GaMO: Geometry-aware Multi-view Diffusion Outpainting for Sparse-View 3D Reconstruction

Yi-Chuan Huang, Hao-Jen Chien, Chin-Yang Lin, Chih-Yu Chang, Ying-Huan Chen, and Yu-Lun Liu

National Yang Ming Chiao Tung University
yichuanh.cs12@nycu.edu.tw, yulunliu@cs.nycu.edu.tw

arXiv:2512.25073v2 [cs.CV] 4 Apr 2026

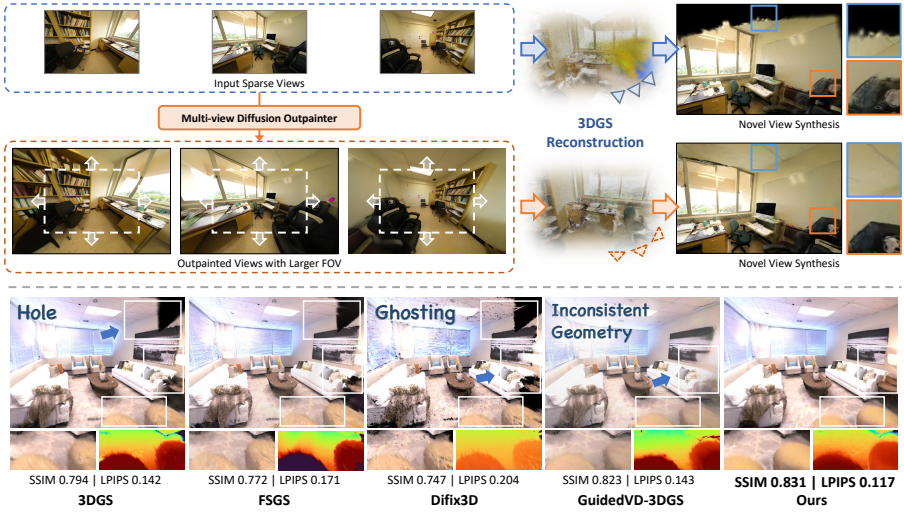


Fig. 1: Overview and comparison. *(Top)* Our method, **GaMO** (**G**eometry-**a**ware **M**ulti-view Diffusion **O**utpainter), expands sparse input views into wide-FOV outpainted views via a multi-view diffusion model, which are then used to refine 3D Gaussian Splatting (3DGS) [32] reconstruction, producing high-fidelity novel views with improved geometric consistency and visual clarity. *(Bottom)* Qualitative comparison with existing methods, including 3DGS [32], FSGS [152], Difix3D [106], and GuidedVD-3DGS [149]. Previous approaches suffer from *holes*, *ghosting*, or *inconsistent geometry* when trained with sparse inputs. In contrast, our method effectively mitigates these artifacts and achieves superior image quality.

Abstract. Recent 3D reconstruction methods achieve impressive results with dense multi-view imagery but struggle when only a few views are available. Various approaches, including regularization techniques, semantic priors, and geometric constraints, have been implemented to address this challenge. Recent diffusion-based approaches further improve performance by generating novel views to augment training data. Despite this progress, we identify three critical limitations in current state-of-the-art approaches: (i) inadequate coverage beyond known view

peripheries, (ii) geometric inconsistencies across generated views, and (iii) computational inefficiency due to expensive pipelines. We introduce **GaMO** (**G**eometry-**a**ware **M**ulti-view **O**utpainter), a framework that reformulates sparse-view reconstruction through multi-view outpainting. Instead of generating new viewpoints, GaMO expands the field of view from existing camera poses, which inherently preserves geometric consistency while providing broader scene coverage. Our approach employs multi-view conditioning and geometry-aware denoising strategies in a zero-shot manner without training. Extensive experiments on Replica, ScanNet++, and Mip-NeRF 360 demonstrate strong reconstruction performance across sparse-view settings (3, 6, and 9 input views). Notably, our method is significantly more efficient than existing diffusion-based approaches, reducing the overall runtime to within 10 minutes. Project page: `https://yichuanh.github.io/GaMO/`

Keywords: Sparse-view 3D Reconstruction · Multi-View Outpainting · Geometry-aware Diffusion

1 Introduction

Reconstructing complete 3D scenes from limited input views is a fundamental problem with numerous tangible applications, ranging from virtual property tours to immersive telepresence. However, it remains notoriously difficult, often resulting in broken geometry and visible visual artifacts. Previous approaches attempted to address the sparsity of input views through regularization, semantic priors, or geometric constraints [26,40,68,92,126,152]. These methods remain limited in handling unobserved regions.

Recently, diffusion-based approaches [1, 106, 108, 110, 149] have generated novel views to improve the reconstruction quality for sparse observations. Nevertheless, these methods show three fundamental limitations: (1) novel view generation mainly focuses on enhancing angular coverage of existing geometry and often overlooks the extension beyond the periphery, leaving persistent **holes** and **ghostings** in the reconstruction; (2) geometric and photometric **inconsistencies** across novel and input views inevitably become prominent as view overlap increases due to internal diffusion variations; (3) novel view generation requires elaborate trajectory planning and camera pose sampling, making the process **time-consuming**.

Recent multi-view diffusion models incorporate geometry-aware priors and achieve strong geometric consistency across generated views. However, we find that directly using them to expand camera viewpoints is not suitable under extremely sparse-view settings. Fig. 2 shows an experiment where three input views are used to generate interpolated novel views via diffusion, which are then used to train 3DGS [32]. Increasing the number of generated views (from 3 to 13 total views) degrades reconstruction quality, leading to lower SSIM and higher LPIPS. This observation suggests that generating additional viewpoints is not always beneficial under sparse-view settings.



Fig. 2: Motivation: Outpainting vs. diffusion-generated novel views. Using multi-view diffusion [4], we train 3DGS [32] with three settings: interpolated views (green), a 3-view baseline (blue), and outpainting (orange). **Top:** Adding more diffusion-generated views (3–13) may introduce geometric inconsistencies under sparse-view conditions. **Bottom:** SSIM/LPIPS show a similar trend: increasing generated views can degrade reconstruction quality, while outpainting remains more stable.

Instead, we observe that *outpainting*, rather than novel view generation, better leverages the geometry-aware priors of multi-view diffusion models and provides a more suitable paradigm for sparse-view 3D reconstruction. By extending content around existing input views, outpainting expands spatial coverage while preserving geometric consistency. Based on this insight, we propose **GaMO** (**G**eometry-**a**ware **M**ultiview **O**utpainting), which (1) expands the field of view (FOV) to cover unobserved regions while preserving geometric consistency, (2) avoids fusing multiple hallucinated views in 3D space, reducing misalignment artifacts, and (3) reconstructs scenes efficiently through a single outpainting pass, achieving tens-of-times speedup over video diffusion-based methods. We evaluate our approach on Replica [80], ScanNet++ [131], and Mip-NeRF 360 [3], demonstrating competitive performance with improved perceptual quality.
We summarize our contributions as follows:

- We establish outpainting as a superior paradigm for sparse-view reconstruction, eliminating common issues including holes, ghosting artifacts, and geometric inconsistencies.
- We develop a geometry-aware outpainting approach with novel conditioning and denoising strategies in zero-shot manner without finetuning.
- We achieve competitive performance across the Replica, ScanNet++, and Mip-NeRF 360 datasets. Our method provides strong geometric accuracy and perceptual quality across 3, 6, and 9 input views, while significantly reducing reconstruction time to under 10 minutes.

2 Related Work

Sparse-view 3D Gaussian Splatting. While 3DGS [32] achieves remarkable quality with dense inputs, sparse-view reconstruction remains challenging, particularly for indoor scenes [41]. Recent methods employ depth regularization [12,40], proximity-guided unpooling [152], dual-field co-regularization [144], structural

regularization via random Gaussian dropping [70], matching-prior-based structure consistency [72], and robust handling of unposed inputs [45]. Diffusion-augmented approaches use generative models to synthesize pseudo-views for enhanced training supervision [33,125]. Feed-forward approaches leverage cost volumes [6, 8], Gaussian bundle adjustment [15], depth-integrated splatting [119], multi-view stereo features [53], or transformer architectures [64, 82, 84, 99, 121]. Methods combining depth priors include DN-Splatter [89] with depth and normal cues, SplatFields [63] regularizing spatial autocorrelation, and large model priors [22,78,97,134]. While these methods *regularize* 3D representations, our work *augments* training data through geometry-aware outpainting for more complete scene coverage.

Multi-view diffusion models for 3D. Multi-view diffusion enables consistent 3D generation through multi-view attention [77, 95], synchronized volume attention [56], orthogonal view generation [76], and cross-domain diffusion [57]. Recent advances enforce consistency via 3D feature unprojection [127], epipolar attention [24, 34], depth-guided attention [23], differentiable rasterization [58], and pose-free dense generation [85]. Large multi-view Gaussian models enable high-resolution content creation [83]. Video diffusion models provide temporal consistency for multi-view synthesis [9, 10, 18, 36, 90, 137, 142], with view-integrated attention [116] and multi-view video generation [39] further improving spatiotemporal coherence. Additional methods include mesh generation [120], epipolar constraints [42], combined 2D-3D priors [51], correspondence-aware attention [86], and 3D feature fields [5]. These methods generate *novel views* from different poses. Our work performs *multi-view outpainting* to expand field-of-view of existing views, maintaining stronger geometric consistency for sparse-view scene reconstruction.

Diffusion priors for 3D reconstruction. Diffusion models provide learned priors through Score Distillation Sampling [44, 73, 101]. Improvements address over-smoothing [60], mode collapse [94], provide unified frameworks [61], classifier-based distillation [138], ODE-based sampling [38, 112], and optimize both 3D models and priors [11,128]. Video diffusion serves as powerful priors [48,62,132]. Reconstruction methods use multi-view conditioning [108], pseudo-observation enhancement [55], scene-grounding guidance [149], iterative refinement [54, 106, 110], inline prior-guided score matching under sparse views [97], visibility-guided decompositional reconstruction [67], and various coupling strategies [25, 52, 66, 118,124]. Native 3D diffusion includes latent approaches [21,37,107,109,129] and RL finetuning [113]. While these methods generate additional views or provide guidance, they face multi-view *inconsistency*. Our insight: diffusion models suit *outpainting known views* better than hallucinating novel perspectives, maintaining stronger geometric grounding.

Geometry-aware generation. Geometric consistency leverages Plücker coordinates for camera conditioning [2,20,27,30,117,123,143] and epipolar constraints or voxel representations [88] for multi-view consistency [24, 35, 66, 100, 115, 130,

148]. Camera motion control in video diffusion extends these representations to temporal settings [103]. Joint pixel-level image and depth synthesis further improves geometric fidelity [19]. Depth and normal conditioning proves critical [13, 17, 23, 31, 57, 58, 71, 74, 93]. Recent panoramic generation [96, 146] and video outpainting [139] typically operate in 2D or single-view scenarios. Our approach uniquely combines *multi-view outpainting* with *geometry awareness* through coarse 3DGS rendering, opacity-based masking, and noise resampling for consistent, geometrically plausible FOV expansion.

Outpainting and FOV expansion. Diffusion-based outpainting includes panoramic methods [16, 29, 47, 59, 75, 111, 133, 140, 141, 146] and restoration tasks [49, 87]. Video outpainting methods leverage input-specific adaptation [91] and temporal diffusion [147]. For 3D scenarios, methods employ visibility-aware inpainting [50, 105], generative scene completion via grid priors [104] or interactive extrapolation [135], reference-adapted diffusion for 3D inpainting [65], video diffusion priors [50], NeRF-guided training [136], iterative 3DGS updates [139], and multi-view SDS [7]. General sparse-view baselines include feed-forward prediction [6, 8], regularized optimization [69, 114], and NeRF-based methods [46, 68, 81, 92, 126]. These works require per-scene fine-tuning or focus on single-view outpainting. Our method performs *zero-shot multi-view outpainting* using pretrained MVGenMaster [4] with geometry-aware mechanisms ensuring cross-view consistency without scene-specific training.

3 Preliminaries

3D Gaussian Splatting (3DGS) [32] uses a collection of anisotropic 3D Gaussian primitives to present a scene. Each Gaussian is defined by its center position $\boldsymbol{\mu} \in \mathbb{R}^3$, a 3D covariance matrix $\boldsymbol{\Sigma}$, an opacity value $\alpha \in [0, 1]$, and spherical harmonic coefficients for view-dependent color. The covariance matrix is decomposed into a scaling vector $\mathbf{s} \in \mathbb{R}^3$ and rotation quaternion $\mathbf{q} \in \mathbb{R}^4$ as $\boldsymbol{\Sigma} = \mathbf{RSS}^T\mathbf{R}^T$, where $\mathbf{R}$ is derived from $\mathbf{q}$ and $\mathbf{S} = \text{diag}(\mathbf{s})$. The Gaussian function is:

$$\mathcal{G}(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right). \tag{1}$$

To render a given viewpoint, 3DGS projects each 3D Gaussian onto the 2D image plane, obtaining a 2D Gaussian $\mathcal{G}'(\mathbf{u})$, where $\mathbf{u}$ is pixel coordinates. The color of pixel $\mathbf{u}$ is computed via α-blending of ordered Gaussians:

$$\mathbf{C}(\mathbf{u}) = \sum_{i\in\mathcal{N}} \mathbf{c}_i\sigma_i \prod_{j=1}^{i-1}(1-\sigma_j), \tag{2}$$

where $\mathcal{N}$ denotes the set of Gaussians overlapping pixel $\mathbf{u}$, sorted in depth order, $\mathbf{c}_i$ represents the color of the i-th Gaussian, and $\sigma_i = \alpha_i\mathcal{G}'_i(\mathbf{u})$ is the opacity contribution.

