over $30k$ iterations.

4.3 Comparison Results

We evaluate all baselines on our proposed splits for each scene. We report averaged quantitative results in Tables 1 and 2 and compare novel view rendering quality in Fig 3. We outperform all baselines for 9-view reconstruction across all metrics and are second only to DiffusioNeRF on FID, KID, and DISTS for 3 and 6-view reconstruction. Unlike the baselines, our approach consistently improves with increasing M across all metrics.

Fig. 3: Qualitative comparison of *Sp*^{2}*360* with few-view methods. Our approach consistently fairs better in recovering image structure from foggy geometry, where baselines typically struggle with "floaters" and color artifacts. We encourage the reader to refer to our **supplemental 360° video**, where the benefits of our method can be observed along a smooth trajectory.

Table 1: Quantitative comparison with state-of-the-art sparse-view reconstruction techniques on classical metrics. Despite being a generative solution, we outperform all baselines across all view splits on both pixel-aligned and perceptual metrics.

	PSNR ↑			SSIM ↑			LPIPS ↓		
Method	3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view
3DGS	10.288	11.628	12.658	0.102	0.141	0.190	0.709	0.661	0.605
FreeNeRF	9.948	9.599	10.641	0.124	0.129	0.145	0.682	0.679	0.668
RegNeRF	11.030	10.764	11.020	0.117	0.134	0.145	0.663	0.660	0.661
DNGaussian	9.867	10.671	11.275	0.124	0.137	0.179	0.754	0.730	0.729
DiffusioNeRF	10.749	11.728	11.430	0.093	0.116	0.112	0.709	0.678	0.654
ZeroNVS*	10.406	9.990	9.719	0.079	0.079	0.082	0.709	0.711	0.700
Ours	12.927	13.701	14.121	0.211	0.231	0.261	0.647	0.622	0.591

Table 2: Quantitative comparison with few-view reconstruction techniques on metrics suited for generative reconstruction. We are second only to DiffusioNeRF for 3 and 6-view reconstruction but achieve higher scores for 9 views.

	FID ↓			KID ↓			DISTS ↓		
Method	3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view
3DGS	392.620	343.336	292.324	0.313	0.268	0.229	0.476	0.429	0.321
FreeNeRF	347.625	343.833	342.917	0.254	0.249	0.258	0.392	0.388	0.388
RegNeRF	362.856	347.045	349.043	0.291	0.266	0.247	0.399	0.403	0.404
DNGaussian	431.687	420.110	414.307	0.311	0.285	0.281	0.581	0.571	0.544
DiffusioNeRF	273.096	225.661	290.184	0.158	0.104	0.183	0.362	0.319	0.378
ZeroNVS*	351.090	335.155	337.457	0.283	0.282	0.290	0.437	0.429	0.428
Ours	318.470	283.504	230.565	0.273	0.229	0.162	0.384	0.357	0.315

4.4 Ablation Studies

Fig. 4: Ablation Study on 9-view reconstruction of *garden* scene. Our fine-tuned artifact removal module and iterative schedule contribute the most toward quality of the final reconstruction. 3D Gaussians from *Sparse 3DGS* act as suitable geometric prior in the absence of explicit view conditioning.

In Tab 3 and Fig 4, we ablate the relative importance of each component towards 360° reconstruction. We pick the garden scene and its 9 view split for this study. We first show the benefits of the regularization heuristics in *Sparse 3DGS* over native 3DGS. We attempt reconstruction using either the in-painting or artifact removal module and show that their combination works best for both novel view synthesis and final reconstruction. Interestingly, leaving out the artifact removal module results in the best PSNR and SSIM scores across all variants, emphasizing that classical metrics reward blurry reconstructions, whereas FID and KID favor sharp, realistic details in novel views. We remove the iterative update formulation and add in-painted clean renders for all novel views simultaneously. This variant performs worse than *Sparse 3DGS*, highlighting the need for autoregressive scene generation in our setting. Additionally, we try to supervise the renderings at novel views using the original 3DGS objective. However, this performs slightly worse than Eq 5. Note that an SDS-based formulation [50] is not applicable here due to the two-step nature of generative view synthesis.

