

IMFine: 3D Inpainting via Geometry-guided Multi-view Refinement

Zhihao Shi^{*1}

Dong Huo^{*o2}

Yuhongze Zhou¹

Yan Min^{o3}

Juwei Lu¹

Xinxin Zuo⁴

¹Huawei Canada Research Institute

²University of Alberta

³McMaster University

⁴Concordia University

zhihaoshi2022@gmail.com, dhuo@ualberta.ca, yuhongze.zhou@mail.mcgill.com

miny13@mcmaster.ca, juwei.lu@huawei.com, xinxin.zuo@concordia.ca



Figure 1. We present a novel 3D inpainting method capable of removing target objects from the 3D scenes and seamlessly inpainting the occluded regions across various scenarios. The proposed method ensures the generation of plausible geometry and consistent textures. *It is highly recommended to see video results in our [project page](#).*

Abstract

Current 3D inpainting and object removal methods are largely limited to front-facing scenes, facing substantial challenges when applied to diverse, “unconstrained” scenes where the camera orientation and trajectory are unrestricted. To bridge this gap, we introduce a novel approach that produces inpainted 3D scenes with consistent visual quality and coherent underlying geometry across both front-facing and unconstrained scenes. Specifically, we propose a robust 3D inpainting pipeline that incorporates geometric priors and a multi-view refinement network trained via test-time adaptation, building on a pre-trained image inpainting model. Additionally, we develop a novel inpainting mask detection technique to derive targeted inpainting masks from object masks, boosting the performance in handling unconstrained scenes. To validate the efficacy of our approach, we create a challenging

and diverse benchmark that spans a wide range of scenes. Comprehensive experiments demonstrate that our proposed method substantially outperforms existing state-of-the-art approaches.

1. Introduction

3D editing is an active research area with significant potential for various applications, including video games, the Metaverse, virtual reality, and augmented reality. For example, after a 3D scene is created or reconstructed, a user might want to remove certain objects from the scene. This paper mainly focuses on the challenging task of inpainting in 3D scenes to fill the occluded regions to enable seamless object removal.

Compared to image inpainting [22, 28], 3D inpainting models lag significantly behind due to the lack of large-scale 3D datasets and efficient 3D data structures. Therefore, existing 3D inpainting methods mainly focus on lifting 2D inpainting priors to 3D. Although the current diffusion-

* denotes equal contribution.

o Work done during an internship at Huawei Canada Research Institute

based image inpainting models can generate images with vivid textures and high-fidelity details, they are trained on single images without considering any 3D consistency. It poses great challenges in maintaining view consistency and preserving 3D coherence when lifting those 2D priors to 3D.

To address the inconsistency issue, there are approaches that implicitly lift the 2D models into 3D via score distillation sampling (SDS) loss [3, 18], which was proposed initially in text-to-3D generation [20], to harness the inherent coherence of 2D diffusion models under the same prompt, complemented by the iterative updating scheme during optimization to ensure view consistency. However, they are constrained by the inherent limitation of SDS loss, which results in over-saturation and over-smoothing in the optimized images, even with different variants of SDS losses. Alternatively, explicit methods [2, 5, 16, 17, 30] first generate one or few images with the image inpainting models and try to preserve the 3D consistency by enforcing geometry constraints or minimizing perceptual losses between rendered images and the inpainted 2D images. Although they have demonstrated good performance, especially for front-facing scenes with very limited view changes, the inpainted 3D scenes have artifacts such as floaters and blurry textures in unconstrained scenarios with complex geometry and large view variation. The reasons are that first, 3D geometry is rather difficult to get perfectly restored; second, perceptual loss can only alleviate the inconsistency by minimizing overall similarity. Inconsistency still exists between those inpainted images, leading to blurry textures for the inpainted 3D scenes. More recently, [2] proposed to train a multi-view inpainting network to generate consistent multi-view inpainted images. However, due to the limited diversity of current multi-view datasets, they cannot generalize to unconstrained 3D scenes (e.g., 360° cases).

To enhance the ability to generalize to various scenes for 3D inpainting, we utilize geometry priors and develop a multi-view refinement network fine-tuned at test time to produce consistent images in a coarse-to-fine manner. Specifically, we start with a single reference view inpainted with image inpainting models, followed by coarse geometry restoration and a warping operation to warp the inpainted content from the reference view to others. However, due to the imperfect reconstructed geometry and view-dependent effects, the warped images tend to have artifacts, such as distorted textures and mismatches between the warped and neighboring contents. To refine the warped contents, we present a multi-view refinement model to generate consistent images with high fidelity, taking the warped images as a condition. The key idea is that instead of training a model with extensive datasets, we take the pre-trained image inpainting model, add space-time attention layers, and fine-tune them with the current test scenes. We take the

originally captured multi-view images and implement various augmentations to synthesize paired data to fine-tune the multi-view refinement model.