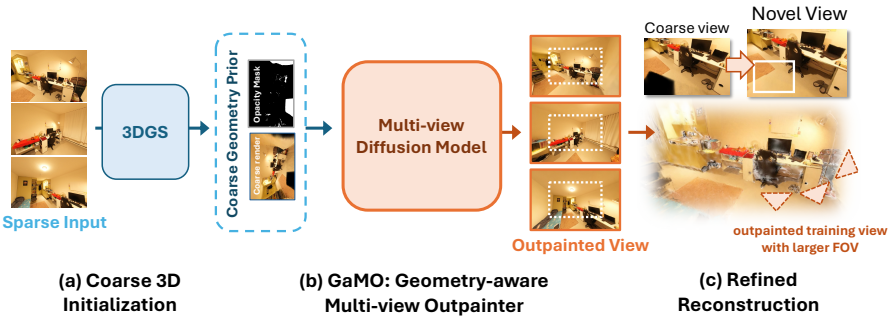


Fig. 3: Overview of Our Pipeline. Given sparse input views, our method follows a three-stage process. **(a) Coarse 3D Initialization:** We obtain geometry priors from initial 3D reconstruction, including an opacity mask and coarse render that provide essential structural cues. **(b) Geometry-aware Multi-view Outpainter:** Using the geometry priors, GaMO generates outpainted views with enlarged FOV via a multi-view diffusion model. **(c) Refined Reconstruction:** The outpainted views are used to refine the 3D reconstruction, resulting in improved completeness and consistency.

Diffusion Models generate samples through a learned denoising process that reverses a forward noising process. The forward process gradually adds Gaussian noise to data $\mathbf{x}_0$ over T timesteps: $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\bar{\alpha}_t$ is a predefined noise schedule. The reverse process learns to denoise $\mathbf{x}_t$ back to $\mathbf{x}_0$ by training a neural network ϵ_θ to predict the noise at each timestep. The training objective is the simplified loss function:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t,\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - \epsilon_\theta(\mathbf{x}_t, t, \mathbf{c})\|^2\right], \tag{3}$$

where $\mathbf{c}$ represents conditioning information, and the model learns to minimize the mean squared error between the true noise $\boldsymbol{\epsilon}$ and the predicted noise. During inference, samples are generated by iteratively denoising from pure noise $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ using the DDIM [79] sampling process.

4 Method

To address challenges in sparse-view 3D reconstruction, we perform geometry-aware outpainting by leveraging multi-view diffusion models. By expanding the field-of-view (FOV) of input images, our method simultaneously fills holes, fixes blurred boundaries, and preserves geometric consistency without modifying existing content, resulting in a significantly simpler and faster reconstruction process.

As illustrated in Fig. 3, our pipeline consists of three stages: coarse 3D initialization to obtain geometry priors (Sec. 4.1), geometry-aware multi-view outpainting to generate enlarged FOV views (Sec. 4.2), and refined 3D reconstruction using the outpainted views (Sec. 4.3).

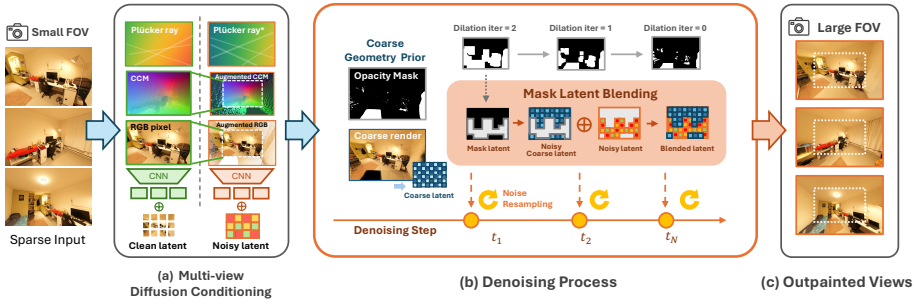


Fig. 4: Overview of GaMO (Geometry-aware Multi-view Diffusion Outpainter). (a) Multi-view Diffusion Conditioning: Sparse input views are encoded into clean latents and combined with multi-view conditions, including Plücker ray embeddings for input views ($\mathcal{P}_r$) and the target view with enlarged FOV ($\mathcal{P}_t^*$), along with original and augmented Canonical Coordinate Map (CCM) and RGB, to provide both geometric and appearance cues for diffusion model conditioning. (b) Denoising Process: Coarse geometry priors (opacity mask and coarse render) guide the denoising through mask latent blending performed at multiple timesteps $(t_1, t_2, ..., t_N)$ with progressive dilation and noise resampling, generating outpainted views with enlarged FOV (c).

4.1 Coarse 3D Initialization

To ensure geometric consistency in the diffusion model, we use DUSt3R [98] to generate an initial point cloud and train a coarse 3DGS model to capture the scene geometry. Using this coarse model, we identify outpainting regions by rendering an opacity mask with a FOV wider than the input views. We also render a coarse color image to provide appearance priors for the outpainting process in Sec. 4.2.

Opacity Mask. We enlarge the FOV by reducing the focal lengths with a scaling ratio $S_k < 1$ (i.e., $f'_x = f_x \times S_k$, $f'_y = f_y \times S_k$). For each target outpainted view, we first render an opacity map $\mathcal{O}$ by α-blending the opacity values of the Gaussians:

$$\mathcal{O}(\mathbf{u}) = \sum_{i \in \mathcal{N}} \sigma_i \prod_{j=1}^{i-1} (1 - \sigma_j), \tag{4}$$

where $\sigma_i = \alpha_i \mathcal{G}'_i(\mathbf{u})$ denotes the opacity contribution of the i-th Gaussian at pixel $\mathbf{u}$. The opacity mask $\mathcal{M}$ is then obtained by thresholding the opacity map with $\mathcal{M} = \mathbb{I}(\mathcal{O} < \eta_{\text{mask}})$, where η_{mask} is a threshold value and $\mathbb{I}(\cdot)$ is the indicator function. Regions where $\mathcal{M} = 1$ correspond to areas with low opacity that require outpainting.

Coarse Rendering. We render a color image I_{coarse} with the enlarged FOV from the coarse 3DGS model. This coarse rendering serves as a reference that provides geometric and appearance priors to the diffusion model, maintaining consistency between outpainted and existing scene content.

4.2 GaMO: Geometry-aware Multi-view Diffusion Outpainter

Our geometry-aware outpainting method operates through three key components: (1) multi-view conditioning that provides structural and appearance guidance; (2) mask latent blending that integrates coarse geometry priors during denoising; and (3) iterative mask scheduling with noise resampling that ensure smooth transitions. The model operates in latent space using DDIM sampling [79] for efficient denoising.

Multi-View Conditioning. Given a set of sparse input RGB images $\{I_i\}_{i=1}^N$ and their corresponding camera parameters $\{\Pi_i\}_{i=1}^N$, our model generates outpainted views conditioned on camera representations, geometric correspondences, and appearance features, as illustrated in Fig. 4(a).

For camera representation, we employ Plücker ray embeddings [122] that provide dense 6D ray parameterizations for each pixel, compactly encoding both ray origin and direction for geometry-aware reasoning. The embedding of each input view $\mathcal{P}_r$ is derived from its corresponding camera parameters Π_r, while the embedding of the outpainted view $\mathcal{P}_t^*$ uses the same camera parameters with scaled focal lengths (f_x', f_y') to align with the enlarged FOV.

For geometric correspondence, we warp input RGB images and Canonical Coordinate Maps (CCM) [43] to align with the expanded FOV by unprojecting pixels to 3D and reprojecting onto the outpainted camera plane, producing $\mathcal{C}_{r\to t}^{\text{warp}}$ and $I_{r\to t}^{\text{warp}}$. We then downsample the original inputs by factor S_k and place them at the center of the warped features, creating augmented signals $I_{r\to t}^{\text{aug}}$ and $\mathcal{C}_{r\to t}^{\text{aug}}$ where the center preserves exact input information while the periphery retains warped geometric structure to guide outpainting.

For appearance features, the input RGB images are encoded through a variational autoencoder (VAE) to obtain clean latent features $\mathbf{z}_r$. The noisy latent features $\mathbf{z}_t$ are randomly generated and will be denoised to generate the outpainted views.

All conditioning signals are processed through lightweight convolutional encoders. For input views, Plücker ray embeddings $\mathcal{P}_r$, CCM $\mathcal{C}_r$, and RGB images I_r are jointly added to the clean latent features $\mathbf{z}_r$. For the target outpainted view, $\mathcal{P}_t^*$, $\mathcal{C}_{r\to t}^{\text{aug}}$, and $I_{r\to t}^{\text{aug}}$ are jointly added to the noisy latent features $\mathbf{z}_t$. We then condition the pre-trained diffusion model with the fused features to generate outpainted view latents in a zero-shot manner:

$$p_\theta(\mathbf{z}_t|\mathbf{z}_r, \mathcal{P}_r, \mathcal{C}_r, I_r, \mathcal{P}_t^*, \mathcal{C}_{r\to t}^{\text{aug}}, I_{r\to t}^{\text{aug}}), \tag{5}$$

where θ denotes the pre-trained model [4] parameters. These multi-view conditions ensure that the diffusion process maintains geometric consistency across views, even under an enlarged FOV.

Denoising Process with Mask Latent Blending. As the central component of our geometry-aware framework, mask latent blending integrates coarse geometry priors from the coarse 3D initialization (Sec. 4.1) into the diffusion loop. As outlined in Alg. 1, this process ensures that outpainted content respects existing scene

structures while generating plausible peripheral regions. Fig. 4(b) shows that the opacity mask $\mathcal{M}$ and coarse rendering I_{coarse} provide consistent structural guidance throughout denoising.

At selected denoising timesteps $\{t_1, t_2, ..., t_N\}$, we perform *mask latent blending* between the denoised latent and the coarse geometry prior. To ensure both latents share the same noise level, we add noise to the coarse latent, which is obtained by encoding the coarse rendering into latent space, before blending them using a latent-space mask $\mathcal{M}_{\text{latent}}$. The mask evolution is controlled by iterative mask scheduling (Sec. 4.2):

$$\mathbf{z}_{t_k}^{\text{blend}} = (1 - \mathcal{M}_{\text{latent}}^{(k)}) \odot \mathbf{z}_{t_k}^{\text{coarse}} + \mathcal{M}_{\text{latent}}^{(k)} \odot \mathbf{z}_{t_k}, \tag{6}$$

where $\mathbf{z}_{t_k}$ is the denoised latent, $\mathbf{z}_{t_k}^{\text{coarse}}$ is the coarse latent with matching noise level (Alg. 1, line 11), $\mathcal{M}_{\text{latent}}^{(k)}$ is the dilated mask at iteration k, and $\odot$ denotes element-wise multiplication.

Iterative Mask Scheduling and Noise Resampling. To gradually integrate generated content with the existing geometric structure, Iterative Mask Scheduling progressively adjusts $\mathcal{M}_{\text{latent}}^{(k)}$ over iterations k to control the ratio between outpainting and known coarse regions. The mask dilation is progressively reduced as denoising proceeds, allowing the model to first explore peripheral content and later refine geometry within coarse regions.