4.5 Scaling to More Views

In Fig 5, we analyze the relevance of diffusion priors with increasing M. We evaluate Sp^{2}360 and 3DGS on view splits of increasing size with $M \in \{3, 6, 9, 18, 27, 54, 81\}$. For $M \leq 27$, our method consistently improves 3DGS' generalization at novel views. We observe that as ambiguities resolve with increasing scene coverage, diffusion-based regularization becomes less important, and for $M \geq 54$, our method performs either equally or slightly worse across the 6 performance measures.

4.6 Efficiency

Our approach focuses on limited use of multi-view data, minimal training times, and fast inference. We fit 3D Gaussians to our initial M inputs inside 30 mins

Table 3: Ablation study on the 9 view split of *garden* scene. Our combination of diffusion priors complements each other effectively. Without an iterative schedule to fuse novel views, our method fairs worse than a regularized baseline. Using the 3DGS objective for novel view renders leads to slightly worse performance on FID and KID.

Method	FID ↓	KID ↓	DISTS ↓	LPIPS ↓	PSNR ↑	SSIM ↑
3DGS	223.594	0.146	0.247	0.502	14.470	0.287
Sparse 3DGS	200.703	0.127	0.247	0.522	16.589	0.367
Sp^{2}360 w/o Artifact R.	209.477	0.120	0.253	0.528	16.850	0.380
Sp^{2}360 w/o In-painting	151.098	0.071	0.230	0.524	15.121	0.299
Sp^{2}360 w/o Schedule	214.674	0.111	0.277	0.576	14.132	0.292
Sp^{2}360 w/ $\mathcal{L}_{D-SSIM}$	133.875	0.051	0.225	0.502	15.677	0.326
Sp^{2}360	124.768	0.048	0.224	0.504	15.759	0.326

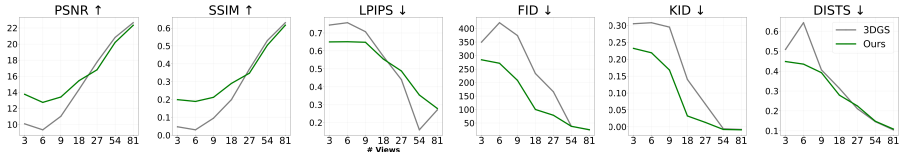

Fig. 5: Scalability of Sp^{2}360 with input views. Our combination of fine-tuned diffusion priors improves performance of 3DGS up to 27 input views of the *bicycle* scene, alleviating the need for dense captures.

and fine-tune the in-painting module in $\sim$ 2 hrs. Sp^{2}360 trains inside 30 mins on a single A100 GPU, significantly faster than all baselines with NeRF backbones. FreeNeRF takes $\sim$ 1 day while RegNeRF takes $>$ 2 days to train on a single A40 GPU. Both ZeroNVS* and DiffusioNeRF require $\sim$ 3 hrs to distill 2D diffusion priors into Instant-NGP [39], whereas ReconFusion trains in 1 hour on 8 A100 GPUs [1]. All 3 approaches use custom diffusion models that take several days to train on large-scale 3D datasets. On the contrary, we only need *10.5K* samples to obtain a generalized artifact removal module that efficiently eliminates Gaussian artifacts and recovers image details in the foreground and background. We currently use a leave-one-out mechanism for the 9 scenes in MipNeRF360 to train the artifact removal module for a particular MipNeRF360 scene. However, we only do this to have enough diverse data pairs across the 3 datasets. This requirement can be easily alleviated using multi-view training data from other sources. Given the availability of a generalized artifact removal module, our entire pipeline can reconstruct a 360 scene in under 3.5 hrs on a single A100 GPU. Courtesy of our 3D representation, we retain the real-time rendering capabilities of 3DGS.