Furthermore, previous works usually take the entire object masks as inpainted regions. However, some regions inside the object masks might have already been exposed in other views, especially for 3D object removal and inpainting in 360° scenes (see Fig. 4). Therefore, we believe it is critical to differentiate between object masks and inpainting masks, fully exploiting the available information of the reconstructed 3D scenes. To this end, a novel inpainting mask detection approach is proposed to extract masks accurately and exclusively for real occluded regions.

Finally, considering that the existing benchmarks mostly cover front-facing scenes or planar surfaces, we introduce a new dataset encompassing 20 diverse scenes, from indoor to outdoor environments, covering large view variations and complex interfacial surfaces, to better evaluate and advance 3D inpainting methods.

In summary, our contributions are as follows:

1. A multi-view consistent inpainting pipeline is proposed. We leverage geometric priors and a multi-view refinement model that is fine-tuned from image inpainting models via test-time adaptation to generate consistent multi-view images.
2. We further develop an inpainting mask detection technique that accurately distinguishes occluded regions, enabling robust processing of complex scenes, such as 360° environments.
3. The proposed inpainting method has achieved state-of-the-art performance across various benchmarks, particularly excelling in challenging, unconstrained scenarios.
4. We contribute a new and challenging 3D scene inpainting benchmark, offering diverse scenes to facilitate broader evaluation and advancement in 3D inpainting research.

2. Related Work

2.1. 3D Inpainting

Existing 3D inpainting methods [3, 5, 16–18, 30, 32] are implemented by lifting 2D inpainting priors to 3D with various strategies designed to ensure 3D consistency. One direction, inspired by score distillation-based approaches [20] in text-to-3D generation, formulates 3D inpainting as a continuous distillation process [3, 18]. While these approaches achieve reasonably consistent inpainting results, they face inherent limitations of score distillation, such as producing over-saturated or overly smooth outputs. Another direction explicitly enforces 3D consistency by independently inpainting one or multiple 2D views and then consolidating multi-view inconsistencies in 3D. Reference-guided NeRF [16] addresses this challenge by using a single view to

guide geometry inpainting to avoid inconsistencies. However, it struggles with accurate geometry capture in complex 360° scenes using just a single view. SPINeRF [17], and InNeRF360 [30] utilize multiple inpainted images to guide 3D inpainting and incorporate perceptual loss to reduce inconsistencies resulting from independently inpainted images. Nevertheless, while perceptual loss can maintain overall consistency, it performs poorly on finer details, often leading to blurry 3D results. Our method lies in the latter direction. It addresses these issues through a geometry-guided refinement network that generates a set of multi-view consistent images, reconstructing clearer and consistent 3D inpainted scenes.

2.2. Video and Multi-View Diffusion

The diffusion model for text-to-image (T2I) generation [7, 8, 22, 27] has achieved significant success in recent years, known for its ability to produce high-quality, diverse, and controllable outputs. Building on this, extensive studies have extended image diffusion models to the video generation domain [9, 10, 26, 40] by incorporating temporal layers within each transformer block to perform space-time attention, which captures correlations across frames and shows promising results in generating realistic videos.

Inspired by these advances in video generation, researchers have begun exploring the adaptation of T2I models into multi-view diffusion models for implicit 3D reasoning [15, 24, 25, 31]. For example, methods such as MVDream [25] and Zero123++ [24] leverage this idea and derive multi-view diffusion models, capable of generating consistent multiple views simultaneously for 3D generation. However, the full space-time attention mechanism used in these methods can lead to prohibitive memory and computational costs, making it impractical for handling large numbers of views. To address this, recent work [35] introduces a sparse space-time attention mechanism, which reduces memory requirements by attending only to the initial and previous frames. However, the sparse attention approach assumes input frames are from a continuous video sequence and remains under-explored in sparse multi-view scenarios.

2.3. Diffusion Model Fine-tuning

Adapting large pre-trained diffusion models has proven to be an effective and efficient approach for completing user-specified tasks, such as synthesizing specific subjects in new contexts [4, 23] and inpainting images with instance matching reference images [29, 36]. Adaptation methods include injecting extra layers [19, 36, 38, 39], searching appropriate unique prompt [23], updating partial or full model weights [14, 18, 29], or combinations of these techniques. Similarly, in this paper, instead of relying on extensive multi-view datasets, we propose adapting the pre-trained image

diffusion model into a multi-view refinement model for 3D inpainting purposes.