Algorithm 1 Geometry-aware Multi-view Outpainter

1: **Input:** Coarse render I_{coarse}, opacity mask $\mathcal{M}$
2: **Output:** Outpainted views $\{S_j^{\text{out}}\}_{j=1}^{M}$
3: **Setup:** Noise schedule $\Sigma = \{\sigma_1, \ldots, \sigma_T\}$
4: **Setup:** latent blending iterations $\{t_1, \ldots, t_N\}$
5: **Setup:** resampling iterations R
6: $\mathbf{z}^{\text{coarse}} \leftarrow \text{Encode}(I_{\text{coarse}})$
7: $\mathbf{z}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
8: **for** $s = T, \ldots, 1$ **do**
9: $\quad \mathbf{z}_{s-1} = \text{Denoise}(\mathbf{z}_s, \text{conditions})$ ▷ Eq. (5)
10: $\quad$ **if** $s \in \{t_1, \ldots, t_N\}$ **then**
11: $\quad\quad \mathbf{z}_{s-1}^{\text{coarse}} = \text{AddNoise}(\mathbf{z}^{\text{coarse}}, \sigma_{s-1})$
12: $\quad\quad \mathbf{z}_{s-1}^{\text{blend}} = \text{IMS}(\mathcal{M}, \mathbf{z}_{s-1}, \mathbf{z}_{s-1}^{\text{coarse}})$ ▷ Eq. (6)
13: $\quad\quad \mathbf{z}_{s-1} = \mathbf{z}_{s-1}^{\text{blend}}$
14: $\quad\quad$ **for** r in R **do**
15: $\quad\quad\quad \hat{\mathbf{z}}_0 = \text{Predict}(\mathbf{z}_{s-1})$
16: $\quad\quad\quad \mathbf{z}_s^{\text{resamp}} \leftarrow \text{AddNoise}(\hat{\mathbf{z}}_0, \sigma_s)$ ▷ Eq. (7)
17: $\quad\quad\quad \mathbf{z}_{s-1} \leftarrow \text{Denoise}(\mathbf{z}_s^{\text{resamp}}, \text{conditions})$
18: $\quad\quad$ **end for**
19: $\quad$ **end if**
20: **end for**
21: $\{S_j^{\text{out}}\}_{j=1}^{M} = \text{Decode}(\mathbf{z}_0)$

To maintain smooth transitions across blended regions, we perform noise resampling after each blending operation. After blending, we perform noise resampling R times on the blended latent to eliminate boundary artifacts and ensure smooth integration between the coarse geometry and generated content (Alg. 1, lines 14–17). Specifically, we first predict the clean latent $\hat{\mathbf{z}}_0$ from the blended latent, then add noise back to the current timestep t_k:

$$\mathbf{z}_{t_k}^{\text{resamp}} = \sqrt{\bar{\alpha}_{t_k}}\hat{\mathbf{z}}_0 + \sqrt{1 - \bar{\alpha}_{t_k}}\boldsymbol{\epsilon}, \tag{7}$$

where $\hat{\mathbf{z}}_0$ is the predicted clean latent from $\mathbf{z}_{t_k}^{\text{blend}}$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ denotes sampled Gaussian noise. This resampling prevents boundary artifacts and ensures smooth blending.

This framework ensures that outpainted regions seamlessly blend with known content while maintaining geometric plausibility, with the coarse 3DGS geometry

providing structural guidance throughout the generation process. Importantly, it requires only inference without fine-tuning the backbone diffusion model.

4.3 3DGS Refinement with Outpainted Views

Given the original input views $\{I_i^{gt}\}_{i=1}^{N}$ and the generated outpainted views $\{S_j^{\text{out}}\}_{j=1}^{M}$ from Sec. 4.2, we refine the 3DGS model by jointly optimizing with both sets of views. During training, we sample either an input view or an outpainted view for supervision at each iteration.

Loss for Input Views. We employ the standard 3DGS reconstruction loss [32] to ensure accurate reconstruction of the observed regions:

$$\mathcal{L}_{\text{input}} = (1 - \lambda_s)\mathcal{L}_1(I_i, I_i^{gt}) + \lambda_s \mathcal{L}_{\text{D-SSIM}}(I_i, I_i^{gt}), \tag{8}$$

where I_i denotes the rendered image from input viewpoint, I_i^{gt} is the ground truth input view, and λ_s is a weighting factor that balances the $\mathcal{L}_1$ loss and structural similarity loss.

Loss for Outpainted Views. Relying solely on reconstruction loss fails to fill unobserved regions and causes artifacts. We incorporate perceptual loss [28] $\mathcal{L}_{\text{LPIPS}}$ to provide balanced gradients across outpainted and original regions, effectively guiding training while maintaining perceptual consistency. The loss is:

$$\begin{aligned} \mathcal{L}_{\text{recon}} &= (1 - \lambda_s)\mathcal{L}_1(S_j, S_j^{\text{out}}) + \lambda_s \mathcal{L}_{\text{D-SSIM}}(S_j, S_j^{\text{out}}), \\ \mathcal{L}_{\text{outpainted}} &= \mathcal{L}_{\text{recon}}(S_j, S_j^{\text{out}}) + \lambda_{\text{perc}} \mathcal{L}_{\text{LPIPS}}(S_j, S_j^{\text{out}}), \end{aligned} \tag{9}$$

where S_j denotes the rendered wide-FOV image and S_j^{out} is the generated outpainted image.

5 Experiments

5.1 Experimental Setups

Datasets and Evaluation Protocol. We evaluate on **Replica** [80], **ScanNet++** [131], and **Mip-NeRF 360** [3]. Following [149,150], we adopt sparse-view settings with 3, 6, and 9 views for indoor datasets (**Replica** and **ScanNet++**), where the 9-view results are reported in the supplementary material, and 6 and 9 views for the unbounded **Mip-NeRF 360** dataset. For Replica and ScanNet++, we follow the view splits used in [149,150], while for Mip-NeRF 360 we use the original 6- and 9-view configurations. For ScanNet++, we additionally construct the 3- and 9-view setups to ensure suitable coverage for training and evaluation. Performance is evaluated using PSNR, SSIM [102], and LPIPS [145].

Implementation Details. We adopt **MVGenMaster** [4] as the diffusion backbone to incorporate geometry-aware priors in a zero-shot manner. For fair comparison, image resolutions are standardized: **Replica** is set to 512×384, while **ScanNet++** and **Mip-NeRF 360** are set to 576×384. GaMO reconstructs a

Table 1: Quantitative comparison on Replica [80] and ScanNet++ [131] with 3 and 6 input views.

Method	Replica (3 views)			Replica (6 views)			ScanNet++ (3 views)			ScanNet++ (6 views)			Run time
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	
3DGS	20.39	0.818	0.154	24.74	0.862	0.124	16.60	0.710	0.313	21.71	0.818	0.186	2 min
FSGS	20.84	0.815	0.172	23.91	0.846	0.145	16.62	0.690	0.359	21.69	0.801	0.298	12 min
InstantSplat	21.42	0.830	0.137	23.09	0.849	0.141	16.72	0.720	0.312	21.19	0.811	0.193	1 min
Difix3D+	19.47	0.783	0.194	21.86	0.811	0.188	15.64	0.659	0.346	20.62	0.764	0.244	31 min
GenFusion	22.34	0.833	0.172	23.98	0.855	0.142	17.97	0.725	0.354	21.96	0.808	0.218	22 min
GuidedVD-3DGS†	23.98	0.848	0.136	26.35	0.872	0.122	-	-	-	23.89	0.850	0.182	-
GuidedVD-3DGS‡	25.26	0.864	0.138	26.68	0.880	0.133	18.82	0.720	0.328	22.98	0.815	0.204	3h 20 min
Ours	24.40	0.865	0.117	26.40	0.882	0.104	20.06	0.759	0.265	23.41	0.835	0.181	8 min

† Reported in paper, ‡ Our reproduction.

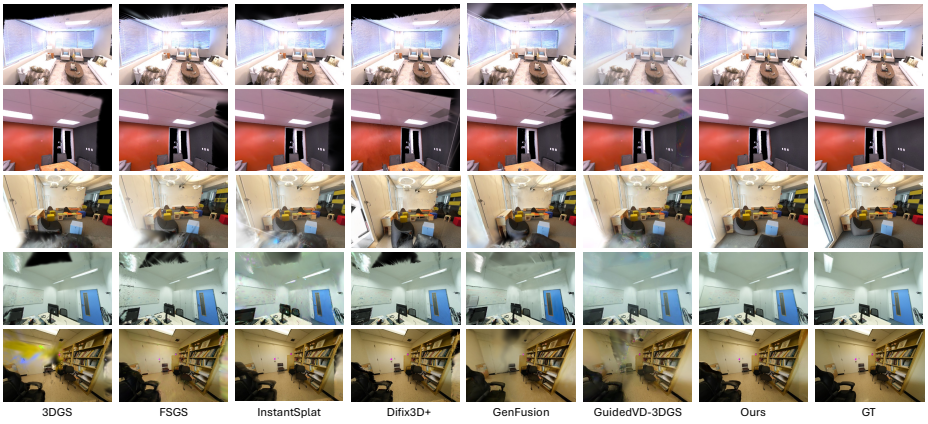


Fig. 5: Qualitative comparison on Replica and ScanNet++ under sparse 3-view and 6-view settings. Our method produces better coverage, improved geometric consistency, and fewer artifacts. White boxes highlight challenging regions.

scene in approximately 8 minutes on a single RTX 4090 GPU, including diffusion outpainting and the final 3DGS optimization. Additional implementation details, hyperparameters, optimization settings, runtime breakdown, and extended results are provided in the supplementary material.

Baselines. We compare against several state-of-the-art sparse-view reconstruction methods: (1) vanilla 3DGS; (2) **FSGS** [152], using depth-guided Gaussian unpooling; (3) **InstantSplat** [15], employing MASt3R priors and self-supervised bundle adjustment; (4) **Difix3D** [106], applying single-step diffusion refinement; (5) **GenFusion** [110], integrating reconstruction and video diffusion via cyclical fusion; and (6) **GuidedVD-3DGS** [149], leveraging video diffusion and evaluated using the authors' official implementation and settings. For fair comparison, all methods except InstantSplat use DUSt3R [98] for initialization.

Table 2: Quantitative comparison on MipNeRF360 [3] with 6 and 9 input views.

Method	Mip-NeRF 360 (6 views)			Mip-NeRF 360 (9 views)		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
3DGS	15.30	0.342	0.459	16.29	0.383	0.385
FSGS	14.76	0.326	0.532	15.36	0.374	0.502
InstantSplat	15.91	0.388	0.443	16.10	0.433	0.419
Difix3D+	14.94	0.308	0.419	16.04	0.368	0.371
GenFusion	16.40	0.384	0.487	17.55	0.435	0.409
GuidedVD-3DGS	13.89	0.273	0.640	15.77	0.386	0.418
Ours	16.74	0.393	0.436	17.56	0.448	0.381



Fig. 6: Qualitative comparison on MipNeRF360 [3]under sparse 6-view and 9-view settings. Large-scale outdoor scenes with wide baselines make sparse-view reconstruction challenging. Our method achieves better coverage, improved geometric consistency, and fewer artifacts than prior methods. White boxes highlight challenging regions.

5.2 Comparisons

Quantitative Results. Tab. 1 and Tab. 2 report comparisons on Replica, ScanNet++, and Mip-NeRF 360 under sparse-view settings. Our method consistently achieves strong performance across all datasets. On Replica and ScanNet++, we obtain the best SSIM and lowest LPIPS in most settings, indicating superior structural consistency and perceptual quality. On the challenging Mip-NeRF 360 benchmark, our method achieves the best PSNR and SSIM for both 6-view and 9-view inputs while maintaining competitive LPIPS. In addition, GaMO is highly efficient, achieving up to 25× speedup over the diffusion-based GuidedVD-3DGS [149], reducing the reconstruction time to under 10 minutes.

Qualitative Results. Fig. 13 and Fig. 6 show visual comparisons on representative scenes. Our method produces more complete reconstructions by effectively addressing the key challenges in sparse-view reconstruction: reducing black holes in unobserved regions, minimizing rendering artifacts, and improving geometric

Table 3: Quantitative ablation of blending components. We evaluate the impact of augmented conditioning (Aug.), hard (H.)/soft (S.) mask blending, and noise resampling (N.). P, S, L denote PSNR↑, SSIM↑, and LPIPS↓, respectively. Row 5 represents our full model.

	Components				Outpainted			Novel View		
#	Aug.	H.	S.	N.	P	S	L	P	S	L
1					18.97	.776	.210	22.37	.821	.197
2	✓				19.11	.779	.207	22.53	.822	.192
3	✓	✓			19.77	.797	.199	23.52	.839	.174
4	✓		✓	✓	19.37	.797	.196	23.22	**.840**	.173
5*	✓	✓	✓	✓	**20.01**	**.800**	**.190**	**23.53**	.839	**.172**

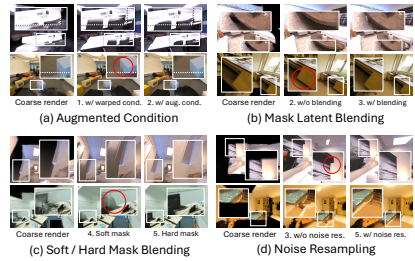


Fig. 7: Qualitative ablation. (a) Aug. cond. aligns content. (b-c) Mask blending improves geometry. (d) Noise resampling removes seams.

consistency. These results demonstrate the effectiveness of GaMO in generating high-quality outpainted views that enhance 3D reconstruction.

5.3 Ablation Studies

We conduct comprehensive ablation studies on Replica and ScanNet++ datasets with 6 input views. To separately assess the outpainted view quality and novel view synthesis performance, we center crop the input images to 0.6× of their original size. For each ablation, we provide quantitative results and visual comparisons, where numbered configurations (e.g., "1.") in figures correspond to table rows, and letters (e.g., (a), (b)) denote different visual comparison aspects.

Latent Blending Strategies. Tab. 3 and Fig. 10 present ablation results on our latent blending design. Augmenting the warped features with downscaled reference RGB and CCM (rows 1-2) prevents incorrect hallucinations in known regions (a). Mask latent blending (rows 2-3) prevents severe geometric misalignment (b, red circle) and improves PSNR by 0.66 dB. Hard masking (rows 4-5) produces sharper boundaries (c) with 0.64 dB gain over soft masking. Finally, noise resampling (rows 3 vs. 5) reduces blending artifacts by 0.24 dB, generating more coherent results (d).