[1] Source: ReconFusion authors

5 Limitations & Future Work

Despite being a low-cost and efficient approach for reconstructing complex 360° scenes from a few views, Sp^{2}360 has certain limitations. Our approach is limited by the sparse geometry prior from an SfM point cloud, estimated from few views. For example, the point cloud for 9 views of the *bicycle* scene only has 628 points. This prevents our method from achieving even higher fidelity and restricting artifacts in distant, ambiguous novel views. Owing to the SfM geometry initialization, our method also cannot reconstruct an entire scene with plausible details from 3 and 6 views and is only limited to novel view synthesis for viewpoints close to the train viewpoints. DUSt3R [63] is a recent stereo reconstruction pipeline that can potentially provide a stronger geometry prior. We also plan to evaluate Sp^{2}360 on the MipNeRF360 splits released by ReconFusion and CAT3D, which provide more scene coverage than our proposed splits, facilitating better extrapolation by diffusion priors at distant novel views. However, due to lower overlap across views, COLMAP is not able to register all training images to a unified point cloud, preventing an appropriate initialization for Sp^{2}360. Integrating DUSt3R would alleviate this issue as it does not rely on high overlap across image pairs for 3D reconstruction. We plan to integrate DUSt3R for pose-free sparse-view reconstruction aided by our diffusion priors for future work.

6 Conclusion

We present Sp^{2}360, a low-cost, data-efficient approach to reconstruct complex 360° scenes from a few input images. We proposed a system that combines diffusion priors specialized for in-painting and Gaussian artifact removal to generate artificial novel views, which are iteratively added to our training image set. In our experiments we show that our approach approves over previous methods on the challenging MipNeRF360 dataset and illustrate the contributions of individual components in our ablation studies. For future work, we see the potential of our system to be improved with additional geometry cues from 3D vision foundation models.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. In: arXiv (2023) 7
2. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: CVPR (2022) 8, 9
3. Bińkowski, M., Sutherland, D.J., Arbel, M., Gretton, A.: Demystifying MMD GANs. In: ICLR (2018) 9
4. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: CVPR (2023) 6

5. Chan, E.R., Nagano, K., Chan, M.A., Bergman, A.W., Park, J.J., Levy, A., Aittala, M., Mello, S.D., Karras, T., Wetzstein, G.: GeNVS: Generative novel view synthesis with 3D-aware diffusion models. In: ICCV (2023) 3, 9
6. Charatan, D., Li, S., Tagliasacchi, A., Sitzmann, V.: pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In: CVPR (2024) 3
7. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In: ICCV (2021) 3
8. Chen, H., Gu, J., Chen, A., Tian, W., Tu, Z., Liu, L., Su, H.: Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. In: ICCV (2023) 4
9. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In: ICCV (2023) 4
10. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images (2024) 3
11. Chung, J., Oh, J., Lee, K.M.: Depth-regularized optimization for 3d gaussian splatting in few-shot images. In: arXiv (2023) 2, 3
12. Das, D., Wewer, C., Yunus, R., Ilg, E., Lenssen, J.E.: Neural parametric gaussians for monocular non-rigid object reconstruction. In: CVPR (2024) 3
13. Deng, C., Jiang, C., Qi, C.R., Yan, X., Zhou, Y., Guibas, L., Anguelov, D., et al.: Nerdi: Single-view nerf synthesis with language-guided diffusion as general image priors. In: CVPR (2023) 4
14. Deng, K., Liu, A., Zhu, J.Y., Ramanan, D.: Depth-supervised NeRF: Fewer views and faster training for free. In: CVPR (2022) 2, 3, 8
15. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. In: NeurIPS (2021) 4
16. Ding, K., Ma, K., Wang, S., Simoncelli, E.P.: Image quality assessment: Unifying structure and texture similarity. In: IEEE TPAMI (2020) 9
17. Guédon, A., Lepetit, V.: Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In: CVPR (2024) 3
18. Guo, P., Bautista, M.A., Colburn, A., Yang, L., Ulbricht, D., Susskind, J.M., Shan, Q.: Fast and explicit neural view synthesis. In: WACV (2022) 3
19. Hedman, P., Philip, J., Price, T., Frahm, J.M., Drettakis, G., Brostow, G.: Deep blending for free-viewpoint image-based rendering. In: ACM TOG (2018) 9
20. Henzler, P., Reizenstein, J., Labatut, P., Shapovalov, R., Ritschel, T., Vedaldi, A., Novotny, D.: Unsupervised learning of 3d object categories from videos in the wild. In: CVPR (2021) 3
21. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: NeurIPS (2017) 9
22. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS (2020) 4
23. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-rank adaptation of large language models. In: ICLR (2022) 6
24. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2d gaussian splatting for geometrically accurate radiance fields. In: SIGGRAPH (2024) 3
25. Irshad, M.Z., Zakharov, S., Liu, K., Guizilini, V., Kollar, T., Gaidon, A., Kira, Z., Ambrus, R.: Neo 360: Neural fields for sparse view synthesis of outdoor scenes. In: ICCV (2023) 3
26. Jain, A., Tancik, M., Abbeel, P.: Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: ICCV (2021) 2, 3

27. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: CVPR (2014) 9
28. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models. In: NeurIPS (2022) 4
29. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. In: ACM TOG (2023) 1, 3, 5
30. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: benchmarking large-scale scene reconstruction. In: ACM TOG (2017) 9
31. Li, J., Zhang, J., Bai, X., Zheng, J., Ning, X., Zhou, J., Gu, L.: Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In: CVPR (2024) 2, 3, 8, 9
32. Lin, C.H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.Y., Lin, T.Y.: Magic3d: High-resolution text-to-3d content creation. In: CVPR (2023) 4
33. Lin, K.E., Yen-Chen, L., Lai, W.S., Lin, T.Y., Shih, Y.C., Ramamoorthi, R.: Vision transformer for nerf-based view synthesis from a single input image. In: WACV (2023) 3
34. Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object. In: ICCV (2023) 2, 4
35. Luiten, J., Kopanas, G., Leibe, B., Ramanan, D.: Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In: 3DV (2024) 3
36. Melas-Kyriazi, L., Rupprecht, C., Vedaldi, A.: Pc2: Projection-conditioned point cloud diffusion for single-image 3d reconstruction. In: CVPR (2023) 4
37. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020) 1, 3
38. Müller, N., Siddiqui, Y., Porzi, L., Bulo, S.R., Kontschieder, P., Nießner, M.: Diffrf: Rendering-guided 3d radiance field diffusion. In: CVPR (2023) 4
39. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. In: ACM TOG (2022) 13
40. Nichol, A.Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., Mcgrew, B., Sutskever, I., Chen, M.: GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In: ICML (2022) 2, 4
41. Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S.M., Geiger, A., Radwan, N.: Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In: CVPR (2022) 2, 3, 8, 9
42. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. In: ICLR (2023) 4
43. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021) 3
44. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. In: arXiv (2022) 2, 4
45. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: ICCV (2021) 9
46. Roessle, B., Barron, J.T., Mildenhall, B., Srinivasan, P.P., Nießner, M.: Dense depth priors for neural radiance fields from sparse input views. In: CVPR (2022) 2, 3
47. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022) 2, 4, 6, 7

48. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In: CVPR (2023) 6
49. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., Ho, J., Fleet, D.J., Norouzi, M.: Photorealistic text-to-image diffusion models with deep language understanding. In: NeurIPS (2022) 2, 4
50. Sargent, K., Li, Z., Shah, T., Herrmann, C., Yu, H.X., Zhang, Y., Chan, E.R., Lagun, D., Fei-Fei, L., Sun, D., Wu, J.: ZeroNVS: Zero-shot 360-degree view synthesis from a single real image. In: CVPR (2024) 2, 4, 9, 12
51. Schönberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: CVPR (2016) 4
52. Schönberger, J.L., Zheng, E., Pollefeys, M., Frahm, J.M.: Pixelwise view selection for unstructured multi-view stereo. In: ECCV (2016) 4
53. Schröppel, P., Wewer, C., Lenssen, J.E., Ilg, E., Brox, T.: Neural point cloud diffusion for disentangled 3d shape and appearance generation. In: CVPR (2024) 4
54. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., Schramowski, P., Kundurthy, S., Crowson, K., Schmidt, L., Kaczmarczyk, R., Jitsev, J.: Laion-5b: An open large-scale dataset for training next generation image-text models. In: NeurIPS (2022) 4
55. Shue, J.R., Chan, E.R., Po, R., Ankner, Z., Wu, J., Wetzstein, G.: 3d neural field generation using triplane diffusion. In: CVPR (2023) 4
56. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: ICLR (2021) 4, 6, 8
57. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In: ICLR (2024) 4
58. Tang, L., Ruiz, N., Qinghao, C., Li, Y., Holynski, A., Jacobs, D.E., Hariharan, B., Pritch, Y., Wadhwa, N., Aberman, K., Rubinstein, M.: Realfill: Reference-driven generation for authentic image completion. In: arXiv (2023) 6
59. Trevithick, A., Yang, B.: Grf: Learning a general radiance field for 3d scene representation and rendering. In: ICCV (2021) 3
60. Wang, G., Chen, Z., Loy, C.C., Liu, Z.: Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In: ICCV (2023) 2, 3
61. Wang, H., Du, X., Li, J., Yeh, R.A., Shakhnarovich, G.: Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In: CVPR (2023) 4
62. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In: NeurIPS (2021) 1
63. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: CVPR (2024) 14
64. Wewer, C., Ilg, E., Schiele, B., Lenssen, J.E.: SimNP: Learning self-similarity priors between neural points. In: ICCV (2023) 3
65. Wewer, C., Raj, K., Ilg, E., Schiele, B., Lenssen, J.E.: latentsplat: Autoencoding variational gaussians for fast generalizable 3d reconstruction. In: arXiv (2024) 3
66. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Xinggang, W.: 4d gaussian splatting for real-time dynamic scene rendering (2024) 3
67. Wu, R., Mildenhall, B., Henzler, P., Park, K., Gao, R., Watson, D., Srinivasan, P.P., Verbin, D., Barron, J.T., Poole, B., Holynski, A.: Reconfusion: 3d reconstruction with diffusion priors. In: CVPR (2024) 2, 4, 9

68. Wynn, J., Turmukhambetov, D.: DiffusioNeRF: Regularizing Neural Radiance Fields with Denoising Diffusion Models. In: CVPR (2023) 2, 3, 9
69. Xiong, H., Muttukuru, S., Upadhyay, R., Chari, P., Kadambi, A.: Sparsegs: Real-time 360° sparse view synthesis using gaussian splatting. In: arXiv (2023) 2, 3, 8
70. Yang, J., Pavone, M., Wang, Y.: Freenerf: Improving few-shot neural rendering with free frequency regularization. In: CVPR (2023) 3, 9
71. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelNeRF: Neural radiance fields from one or few images. In: CVPR (2021) 3, 4
72. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018) 8, 9
73. Zhou, L., Du, Y., Wu, J.: 3d shape generation and completion through point-voxel diffusion. In: ICCV (2021) 4
74. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: Learning view synthesis using multiplane images. In: SIGGRAPH (2018) 9
75. Zhou, Z., Tulsiani, S.: Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In: CVPR (2023) 4, 8
76. Zhu, Z., Fan, Z., Jiang, Y., Wang, Z.: Fsgs: Real-time few-shot view synthesis using gaussian splatting. In: arXiv (2023) 2, 3, 8