Mask Blending Scheduling. Tab. 4 and Fig. 8 present ablation results on mask blending scheduling strategies. Single-step blending (row 1) is insufficient as coarse geometry is easily washed out during denoising, while multi-step blending (row 2) better preserves geometric cues (a-top). Blending at every step (row 3) achieves slightly higher PSNR/SSIM but causes blurred boundaries (a-bottom) and increases denoising time, making the range-based approach preferable. Finally, Iterative Mask Scheduling (rows 2 vs. 4) substantially improves perceptual quality through progressive mask dilation, providing better geometric guidance and smoother transitions for more coherent details (b).

Table 4: Ablation on mask blending scheduling. Comparison of blending at different timesteps: t_k (single-step), $t_{1\to t_N}$ (multi-step), All (every step), and IMS (Iterative Mask Scheduling). Row 4* is our full method. *Time (s)* is total generation time.

	Scheduling				Outpainted				Novel View		
#	t_k	$t_{1\to t_N}$	All	IMS	P↑	S↑	L↓	T(s)↓	P↑	S↑	L↓
1	✓				20.09	.804	.198	**85**	23.38	.837	.176
2		✓			19.85	.799	.173	93	23.53	.839	.179
3			✓		**20.31**	**.809**	.201	167	**23.67**	**.842**	.178
4*		✓		✓	20.07	.801	**.169**	93	23.65	.839	**.171**

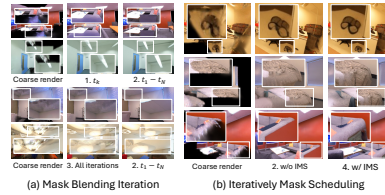


Fig. 8: Qualitative ablation. (a) Blending timesteps impact. (b) IMS progressively improves detail coherence and alignment.

Table 5: Ablation on 3DGS refinement. Impact of point re-init. and perceptual loss on reconstruction quality. Row 3* is our full method.

	Components		Novel View		
#	Re-init.	Percep.	P↑	S↑	L↓
1		✓	24.80	.860	.140
2	✓		**25.14**	.857	.139
3*	✓	✓	24.93	**.861**	**.135**



Fig. 9: Qualitative ablation. (a) Re-init. enables Gaussian generation in new regions. (b) Perceptual loss recovers sharp details.

3DGS Refinement Components. Tab. 5 and Fig. 9 present ablation results on 3DGS refinement components. Point cloud re-initialization using outpainted views (rows 1 vs. 3) enables the successful generation of Gaussian points in outpainted regions (a). Perceptual loss (rows 2 vs. 3) effectively fills holes and reduces artifacts by providing better gradient guidance for outpainted regions (b), producing cleaner and more realistic renderings.

6 Conclusion

We present *GaMO*, which formulates novel view generation as an outpainting problem for sparse-view 3D reconstruction. By extending existing views instead of generating new perspectives, our approach preserves geometric consistency while expanding spatial coverage, effectively reducing holes and artifacts. Experiments show consistent improvements over prior methods with superior reconstruction quality and several tens of times faster runtime. GaMO also demonstrates strong zero-shot generalization, highlighting outpainting as an efficient and principled paradigm for sparse-view reconstruction.

Limitations. GaMO cannot recover content fully occluded from all input views. Its performance also depends on input view distribution, where clustered or misaligned views may degrade results. Future work could explore adaptive outpaint scaling and hybrid strategies for more challenging cases.

Acknowledgements

This research was funded by the National Science and Technology Council, Taiwan, under Grants NSTC 112-2222-E-A49-004-MY2 and 113-2628-E-A49-023-. The authors are grateful to Google, NVIDIA, and MediaTek Inc. for their generous donations. Yu-Lun Liu acknowledges the Yushan Young Fellow Program by the MOE in Taiwan.

References

1. Ancikevičius, T., Xu, Z., Fisher, M., Henderson, P., Bilen, H., Mitra, N.J., Guerrero, P.: Renderdiffusion: Image diffusion for 3d reconstruction, inpainting and generation. In: CVPR (2023)
2. Bahmani, S., Skorokhodov, I., Siarohin, A., Menapace, W., Qian, G., Vasilkovsky, M., Lee, H.Y., Wang, C., Zou, J., Tagliasacchi, A., Lindell, D.B., Tulyakov, S.: VD3D: Taming large video diffusion transformers for 3D camera control. In: International Conference on Learning Representations (ICLR) (2025)
3. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5470–5479 (2022)
4. Cao, C., Yu, C., Liu, S., Wang, F., Xue, X., Fu, Y.: Mvgenmaster: Scaling multi-view generation from any image via 3d priors enhanced diffusion model. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6045–6056 (2025)
5. Chan, E.R., Nagano, K., Chan, M.A., Bergman, A.W., Park, J.J., Levy, A., Aittala, M., De Mello, S., Karras, T., Wetzstein, G.: Generative novel view synthesis with 3d-aware diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4217–4229 (2023)
6. Charatan, D., Li, S.L., Tagliasacchi, A., Sitzmann, V.: pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19457–19467 (2024)
7. Chen, H., Loy, C.C., Pan, X.: Mvip-nerf: Multi-view 3d inpainting on nerf scenes via diffusion prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5344–5353 (2024)
8. 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. In: European Conference on Computer Vision. pp. 370–386. Springer (2024)
9. Chen, Y., Zheng, C., Xu, H., Zhuang, B., Vedaldi, A., Cham, T.J., Cai, J.: Mvsplat360: Feed-forward 360 scene synthesis from sparse views. Advances in Neural Information Processing Systems **37**, 107064–107086 (2024)
10. Chen, Z., Wang, Y., Wang, F., Wang, Z., Liu, H.: V3d: Video diffusion models are effective 3d generators. arXiv preprint arXiv:2403.06738 (2024)
11. Chen, Z., Su, R., Zhu, J., Yang, L., Lai, J.H., Xie, X.: Vividdreamer: Towards high-fidelity and efficient text-to-3d generation. arXiv preprint arXiv:2406.14964 (2024)
12. Chung, J., Oh, J., Lee, K.M.: Depth-regularized optimization for 3d Gaussian Splatting in few-shot images. arXiv preprint arXiv:2311.13398 (2023)

13. Duan, Y., Guo, X., Zhu, Z.: Diffusiondepth: Diffusion denoising approach for monocular depth estimation. In: European Conference on Computer Vision. pp. 432–449. Springer (2024)
14. Duisterhof, B.P., Zust, L., Weinzaepfel, P., Leroy, V., Cabon, Y., Revaud, J.: MASt3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. In: International Conference on 3D Vision 2025 (2025), https://openreview.net/forum?id=5uw1GRBFoT
15. Fan, Z., Cong, W., Wen, K., Wang, K., Zhang, J., Ding, X., Xu, D., Ivanovic, B., Pavone, M., Pavlakos, G., Wang, Z., Wang, Y.: InstantSplat: Sparse-view Gaussian Splatting in seconds. arXiv preprint arXiv:2403.20309 (2024)
16. Feng, M., Liu, J., Cui, M., Xie, X.: Diffusion360: Seamless 360 degree panoramic image generation based on diffusion models. arXiv preprint arXiv:2311.13141 (2023)
17. Fu, X., Yin, W., Hu, M., Wang, K., Ma, Y., Tan, P., Shen, S., Lin, D., Long, X.: Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In: European Conference on Computer Vision. pp. 241–258. Springer (2024)
18. Gao, R., Holynski, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P., Barron, J.T., Poole, B.: Cat3d: Create anything in 3d with multi-view diffusion models. arXiv preprint arXiv:2405.10314 (2024)
19. Guizilini, V., Irshad, M.Z., Chen, D., Shakhnarovich, G., Ambrus, R.: Zero-shot novel view and depth synthesis with multi-view geometric diffusion. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 764–776 (2025)
20. He, H., Xu, Y., Guo, Y., Wetzstein, G., Dai, B., Li, H., Yang, C.: Cameractrl: Enabling camera control for text-to-video generation. arXiv preprint arXiv:2404.02101 (2024)
21. He, X., Chen, J., Peng, S., Huang, D., Li, Y., Huang, X., Yuan, C., Ouyang, W., He, T.: Gvgen: Text-to-3d generation with volumetric representation. In: European Conference on Computer Vision. pp. 463–479. Springer (2024)
22. He, Z., Xiao, Z., Chan, K.C., Zuo, Y., Xiao, J., Lam, K.M.: See in detail: Enhancing sparse-view 3d gaussian splatting with local depth and semantic regularization. In: ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1–5. IEEE (2025)
23. Hu, H., Zhou, Z., Jampani, V., Tulsiani, S.: Mvd-fusion: Single-view 3d via depth-consistent multi-view generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9698–9707 (2024)
24. Huang, Z., Wen, H., Dong, J., Wang, Y., Li, Y., Chen, X., Cao, Y.P., Liang, D., Qiao, Y., Dai, B., et al.: Epidiff: Enhancing multi-view synthesis via localized epipolar-constrained diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9784–9794 (2024)
25. Hui, M., Wei, Z., Zhu, H., Xia, F., Zhou, Y.: Microdiffusion: Implicit representation-guided diffusion for 3d reconstruction from limited 2d microscopy projections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11460–11469 (2024)
26. Jain, A., Tancik, M., Abbeel, P.: Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: ICCV (2021)
27. Ji, C., Yu, C., Gao, J., Wang, F., Zhao, C.: Campvg: Camera-controlled panoramic video generation with epipolar-aware diffusion. arXiv preprint arXiv:2509.19979 (2025)

28. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision (ECCV). pp. 694–711. Springer (2016)
29. Kalischek, N., Oechsle, M., Manhardt, F., Henzler, P., Schindler, K., Tombari, F.: Cubediff: Repurposing diffusion-based image models for panorama generation. In: The Thirteenth International Conference on Learning Representations (2025)
30. Kant, Y., Siarohin, A., Wu, Z., Vasilkovsky, M., Qian, G., Ren, J., Guler, R.A., Ghanem, B., Tulyakov, S., Gilitschenski, I.: Spad: Spatially aware multi-view diffusers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10026–10038 (2024)
31. Ke, B., Obukhov, A., Huang, S., Metzger, N., Daudt, R.C., Schindler, K.: Repurposing diffusion-based image generators for monocular depth estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9492–9502 (2024)
32. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
33. Kong, H., Yang, X., Wang, X.: Generative sparse-view gaussian splatting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26745–26755 (2025)
34. Kuang, Z., Cai, S., He, H., Xu, Y., Li, H., Guibas, L.J., Wetzstein, G.: Collaborative video diffusion: Consistent multi-video generation with camera control. Advances in Neural Information Processing Systems **37**, 16240–16271 (2024)
35. Kupyn, O., Manhardt, F., Tombari, F., Rupprecht, C.: Epipolar geometry improves video generation models. arXiv preprint arXiv:2510.21615 (2025)
36. Kwak, J.g., Dong, E., Jin, Y., Ko, H., Mahajan, S., Yi, K.M.: Vivid-1-to-3: Novel view synthesis with video diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6775–6785 (2024)
37. Lan, Y., Hong, F., Yang, S., Zhou, S., Meng, X., Dai, B., Pan, X., Loy, C.C.: Ln3diff: Scalable latent neural fields diffusion for speedy 3d generation. In: European Conference on Computer Vision. pp. 112–130. Springer (2024)
38. Lee, K., Sohn, K., Shin, J.: Dreamflow: High-quality text-to-3d generation by approximating probability flow. arXiv preprint arXiv:2403.14966 (2024)
39. Li, B., Zheng, C., Zhu, W., Mai, J., Zhang, B., Wonka, P., Ghanem, B.: Vivid-zoo: Multi-view video generation with diffusion model. Advances in Neural Information Processing Systems **37**, 62189–62222 (2024)
40. 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: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20775–20785 (2024)
41. Li, M.F., Ku, Y.F., Yen, H.X., Liu, C., Liu, Y.L., Chen, A.Y., Kuo, C.H., Sun, M.: Genrc: Generative 3d room completion from sparse image collections. In: European Conference on Computer Vision. pp. 146–163. Springer (2024)
42. Li, P., Liu, Y., Long, X., Zhang, F., Lin, C., Li, M., Qi, X., Zhang, S., Luo, W., Tan, P., Wang, W., Liu, Q., Guo, Y.: Era3D: High-resolution multiview diffusion using efficient row-wise attention. In: Advances in Neural Information Processing Systems (NeurIPS) (2024)
43. Li, W., Chen, R., Chen, X., Tan, P.: Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. In: International Conference on Learning Representations (ICLR) (2024)

44. Liang, Y., Yang, X., Lin, J., Li, H., Xu, X., Chen, Y.: Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6517–6526 (2024)
45. Lin, C.Y., Sun, C., Yang, F.E., Chen, M.H., Lin, Y.Y., Liu, Y.L.: Longsplat: Robust unposed 3d gaussian splatting for casual long videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 27412–27422 (2025)
46. Lin, C.Y., Wu, C.H., Yeh, C.H., Yen, S.H., Sun, C., Liu, Y.L.: Frugalnerf: Fast convergence for extreme few-shot novel view synthesis without learned priors. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 11227–11238 (2025)
47. Liu, A., Li, Z., Chen, Z., Li, N., Xu, Y., Plummer, B.A.: Panofree: Tuning-free holistic multi-view image generation with cross-view self-guidance. In: European Conference on Computer Vision. pp. 146–164. Springer (2024)
48. Liu, F., Sun, W., Wang, H., Wang, Y., Sun, H., Ye, J., Zhang, J., Duan, Y.: Reconx: Reconstruct any scene from sparse views with video diffusion model. arXiv preprint arXiv:2408.16767 (2024)
49. Liu, K.H., Yang, C.K., Chen, M.H., Liu, Y.L., Lin, Y.Y.: Corrfill: Enhancing faithfulness in reference-based inpainting with correspondence guidance in diffusion models. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 1618–1627. IEEE (2025)
50. Liu, K., Shao, L., Lu, S.: Novel view extrapolation with video diffusion priors. arXiv preprint arXiv:2411.14208 (2024)
51. Liu, M., Shi, R., Chen, L., Zhang, Z., Xu, C., Wei, X., Chen, H., Zeng, C., Gu, J., Su, H.: One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10072–10083 (2024)
52. Liu, M., Zeng, C., Wei, X., Shi, R., Chen, L., Xu, C., Zhang, M., Wang, Z., Zhang, X., Liu, I., Wu, H., Su, H.: MeshFormer: High-quality mesh generation with 3D-guided reconstruction model. arXiv preprint arXiv:2408.10198 (2024)
53. Liu, T., Wang, G., Hu, S., Shen, L., Ye, X., Zang, Y., Cao, Z., Li, W., Liu, Z.: Mvsgaussian: Fast generalizable gaussian splatting reconstruction from multi-view stereo. In: European Conference on Computer Vision. pp. 37–53. Springer (2024)
54. Liu, X., Zhou, C., Huang, S.: 3dgs-enhancer: Enhancing unbounded 3d gaussian splatting with view-consistent 2d diffusion priors. Advances in Neural Information Processing Systems **37**, 133305–133327 (2024)
55. Liu, X., Chen, J., Kao, S.h., Tai, Y.W., Tang, C.K.: Deceptive-NeRF/3DGS: Diffusion-generated pseudo-observations for high-quality sparse-view reconstruction. In: European Conference on Computer Vision (ECCV) (2024)
56. Liu, Y., Lin, C., Zeng, Z., Long, X., Liu, L., Komura, T., Wang, W.: Syncdreamer: Generating multiview-consistent images from a single-view image. arXiv preprint arXiv:2309.03453 (2023)
57. Long, X., Guo, Y.C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.H., Habermann, M., Theobalt, C., et al.: Wonder3d: Single image to 3d using cross-domain diffusion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9970–9980 (2024)
58. Lu, Y., Zhang, J., Li, S., Fang, T., McKinnon, D., Tsin, Y., Quan, L., Cao, X., Yao, Y.: Direct2. 5: Diverse text-to-3d generation via multi-view 2.5 d diffusion.

In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8744–8753 (2024)
59. Lu, Z., Hu, K., Wang, C., Bai, L., Wang, Z.: Autoregressive omni-aware outpainting for open-vocabulary 360-degree image generation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 14211–14219 (2024)
60. Lukoianov, A., Borde, H.S.d.O., Greenewald, K., Guizilini, V.C., Bagautdinov, T., Sitzmann, V., Solomon, J.: Score distillation via reparametrized DDIM. In: Advances in Neural Information Processing Systems (NeurIPS) (2024)
61. McAllister, D., Ge, S., Huang, J.B., Jacobs, D., Efros, A., Holynski, A., Kanazawa, A.: Rethinking score distillation as a bridge between image distributions. Advances in Neural Information Processing Systems **37**, 33779–33804 (2024)
62. Melas-Kyriazi, L., Laina, I., Rupprecht, C., Neverova, N., Vedaldi, A., Gafni, O., Kokkinos, F.: Im-3d: Iterative multiview diffusion and reconstruction for high-quality 3d generation. arXiv preprint arXiv:2402.08682 (2024)
63. Mihajlovic, M., Prokudin, S., Tang, S., Maier, R., Bogo, F., Tung, T., Boyer, E.: Splatfields: Neural gaussian splats for sparse 3d and 4d reconstruction. In: European Conference on Computer Vision. pp. 313–332. Springer (2024)
64. Min, Z., Luo, Y., Sun, J., Yang, Y.: Epipolar-free 3d gaussian splatting for generalizable novel view synthesis. Advances in Neural Information Processing Systems **37**, 39573–39596 (2024)
65. Mirzaei, A., De Lutio, R., Kim, S.W., Acuna, D., Kelly, J., Fidler, S., Gilitschenski, I., Gojcic, Z.: Reffusion: Reference adapted diffusion models for 3d scene inpainting. arXiv preprint arXiv:2404.10765 (2024)
66. Müller, N., Schwarz, K., Rössle, B., Porzi, L., Bulo, S.R., Nießner, M., Kontschieder, P.: Multidiff: Consistent novel view synthesis from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10258–10268 (2024)
67. Ni, J., Liu, Y., Lu, R., Zhou, Z., Zhu, S.C., Chen, Y., Huang, S.: Decompositional neural scene reconstruction with generative diffusion prior. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6022–6033 (2025)
68. Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S., Geiger, A., Radwan, N.: Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In: CVPR (2022)
69. Paliwal, A., Ye, W., Xiong, J., Kotovenko, D., Ranjan, R., Chandra, V., Kalantari, N.K.: Coherentgs: Sparse novel view synthesis with coherent 3d gaussians. In: European Conference on Computer Vision. pp. 19–37. Springer (2024)
70. Park, H., Ryu, G., Kim, W.: Dropgaussian: Structural regularization for sparse-view gaussian splatting. In: Proceedings of the computer vision and pattern recognition conference. pp. 21600–21609 (2025)
71. Patni, S., Agarwal, A., Arora, C.: Ecodepth: Effective conditioning of diffusion models for monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 28285–28295 (2024)
72. Peng, R., Xu, W., Tang, L., Liao, L., Jiao, J., Wang, R.: SCGaussian: Structure consistent Gaussian Splatting with matching prior for few-shot novel view synthesis. In: Advances in Neural Information Processing Systems (NeurIPS) (2024)
73. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
74. Qiu, L., Chen, G., Gu, X., Zuo, Q., Xu, M., Wu, Y., Yuan, W., Dong, Z., Bo, L., Han, X.: Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9914–9925 (2024)

75. Shi, H., Li, Y., Yang, K., Zhang, J., Peng, K., Roitberg, A., Ye, Y., Ni, H., Wang, K., Stiefelhagen, R.: Fishdreamer: Towards fisheye semantic completion via unified image outpainting and segmentation. arXiv preprint arXiv:2303.13842 (2023)
76. Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., Chen, L., Zeng, C., Su, H.: Zero123++: a single image to consistent multi-view diffusion base model. arXiv preprint arXiv:2310.15110 (2023)
77. Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. arXiv preprint arXiv:2308.16512 (2023)
78. Shih, M.L., Chen, Y.H., Liu, Y.L., Curless, B.: Prior-enhanced gaussian splatting for dynamic scene reconstruction from casual video. arXiv preprint arXiv:2512.11356 (2025)
79. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: International Conference on Learning Representations (2021)
80. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., et al.: Replica: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
81. Su, C.H., Hu, C.Y., Tsai, S.R., Lee, J.Y., Lin, C.Y., Liu, Y.L.: Boostmvsnerfs: Boosting mvs-based nerfs to generalizable view synthesis in large-scale scenes. In: ACM SIGGRAPH 2024 Conference Papers. pp. 1–12 (2024)
82. Szymanowicz, S., Rupprecht, C., Vedaldi, A.: Splatter image: Ultra-fast single-view 3d reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10208–10217 (2024)
83. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In: European Conference on Computer Vision. pp. 1–18. Springer (2024)
84. Tang, S., Ye, W., Ye, P., Lin, W., Zhou, Y., Chen, T., Ouyang, W.: Hisplat: Hierarchical 3d gaussian splatting for generalizable sparse-view reconstruction. arXiv preprint arXiv:2410.06245 (2024)
85. Tang, S., Chen, J., Wang, D., Tang, C., Zhang, F., Fan, Y., Chandra, V., Furukawa, Y., Ranjan, R.: Mvdiffusion++: A dense high-resolution multi-view diffusion model for single or sparse-view 3d object reconstruction. In: European Conference on Computer Vision. pp. 175–191. Springer (2024)
86. Tang, S., Zhang, F., Chen, J., Wang, P., Furukawa, Y.: Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. Advances in Neural Information Processing Systems (2023)
87. Tsai, S.R., Chang, W.C., Lee, J.Y., Su, C.H., Liu, Y.L.: Lightsout: Diffusion-based outpainting for enhanced lens flare removal. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6353–6363 (2025)
88. Tu, T., Chuang, S.P., Liu, Y.L., Sun, C., Zhang, K., Roy, D., Kuo, C.H., Sun, M.: Imgeonet: Image-induced geometry-aware voxel representation for multi-view 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6996–7007 (2023)
89. Turkulainen, M., Ren, X., Melekhov, I., Seiskari, O., Rahtu, E., Kannala, J.: Dn-splatter: Depth and normal priors for gaussian splatting and meshing. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 2421–2431. IEEE (2025)
90. Voleti, V., Yao, C.H., Boss, M., Letts, A., Pankratz, D., Tochilkin, D., Laforte, C., Rombach, R., Jampani, V.: Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In: European Conference on Computer Vision. pp. 439–457. Springer (2024)

91. Wang, F.Y., Wu, X., Huang, Z., Shi, X., Shen, D., Song, G., Liu, Y., Li, H.: Be-your-outpainter: Mastering video outpainting through input-specific adaptation. In: European Conference on Computer Vision. pp. 153–168. Springer (2024)
92. Wang, G., Chen, Z., Loy, C.C., Liu, Z.: Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In: ICCV (2023)
93. Wang, N.H., Liu, Y.L.: Depth Anywhere: Enhancing 360 monocular depth estimation via perspective distillation and unlabeled data augmentation. In: Advances in Neural Information Processing Systems (NeurIPS) (2024)
94. Wang, P., Xu, D., Fan, Z., Wang, D., Mohan, S., Iandola, F., Ranjan, R., Li, Y., Liu, Q., Wang, Z., Chandra, V.: Taming mode collapse in score distillation for text-to-3D generation. arXiv preprint arXiv:2401.00909 (2024)
95. Wang, P., Shi, Y.: Imagedream: Image-prompt multi-view diffusion for 3d generation. arXiv preprint arXiv:2312.02201 (2023)
96. Wang, Q., Li, W., Mou, C., Cheng, X., Zhang, J.: 360dvd: Controllable panorama video generation with 360-degree video diffusion model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6913–6923 (2024)
97. Wang, Q., Zhao, Y., Ma, J., Li, J.: How to use diffusion priors under sparse views? Advances in Neural Information Processing Systems **37**, 30394–30424 (2024)
98. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20697–20709 (2024)
99. Wang, Y., Huang, T., Chen, H., Lee, G.H.: Freesplat: Generalizable 3d gaussian splatting towards free view synthesis of indoor scenes. Advances in Neural Information Processing Systems **37**, 107326–107349 (2024)
100. Wang, Z., Xu, Q., Tan, F., Chai, M., Liu, S., Pandey, R., Fanello, S., Kadambi, A., Zhang, Y.: Mvdd: Multi-view depth diffusion models. In: European Conference on Computer Vision. pp. 236–253. Springer (2024)
101. Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. Advances in neural information processing systems **36**, 8406–8441 (2023)
102. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)
103. Wang, Z., Yuan, Z., Wang, X., Li, Y., Chen, T., Xia, M., Luo, P., Shan, Y.: Motionctrl: A unified and flexible motion controller for video generation. In: ACM SIGGRAPH 2024 Conference Papers. pp. 1–11 (2024)
104. Weber, E., Holynski, A., Jampani, V., Saxena, S., Snavely, N., Kar, A., Kanazawa, A.: Nerfiller: Completing scenes via generative 3d inpainting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20731–20741 (2024)
105. Wu, C.H., Chen, Y.J., Chen, Y.H., Lee, J.Y., Ke, B.H., Mu, C.W.T., Huang, Y.C., Lin, C.Y., Chen, M.H., Lin, Y.Y., et al.: Aurafusion360: Augmented unseen region alignment for reference-based 360deg unbounded scene inpainting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16366–16376 (2025)
106. Wu, J.Z., Zhang, Y., Turki, H., Ren, X., Gao, J., Shou, M.Z., Fidler, S., Gojcic, Z., Ling, H.: Difix3d+: Improving 3d reconstructions with single-step diffusion models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26024–26035 (2025)

107. Wu, K., Liu, F., Cai, Z., Yan, R., Wang, H., Hu, Y., Duan, Y., Ma, K.: Unique3d: High-quality and efficient 3d mesh generation from a single image. Advances in Neural Information Processing Systems **37**, 125116–125141 (2024)
108. Wu, R., Mildenhall, B., Henzler, P., Park, K., Gao, R., Watson, D., Srinivasan, P.P., Verbin, D., Barron, J.T., Poole, B., et al.: Reconfusion: 3d reconstruction with diffusion priors. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 21551–21561 (2024)
109. Wu, S., Lin, Y., Zhang, F., Zeng, Y., Xu, J., Torr, P., Cao, X., Yao, Y.: Direct3d: Scalable image-to-3d generation via 3d latent diffusion transformer. Advances in Neural Information Processing Systems **37**, 121859–121881 (2024)
110. Wu, S., Xu, C., Huang, B., Geiger, A., Chen, A.: Genfusion: Closing the loop between reconstruction and generation via videos. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6078–6088 (2025)
111. Wu, T., Zheng, C., Cham, T.J.: Panodiffusion: 360-degree panorama outpainting via diffusion. arXiv preprint arXiv:2307.03177 (2023)
112. Wu, Z., Zhou, P., Yi, X., Yuan, X., Zhang, H.: Consistent3d: Towards consistent high-fidelity text-to-3d generation with deterministic sampling prior. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9892–9902 (2024)
113. Xie, D., Li, J., Tan, H., Sun, X., Shu, Z., Zhou, Y., Bi, S., Pirk, S., Kaufman, A.E.: Carve3d: Improving multi-view reconstruction consistency for diffusion models with rl finetuning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6369–6379 (2024)
114. Xu, C., Li, A., Chen, L., Liu, Y., Shi, R., Su, H., Liu, M.: Sparp: Fast 3d object reconstruction and pose estimation from sparse views. In: European Conference on Computer Vision. pp. 143–163. Springer (2024)
115. Xu, C., Ling, H., Fidler, S., Litany, O.: 3difftection: 3d object detection with geometry-aware diffusion features. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10617–10627 (2024)
116. Xu, D., Jiang, Y., Huang, C., Song, L., Gernoth, T., Cao, L., Wang, Z., Tang, H.: Cavia: Camera-controllable multi-view video diffusion with view-integrated attention. arXiv preprint arXiv:2410.10774 (2024)
117. Xu, D., Nie, W., Liu, C., Liu, S., Kautz, J., Wang, Z., Vahdat, A.: Camco: Camera-controllable 3d-consistent image-to-video generation. arXiv preprint arXiv:2406.02509 (2024)
118. Xu, H., Lei, Y., Chen, Z., Zhang, X., Zhao, Y., Wang, Y., Tu, Z.: Bayesian diffusion models for 3d shape reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10628–10638 (2024)
119. Xu, H., Peng, S., Wang, F., Blum, H., Barath, D., Geiger, A., Pollefeys, M.: Depthsplat: Connecting gaussian splatting and depth. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16453–16463 (2025)
120. Xu, J., Cheng, W., Gao, Y., Wang, X., Gao, S., Shan, Y.: Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. arXiv preprint arXiv:2404.07191 (2024)
121. Xu, J., Gao, S., Shan, Y.: Freesplatter: Pose-free gaussian splatting for sparse-view 3d reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 25442–25452 (2025)
122. Xu, Y., Tan, H., Luan, F., Bi, S., Wang, P., Li, J., Shi, Z., Sunkavalli, K., Wetzstein, G., Xu, Z., Zhang, K.: Dmv3d: Denoising multi-view diffusion using 3d large reconstruction model. In: International Conference on Learning Representations (2024)

123. Xu, Y., Tan, H., Luan, F., Bi, S., Wang, P., Li, J., Shi, Z., Sunkavalli, K., Wetzstein, G., Xu, Z., et al.: Dmv3d: Denoising multi-view diffusion using 3d large reconstruction model. arXiv preprint arXiv:2311.09217 (2023)
124. Xue, Y., Xie, X., Marin, R., Pons-Moll, G.: Human-3diffusion: Realistic avatar creation via explicit 3d consistent diffusion models. Advances in Neural Information Processing Systems **37**, 99601–99645 (2024)
125. Yang, C., Li, S., Fang, J., Liang, R., Xie, L., Zhang, X., Shen, W., Tian, Q.: Gaussianobject: High-quality 3d object reconstruction from four views with gaussian splatting. arXiv preprint arXiv:2402.10259 (2024)
126. Yang, J., Pavone, M., Wang, Y.: Freenerf: Improving few-shot neural rendering with free frequency regularization. In: CVPR (2023)
127. Yang, J., Cheng, Z., Duan, Y., Ji, P., Li, H.: Consistnet: Enforcing 3d consistency for multi-view images diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7079–7088 (2024)
128. Yang, X., Chen, Y., Chen, C., Zhang, C., Xu, Y., Yang, X., Liu, F., Lin, G.: Learn to optimize denoising scores: A unified and improved diffusion prior for 3d generation. In: European Conference on Computer Vision. pp. 136–152. Springer (2024)
129. Yang, X., Man, Y., Chen, J.K., Wang, Y.X.: SceneCraft: Layout-guided 3D scene generation. In: Advances in Neural Information Processing Systems (NeurIPS) (2024)
130. Ye, W., Ji, C., Chen, Z., Gao, J., Huang, X., Zhang, S.H., Ouyang, W., He, T., Zhao, C., Zhang, G.: Diffpano: Scalable and consistent text to panorama generation with spherical epipolar-aware diffusion. Advances in Neural Information Processing Systems **37**, 1304–1332 (2024)
131. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023)
132. Yi, T., Fang, J., Wang, J., Wu, G., Xie, L., Zhang, X., Liu, W., Tian, Q., Wang, X.: Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6796–6807 (2024)
133. Yu, H., Li, R., Xie, S., Qiu, J.: Shadow-enlightened image outpainting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7850–7860 (2024)
134. Yu, H., Long, X., Tan, P.: Lm-gaussian: Boost sparse-view 3d gaussian splatting with large model priors. arXiv preprint arXiv:2409.03456 (2024)
135. Yu, H.X., Duan, H., Herrmann, C., Freeman, W.T., Wu, J.: Wonderworld: Interactive 3d scene generation from a single image. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5916–5926 (2025)
136. Yu, R., Liu, J., Zhou, Z., Huang, S.X.: Nerf-enhanced outpainting for faithful field-of-view extrapolation. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 16826–16833. IEEE (2024)
137. Yu, W., Xing, J., Yuan, L., Hu, W., Li, X., Huang, Z., Gao, X., Wong, T.T., Shan, Y., Tian, Y.: Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. arXiv preprint arXiv:2409.02048 (2024)
138. Yu, X., Guo, Y.C., Li, Y., Liang, D., Zhang, S.H., Qi, X.: Text-to-3d with classifier score distillation. arXiv preprint arXiv:2310.19415 (2023)
139. Yu, Z., Megaro-Boldini, M., Sumner, R.W., Djelouah, A.: Unboxed: Geometrically and temporally consistent video outpainting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 7309–7319 (2025)

140. Yuan, X., Tang, S., Li, K., Wang, P.: Camfreediff: Camera-free image to panorama generation with diffusion model. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16408–16417 (2025)
141. Zhang, C., Wu, Q., Gambardella, C.C., Huang, X., Phung, D., Ouyang, W., Cai, J.: Taming stable diffusion for text to 360 panorama image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6347–6357 (2024)
142. Zhang, H., Chen, X., Wang, Y., Liu, X., Wang, Y., Qiao, Y.: 4diffusion: Multi-view video diffusion model for 4d generation. Advances in Neural Information Processing Systems **37**, 15272–15295 (2024)
143. Zhang, J.Y., Lin, A., Kumar, M., Yang, T.H., Ramanan, D., Tulsiani, S.: Cameras as rays: Pose estimation via ray diffusion. arXiv preprint arXiv:2402.14817 (2024)
144. Zhang, J., Li, J., Yu, X., Huang, L., Gu, L., Zheng, J., Bai, X.: Cor-gs: sparse-view 3d gaussian splatting via co-regularization. In: European Conference on Computer Vision. pp. 335–352. Springer (2024)
145. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018)
146. Zhang, S., Huang, J., Zhou, Q., Wang, Z., Wang, F., Luo, J., Yan, J.: Continuous-multiple image outpainting in one-step via positional query and a diffusion-based approach. arXiv preprint arXiv:2401.15652 (2024)
147. Zhang, Z., Wu, B., Wang, X., Luo, Y., Zhang, L., Zhao, Y., Vajda, P., Metaxas, D., Yu, L.: Avid: Any-length video inpainting with diffusion model. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7162–7172 (2024)
148. Zheng, G., Li, T., Jiang, R., Lu, Y., Wu, T., Li, X.: Cami2v: Camera-controlled image-to-video diffusion model. arXiv preprint arXiv:2410.15957 (2024)
149. Zhong, Y., Li, Z., Chen, D.Z., Hong, L., Xu, D.: Taming video diffusion prior with scene-grounding guidance for 3d gaussian splatting from sparse inputs. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6133–6143 (2025)
150. Zhong, Y., Zhou, K., Li, Z., Hong, L., Li, Z., Xu, D.: Empowering sparse-input neural radiance fields with dual-level semantic guidance from dense novel views. arXiv preprint arXiv:2503.02230 (2025)
151. Zhou, J., Gao, H., Voleti, V., Vasishta, A., Yao, C.H., Boss, M., Torr, P., Rupprecht, C., Jampani, V.: Stable virtual camera: Generative view synthesis with diffusion models. arXiv preprint arXiv:2503.14489 (2025)
152. Zhu, Z., Fan, Z., Jiang, Y., Wang, Z.: Fsgs: Real-time few-shot view synthesis using gaussian splatting. In: European conference on computer vision. pp. 145–163. Springer (2024)

A Overview

This supplementary material provides additional details and analyses complementing the main paper. It is organized as follows:

1. **Generation-based Comparison (Sec. B)**. We compare our method with a diffusion-based novel view generation approach, **SEVA** [151], to highlight the differences between generation-based pipelines and our outpainting-based sparse-view reconstruction framework.
2. **Implementation Details (Sec. C)**. We provide detailed implementation information for GaMO, including preprocessing, diffusion inference settings, mask generation, and integration with 3D Gaussian Splatting.
3. **Outpainting Comparison with Multi-View Diffusion Models (Sec. D)**. We evaluate different diffusion backbones for the outpainting stage and analyze their impact on reconstruction quality and geometric consistency.
4. **Iterative Mask Scheduling (Sec. E)**. We present additional details and visualizations of the proposed iterative mask scheduling strategy and its effect on boundary coherence and generation stability.
5. **Additional Quantitative Comparisons (Sec. F)**. We report extended quantitative evaluations across datasets and sparse-view settings (3, 6, and 9 views).
6. **Additional Qualitative Comparisons (Sec. G)**. We provide additional visual comparisons illustrating reconstruction quality and scene completeness.
7. **Runtime Analysis (Sec. H)**. We analyze the computational efficiency of GaMO and compare runtime with existing diffusion-based reconstruction pipelines.
8. **Failure Cases (Sec. I)**. We present representative failure cases and discuss limitations of the proposed approach.
9. **Baseline Implementation Details (Sec. J)**. We describe the implementation details of all baseline methods, including training configurations, initialization strategies, and evaluation protocols.
10. **Per-Scene Quantitative Results (Sec. K)**. We report detailed PSNR, SSIM, and LPIPS results for every scene across Replica [80], ScanNet++ [131], and Mip-NeRF 360 [3].

In addition, we provide an interactive HTML visualization (`main.html`) showing rendered videos along novel-view trajectories across scenes, enabling qualitative inspection of reconstruction quality beyond the input viewpoints.

B Generation-based Comparison

To further compare with approaches designed for direct novel view generation, we evaluate our method against the diffusion-based method SEVA [151]. As shown in Table 6, our approach consistently outperformed SEVA across both datasets and view settings. In particular, our method achieved significantly higher PSNR

and SSIM while reducing LPIPS, indicating more accurate reconstruction and better perceptual quality.

Qualitative results in Fig. 10 further illustrated these differences. While SEVA produced visually plausible results, it often exhibited inaccurate pixel-level details and geometrically inconsistent structures. In contrast, our method preserved accurate details and more coherent scene geometry.

Table 6: Comparison with the diffusion-based novel view generation method SEVA. We report PSNR↑, SSIM↑, and LPIPS↓.

	Replica (3v)			ScanNet++ (3v)		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
SEVA	18.52	0.677	0.193	15.79	0.631	0.343
Ours	**25.40**	**0.864**	**0.117**	**20.01**	**0.765**	**0.266**
	Replica (6v)			**ScanNet++ (6v)**		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
SEVA	18.64	0.691	0.206	16.93	0.640	0.276
Ours	**25.84**	**0.877**	**0.109**	**23.41**	**0.835**	**0.181**



Fig. 10: Qualitative comparison with SEVA. While SEVA produces visually plausible results, it often lacks pixel-level accuracy and geometric consistency.

C Implementation Details

For coarse initialization, we train 3DGS for 10,000 iterations with $\lambda_s = 0.2$ and opacity threshold $\eta_{\text{mask}} = 0.6$.

For outpainting, we use the multi-view diffusion model [4] with focal-length scaling $S_k \in [0.5, 0.7]$, adjusted per scene depending on scene scale. We use DDIM sampling [79] with $T = 50$ steps, and perform latent blending at timesteps $t_1 = 0.7T$, $t_2 = 0.5T$, $t_3 = 0.3T$ with noise resampling $R = 3$. Input and outpainted views share the same resolution (differing only in FOV), with dimensions set as multiples of 64. Before refinement, we alpha-blend downscaled inputs at the center.

For refinement, we optimize 3DGS for 3,000 iterations (3 views) or 7,000 iterations (6/9 views) with $\lambda_{\text{perc}} = 0.1$, alternating supervision between input and outpainted views. In addition, we make minor adjustments to several 3DGS refinement hyperparameters based on scene characteristics to ensure stable optimization. We will release all code, configurations, and scripts used in our experiments. All experiments are conducted on a single NVIDIA RTX 4090 GPU.

D Outpainting Comparison Using Multi-View Diffusion Models

We compare our method against adapted multi-view diffusion models for outpainting. Specifically, we adapt SEVA [151] and MVGenMaster [4] by modifying the camera intrinsics to generate outpainted versions of the input views with



Fig. 11: Comparison of outpainting using adapted multi-view diffusion models. Top: input views. Middle: outpainted views generated by adapted SEVA [151], MVGenMaster [4], and our GaMO. Bottom: novel views after 3DGS refinement using the generated outpainted views. Adapted multi-view diffusion models suffer from multi-view inconsistency, resulting in noisy reconstructions, while our method produces consistent outpainted views that improve reconstruction quality.

extended FOV. These outpainted input views are then used to train 3DGS for improved novel view synthesis.

As shown in Fig. 11, SEVA produces highly noisy novel views after 3DGS refinement due to severe multi-view inconsistency caused by lack of geometric constraints. While MVGenMaster incorporates additional geometric mechanisms (e.g., multi-view conditioning), it still suffers from inconsistency issues that introduce artifacts in the refined reconstruction. In contrast, our GaMO effectively addresses the multi-view inconsistency problem, providing consistent outpainted views across multiple viewpoints that successfully refine 3DGS quality without introducing additional noise or artifacts.

E Iterative Mask Scheduling Implementation

To maximize the utilization of coarse geometry during outpainting while preserving the generative diversity of the diffusion model, we introduce an **Iterative Mask Scheduling (IMS)** strategy. IMS dynamically adjusts the mask region throughout the denoising process, allowing the diffusion model to first freely hallucinate missing regions and later progressively align the generated content with the coarse 3D initialization.

Design Rationale. As demonstrated in the ablation studies in the main paper (Tab. 3), we found that applying mask latent blending at specific denoising steps

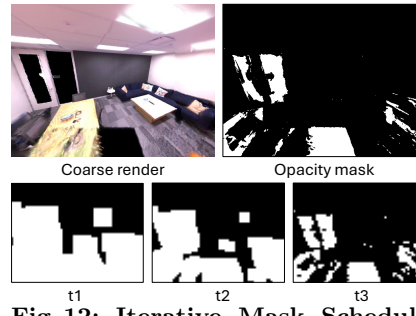


Fig. 12: Iterative Mask Scheduling visualization. Top: coarse render and opacity mask derived from the coarse 3D initialization. Bottom: progressive mask shrinking during denoising at timesteps $t = 35, 25, 15$, with 2, 1, and 0 dilation iterations, respectively.

yields significantly better results than continuous blending throughout the entire denoising process. Based on these findings, we strategically select three representative timesteps corresponding to the early, middle, and late stages of denoising to participate in the latent blending process. At each stage, we employ progressively shrinking mask sizes to control the degree of interference with the denoising process: larger masks in early stages allow more freedom for generation, while smaller masks in later stages enforce stronger alignment with coarse geometry.

Implementation Details. As illustrated in Fig. 12, we generate three mask levels through morphological dilation:

$$\mathcal{M}_{\text{latent}}^{(k)} = \text{Dilate}(\mathcal{M}_{\text{base}}^{\downarrow}, \text{kernel} = 5, \text{iterations} = \frac{k-15}{10}), \tag{10}$$

where $\mathcal{M}_{\text{base}}^{\downarrow}$ denotes the downsampled base mask from the coarse geometry opacity map, aligned to the latent space resolution of 64×48 via adaptive max pooling. The $\text{Dilate}(\cdot)$ operation applies iterative max pooling with 5×5 kernel to expand the masked region. During the denoising process from $t = 50$ to $t = 0$, we apply $\mathcal{M}_{\text{latent}}^{(35)}$ at $t = 35$, $\mathcal{M}_{\text{latent}}^{(25)}$ at $t = 25$, and $\mathcal{M}_{\text{latent}}^{(15)}$ at $t = 15$, as visualized in Fig. 12. This staged approach balances generative freedom with geometric consistency, as validated by our ablation experiments.

F More Quantitative Comparison

We provide additional quantitative results on Replica [80] and ScanNet++ [131] datasets with varying numbers of input views (3, 6, and 9 views), as shown in Tab. 7 and Tab. 8. We focus our comparison on 3DGS [32] and GuidedVD-3DGS [149], a competitive state-of-the-art diffusion-based method.

Evaluation Protocol. For Replica, we follow the evaluation protocol from [149] for all three view settings. For ScanNet++, the 6-view setting follows [149], while the 3-view and 9-view settings use manually selected views to maximize spatial coverage. All methods use DUSt3R [98] for point cloud initialization.

Results. Our method consistently outperforms baselines across most metrics and view settings. On Replica, we achieve the best SSIM and LPIPS scores across all view counts. On ScanNet++, we obtain superior performance across all metrics in all view settings. Notably, our method maintains competitive quality with GuidedVD-3DGS [149] while being significantly faster (approximately 6-9 minutes vs. 3+ hours).

G More Qualitative Comparison

We provide additional qualitative results across 3-, 6-, and 9-view settings on Replica [80] and ScanNet++ [131] datasets, as shown in Fig. 13. We compare against 3DGS [32], FSGS [152], InstantSplat [15], DiFix3D [106], GenFusion [110], and GuidedVD-3DGS [149] using the same baseline configurations as described in the main paper.

Table 7: Quantitative comparison on Replica [80] with 3, 6, and 9 input views.

Method	Replica (3 views)			Replica (6 views)			Replica (9 views)		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
3DGS [32]	20.39	0.818	0.154	24.41	0.862	0.124	26.09	0.890	0.100
GuidedVD-3DGS [149]	25.26	0.864	0.138	26.68	0.880	0.133	28.08	0.901	0.108
Ours	24.40	0.865	0.117	26.40	0.882	0.104	27.58	0.902	0.096

Table 8: Quantitative comparison on ScanNet++ [131] with 3, 6, and 9 input views.

Method	ScanNet++ (3 views)			ScanNet++ (6 views)			ScanNet++ (9 views)		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
3DGS [32]	16.60	0.710	0.313	21.71	0.808	0.186	24.55	0.845	0.155
GuidedVD-3DGS [149]	19.93	0.759	0.297	22.98	0.815	0.204	24.65	0.843	0.159
Ours	20.00	0.765	0.268	23.41	0.835	0.181	25.17	0.860	0.152

As illustrated in Fig 13, even with extremely sparse inputs (3 views), our method produces reasonable content and geometry while maintaining consistency. Compared to baselines, our approach demonstrates better scene coverage with fewer missing regions (black holes), improved geometric consistency with reduced ghosting artifacts, and overall higher visual quality. These improvements are particularly evident in challenging regions highlighted by white boxes.

H Runtime Analysis

We report the end-to-end runtime of our pipeline on a representative indoor scene (Replica_6, *office_2*) with 6 input views at 512 × 384 resolution, evaluated on a single NVIDIA RTX 4090 GPU. The pipeline consists of three main stages: coarse 3DGS reconstruction and rendering, multi-view diffusion outpainting, and the final 3DGS refinement stage that incorporates DUSt3R point cloud initialization and refined 3DGS training.

Table 9: Runtime breakdown of our method on (RTX 4090).

Stage	Time (s)	Time (min)
Coarse 3DGS init. & render	118	1.97
Multi-view outpainting	93	1.55
3DGS refine (train + render)	280	4.67
Total	491	8.18

Table 10: End-to-end runtime comparison (RTX A6000).

Method	Total Time
GuidedVD-3DGS [149]	∼3h 20min
Ours	**9min 54s**

Since GuidedVD-3DGS requires significantly larger GPU memory, it is evaluated on an NVIDIA RTX A6000 GPU. For a fair comparison, we additionally report the runtime of our method on the same hardware. On the Replica *office_2* scene, our pipeline takes **9 min 54 s**, while GuidedVD-3DGS requires approximately **3 hours 20 minutes**. This demonstrates that our method achieves substantially faster reconstruction while maintaining competitive reconstruction quality.

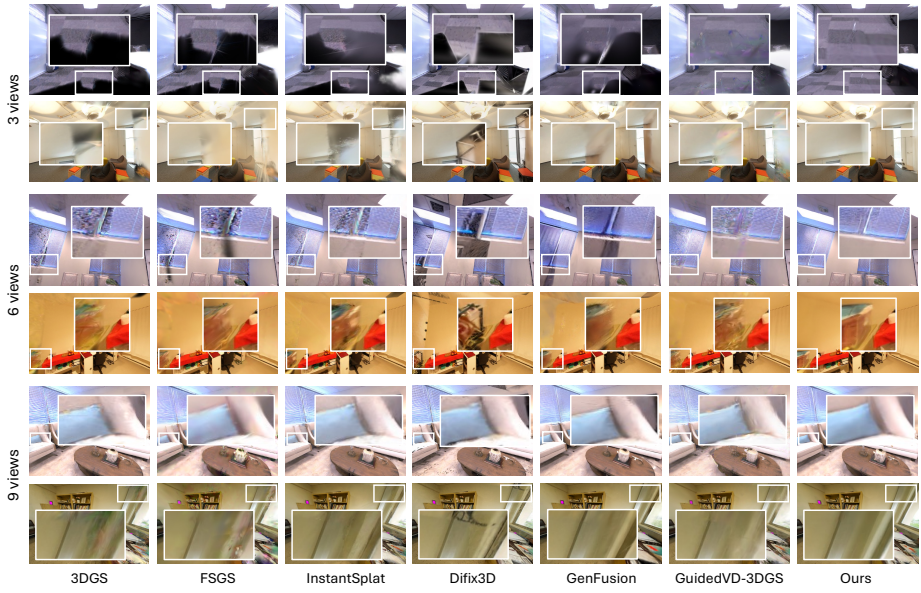


Fig. 13: Qualitative comparison on Replica [80] and ScanNet++ [131] with 3, 6, and 9 sparse views. Our method produces better coverage, geometric consistency, and fewer artifacts compared to baselines. White boxes highlight challenging regions. Best viewed zoomed in.

I Failure Cases

While our method demonstrates strong performance across a wide range of scenarios, it remains limited in scenes containing severe occlusions. This limitation is inherent to all methods that generate novel views for 3D reconstruction; current multi-view diffusion models face the same challenge, as even densely sampled novel viewpoints struggle to reconstruct regions that are severely occluded by obstacles. As shown in Fig. 14, such heavily occluded areas remain challenging for both geometry-aware outpainting and multi-view diffusion methods.

Potential Solutions. A promising direction to address viewpoint-specific occlusions is to generate outpainted views from alternative camera perspectives with geometry-aware mechanisms, such as bird's-eye or top-down views with a larger FOV. By generating content from drastically different viewing angles, these views could potentially observe regions that are occluded from the original camera poses, thereby providing complementary supervision for the occluded areas.

J Baseline Implementation Details

In this section, we describe the implementation details of all baseline methods used in our experiments. Unless otherwise stated, we follow the official implemen-



Fig. 14: Failure cases in heavily occluded regions. Due to severe occlusions in the scene, certain regions are never observed across all input views. Both outpainting (ours) and novel view generation methods struggle to reconstruct these completely unobserved areas. Red boxes highlight the occluded regions where reconstruction fails.

tations and recommended settings provided by the authors. For most methods, the input camera poses are provided and the reconstruction is initialized using point clouds estimated by DUSt3R [98]. InstantSplat follows its original pipeline and uses MASt3R [14] for initialization. Method-specific differences (e.g., initialization strategies) are described in the corresponding subsections.

3DGS [32]. We use the vanilla 3D Gaussian Splatting implementation [32] with its default training settings. The Gaussian primitives are initialized using the DUSt3R point cloud described above, and the input camera poses are kept fixed during optimization.

FSGS [152]. For FSGS, we follow the official implementation and training configuration provided by the authors [152]. In particular, we adopt the version that incorporates the depth supervision term, as suggested in the original implementation. All other training parameters remain consistent with the default configuration.

InstantSplat [15]. InstantSplat is implemented using the official codebase [15] with its default settings. The Gaussian initialization is obtained from point clouds reconstructed using MASt3R [14], following the standard pipeline described in the original work.

Difix3D+ [106]. The Difix3D framework contains multiple variants. In the original implementation, *DiFix3D* and *DiFix3D+* correspond to different variants within the same framework [106], differing in whether a time neural enhancer is used during inference. In our experiments, we adopt the version without the time neural enhancer to maintain consistency with the base reconstruction pipeline.

GenFusion [110]. GenFusion is evaluated using the official implementation and recommended hyperparameters [110]. We follow the standard training and inference pipeline without modifying any of the default settings.

GuidedVD-3DGS [149]. GuidedVD-3DGS is implemented using the official codebase [149]. However, the results reported in the original paper use a different rendering resolution compared to our evaluation protocol. This difference leads to minor numerical discrepancies between the reported numbers in the paper and those reproduced in our environment. For transparency and fair comparison, we report both the original results from the paper and the results reproduced under our experimental settings.

K Per-Scene Quantitative Results

We provide detailed per-scene quantitative results for every method across all datasets and view settings. Tab. 11 reports results on ScanNet++ and Replica with 3 input views, Tab. 12 with 6 input views, and Tab. 13 reports results on Mip-NeRF360 with 6 and 9 input views. Each entry reports PSNR, SSIM, and LPIPS from top to bottom. Our method achieves consistently strong performance across scenes and datasets, ranking first or second in the majority of per-scene metrics while maintaining competitive results throughout, all while being **20× faster** than the strongest competing method.

Table 11: Per-scene quantitative results on ScanNet++ and Replica (3 views). Each entry reports PSNR↑, SSIM↑, and LPIPS↓ from top to bottom.

	ScanNet++ (3 Views)					Replica (3 Views)						
Method	8a20	94ee	7831	a29c	avg	office_2	office_3	office_4	room_0	room_1	room_2	avg
3DGS	15.63 0.640 0.340	16.61 0.747 0.283	19.08 0.724 0.311	14.68 0.673 0.345	16.50 0.696 0.320	18.87 0.853 0.147	22.51 0.870 0.104	20.96 0.852 0.131	20.96 0.852 0.131	22.14 0.829 0.129	19.34 0.757 0.223	20.39 0.818 0.154
FSGS	18.09 0.689 0.299	15.97 0.718 0.355	17.44 0.666 0.412	14.82 0.688 0.369	16.58 0.690 0.359	21.07 0.872 0.177	22.31 0.864 0.117	20.24 0.846 0.140	18.40 0.719 0.213	21.81 0.801 0.167	21.20 0.789 0.220	20.84 0.815 0.172
InstantSplat	17.27 0.733 0.232	17.42 0.754 0.311	17.79 0.694 0.399	14.38 0.698 0.317	16.72 0.720 0.315	19.78 0.862 0.124	22.97 0.879 0.096	22.43 0.872 0.103	18.05 0.729 0.177	22.62 0.822 0.130	18.05 0.726 0.219	20.65 0.815 0.142
Difix3D+	16.05 0.656 0.297	16.05 0.692 0.364	16.38 0.650 0.355	14.09 0.636 0.366	15.64 0.659 0.346	19.58 0.834 0.203	20.77 0.838 0.145	20.65 0.836 0.138	17.89 0.712 0.232	18.79 0.737 0.201	19.14 0.742 0.245	19.47 0.783 0.194
GenFusion	20.58 0.747 0.356	16.63 0.725 0.344	17.46 0.693 0.376	17.22 0.734 0.340	17.97 0.725 0.354	22.43 0.882 0.165	23.53 0.871 0.138	23.46 0.861 0.156	19.40 0.745 0.215	24.18 0.835 0.142	21.02 0.803 0.216	22.34 0.833 0.172
GuidedVD-3DGS	17.32 0.642 0.342	**19.04** 0.756 0.316	19.23 0.716 0.359	19.70 0.764 0.306	18.82 0.720 0.328	25.07 0.905 0.114	**26.80** 0.910 0.097	**25.63** **0.898** 0.109	**24.14** **0.791** 0.171	**26.84** **0.854** 0.126	**23.08** 0.825 0.208	**25.26** 0.864 0.138
Ours	**20.77** **0.753** **0.225**	18.81 **0.770** **0.273**	**20.35** **0.748** **0.299**	**20.31** **0.766** **0.263**	**20.06** **0.759** **0.265**	**25.13** **0.913** **0.088**	25.68 **0.911** **0.075**	24.57 0.897 **0.095**	23.71 0.789 **0.150**	25.17 0.851 **0.114**	22.14 **0.827** **0.179**	24.40 **0.865** **0.117**

Table 12: Per-scene quantitative results on ScanNet++ and Replica (6views). Each entry reports PSNR↑, SSIM↑, and LPIPS↓ from top to bottom.

	ScanNet++ (6 Views)					Replica (6 Views)						
Method	8a20	94ee	7831	a29c	avg	office_2	office_3	office_4	room_0	room_1	room_2	avg
3DGS	24.33 0.868 0.109	20.22 0.819 0.200	21.41 0.765 0.262	20.87 0.821 0.169	21.71 0.818 0.186	27.40 0.922 0.070	25.05 0.889 0.101	23.66 0.866 0.135	21.92 0.794 0.142	24.03 0.846 0.140	24.38 0.854 0.156	24.74 0.862 0.124
FSGS	23.56 0.831 0.234	20.72 0.755 0.402	20.86 0.806 0.309	21.60 0.812 0.246	21.69 0.801 0.298	25.99 0.903 0.092	24.24 0.877 0.114	23.51 0.866 0.148	21.65 0.772 0.171	22.89 0.820 0.171	23.18 0.836 0.175	23.91 0.846 0.145
InstantSplat	22.84 0.844 0.121	20.27 0.815 0.209	21.16 0.761 0.275	20.50 0.822 0.165	21.19 0.811 0.193	26.62 0.913 0.071	20.04 0.850 0.187	22.72 0.864 0.150	21.35 0.782 0.138	23.65 0.835 0.142	24.14 0.850 0.150	23.09 0.849 0.140
Difix3D+	23.86 0.811 0.175	19.42 0.765 0.269	18.86 0.695 0.312	20.34 0.784 0.219	20.62 0.764 0.244	23.40 0.862 0.160	21.70 0.845 0.149	22.79 0.831 0.176	20.42 0.747 0.204	21.22 0.776 0.219	21.62 0.800 0.218	21.86 0.810 0.188
GenFusion	24.69 0.860 0.127	21.01 0.818 0.211	20.21 0.750 0.312	21.91 0.805 0.221	21.96 0.808 0.218	25.82 0.909 0.085	23.42 0.869 0.127	24.91 0.878 0.139	23.57 0.808 0.149	23.04 0.830 0.165	23.14 0.837 0.187	23.98 0.855 0.142
GuidedVD-3DGS†	25.10 0.882 0.118	23.10 0.860 0.201	22.16 0.803 0.269	25.21 0.857 0.157	23.89 0.850 0.182	27.46 0.916 0.083	26.81 0.902 0.099	27.43 0.897 0.122	24.85 0.796 0.145	26.00 0.851 0.142	25.53 0.872 0.142	26.35 0.872 0.122
GuidedVD-3DGS‡	24.18 0.844 0.143	22.11 0.824 0.219	**21.84** 0.768 0.283	23.79 0.824 0.172	22.98 0.815 0.204	**28.04** 0.925 0.087	**26.46** **0.901** 0.112	**27.63** 0.900 0.130	25.55 0.823 0.143	**26.18** **0.858** 0.157	**26.24** 0.871 0.167	**26.68** 0.880 0.133
Ours	**24.98** **0.877** **0.104**	**22.38** **0.840** **0.186**	21.56 **0.783** **0.269**	**24.70** **0.841** **0.164**	**23.41** **0.835** **0.181**	28.18 **0.927** **0.062**	26.23 0.900 **0.087**	26.85 **0.900** **0.104**	**25.65** **0.831** **0.117**	25.47 0.857 **0.127**	26.00 **0.878** **0.128**	26.40 **0.882** **0.104**

† Reported in paper, ‡ Our reproduction.

Table 13: Per-scene quantitative results on Mip-NeRF360 with 6 and 9 input views. Each entry reports PSNR↑, SSIM↑, and LPIPS↓ from top to bottom.

	Mip-NeRF360 (6 Views)										Mip-NeRF360 (9 Views)									
Method	bicycle	bonsai	counter	flowers	garden	kitchen	room	stump	treehill	avg	bicycle	bonsai	counter	flowers	garden	kitchen	room	stump	treehill	avg
3DGS	15.53 0.239 0.510	13.69 0.338 0.551	15.22 0.420 0.415	13.14 0.163 0.539	17.02 0.403 0.312	17.51 0.543 0.290	14.99 0.479 0.448	16.07 0.188 0.511	13.69 0.308 0.558	15.21 0.342 0.459	14.69 0.263 0.490	14.89 0.406 0.461	16.74 0.504 0.344	13.95 0.207 0.472	18.70 0.477 **0.241**	18.30 0.605 0.274	17.25 0.585 0.335	17.75 0.273 0.420	13.76 0.344 0.515	16.23 0.407 0.395
FSGS	14.18 0.193 0.604	13.39 0.322 0.588	15.02 0.409 0.452	12.49 0.161 0.653	17.17 0.397 0.354	17.14 0.520 0.355	14.58 0.472 0.487	15.75 0.183 0.556	13.16 0.278 0.736	14.76 0.326 0.532	14.14 0.251 0.603	14.36 0.387 0.508	16.18 0.486 0.387	13.60 0.190 0.636	17.75 0.419 0.375	15.02 0.495 0.448	15.99 0.554 0.393	17.37 0.266 0.463	13.84 0.307 0.703	15.36 0.373 0.502
InstantSplat	16.93 **0.334** **0.457**	13.89 0.369 0.550	16.14 0.463 0.410	**13.65** 0.186 0.546	17.37 0.405 0.296	18.26 **0.606** **0.259**	15.51 0.511 0.432	**18.11** **0.277** 0.459	13.33 0.342 0.577	15.91 0.388 0.443	12.86 0.291 0.617	14.84 0.424 0.482	17.30 0.526 0.362	13.38 0.198 0.571	18.41 0.446 0.257	**19.82** **0.666** **0.211**	17.79 0.601 0.331	**19.19** **0.358** 0.405	13.29 0.385 0.532	16.32 0.433 0.419
Difix3D+	15.04 0.187 0.468	13.47 0.323 **0.494**	14.96 0.397 **0.380**	12.95 0.141 **0.493**	17.13 0.374 **0.291**	17.37 0.500 0.281	14.62 0.450 0.431	15.39 0.141 **0.445**	13.57 0.263 **0.491**	14.94 0.309 **0.419**	15.38 0.219 **0.451**	14.55 0.387 **0.427**	16.65 0.484 **0.323**	13.83 0.182 **0.434**	18.52 0.437 0.244	17.98 0.546 0.259	16.51 0.536 0.355	16.71 0.215 **0.395**	14.25 0.308 **0.451**	16.04 0.368 **0.371**
GenFusion	16.29 0.292 0.570	14.43 0.389 0.534	**16.54** 0.474 0.420	13.90 **0.200** 0.595	18.23 0.413 0.379	18.09 0.521 0.352	17.24 0.579 0.407	17.68 0.226 0.604	**15.18** **0.363** 0.613	16.40 0.384 0.487	**16.53** 0.309 0.537	**15.29** **0.429** 0.448	**17.76** **0.547** 0.364	**14.91** **0.234** 0.496	19.38 0.475 0.302	19.88 0.586 0.254	19.03 0.651 0.307	18.53 0.290 0.486	**16.62** **0.393** 0.488	17.55 0.435 0.409
GuidedVD-3DGS	14.80 0.189 0.670	12.75 0.293 0.722	12.05 0.305 0.714	12.48 0.135 0.717	15.20 0.210 0.541	14.32 0.359 0.658	13.57 0.462 0.566	15.76 0.173 0.602	14.12 0.333 0.569	13.89 0.273 0.640	14.49 0.238 0.519	14.56 0.383 0.483	16.81 0.506 0.346	13.49 0.194 0.492	18.52 0.456 0.253	15.81 0.542 0.373	17.57 0.598 0.327	17.27 0.246 0.432	13.44 0.313 0.538	15.77 0.386 0.418
Ours	**17.33** 0.285 0.476	**14.74** **0.392** 0.524	16.45 **0.481** 0.398	13.55 0.192 0.547	**18.84** **0.430** 0.294	**18.68** 0.567 0.304	**17.93** **0.615** **0.370**	17.72 0.231 0.488	15.39 0.341 0.522	**16.74** **0.393** 0.436	16.21 **0.314** **0.450**	16.15 0.461 0.444	17.34 0.539 0.365	14.98 0.229 0.463	**19.89** **0.492** 0.244	19.38 0.626 0.266	**20.55** **0.700** **0.282**	18.66 0.289 0.423	14.87 0.384 0.488	**17.56** **0.448** 0.381