3. Preliminary

Gaussian splatting (GS) methods [11, 12] represent a scene as a collection of multivariate Gaussian primitives: $\mathcal{G} = \{(\mu_m, \Sigma_m, \Gamma_m, \text{SH}_m)\}_{m=1}^M$, where μ_m , Σ_m , Γ_m and SH_m denote the mean vector, covariance matrix, opacity, and spherical harmonic coefficients of a multivariate Gaussian, respectively. View synthesis is performed by standard point-based α' -rendering, where a single pixel color is computed by sorting \mathcal{G} (or their 2D projection) in their depth order and alpha-blending each contributed primitive:

$$C = \sum_m c_m \alpha_m \prod_{j=1}^{m-1} (1 - \alpha_j), \quad (1)$$

where C is the output pixel color, c_m is the pre-computed color of the m -th primitive, and α_m is a calculated opacity based on distance from the Gaussian center to the interaction point. Due to the excellent performance in geometry reconstruction, in this work, we choose 2D-GS [11] as the 3D representation.

4. Method

Given a GS scene \mathcal{G} , along with corresponding training images $\{I_n\}_{n=1}^N$, associated camera poses $\{\Pi_n\}_{n=1}^N$, and a set of object masks $\{M_n\}_{n=1}^N$ identifying objects to be removed, our goal is to generate an inpainted scene \mathcal{G}^* where the target objects are removed and occluded regions are filled seamlessly with the background. The entire pipeline contains two stages, namely object removal and 3D scene inpainting. We first remove object primitives (Sec. 4.1), exposing never-before-seen (NBS) regions which are the areas that truly need inpainting. We then either use the object masks as inpainting masks when they are identical or detect specific inpainting masks (Sec. 4.3), ensuring that the inpainting mask exclusively covers the NBS regions in each view, denoted as $\{P_n\}_{n=1}^N$. Given the pruned GS scene and inpainting masks, we propose a novel 3D inpainting pipeline to inpaint \mathcal{G} as \mathcal{G}^* (Sec. 4.2).

4.1. Object Removal

To remove the objects from the reconstructed GS scene, we append an additional learnable parameter $\{p_m\}_{m=1}^M \in [0, 1]$ to each Gaussian primitive, indicating whether it belongs to the object to be removed, which is initialized as 0.5. To compute the value of p_m for each Gaussian, we map the object masks back to the Gaussian primitives. Specifically, we render a set of object masks $\{\hat{M}\}_{n=1}^N$ using α' rendering (similar to color rendering) at $\{\Pi_n\}_{n=1}^N$ but replacing c_m with p_m in Eq. 1, followed by K_1 ($K_1 = 1000$) steps of the

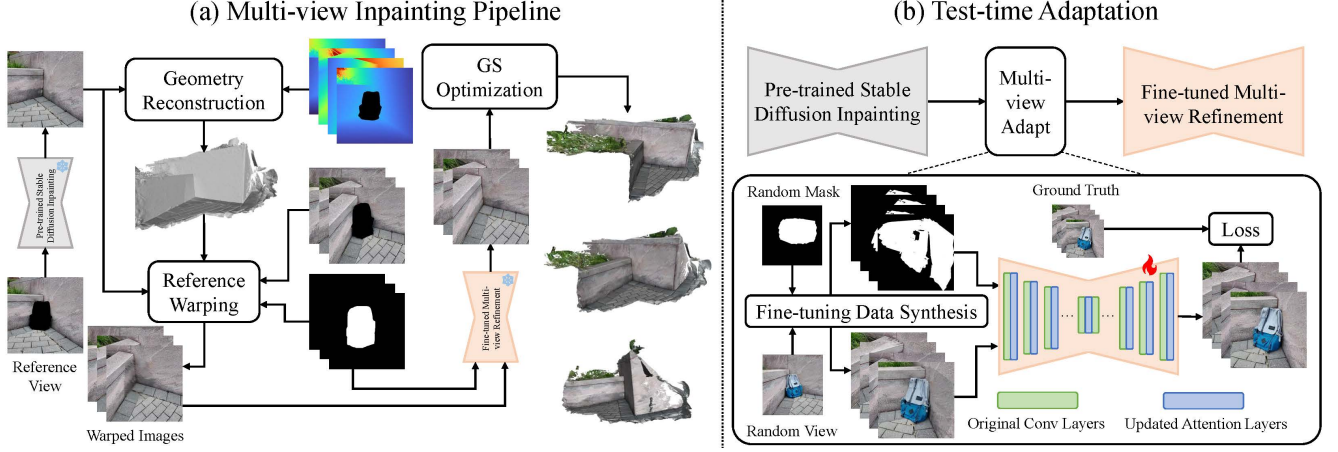


Figure 2. **Overview of our proposed approach.** (a) shows the proposed 3D inpainting pipeline. Specifically, a reference view is first inpainted, which will be used to reconstruct complete geometry, along with rendered multi-view depth maps. The inpainted reference image is then warped to other views, which are taken as input conditions for the multi-view refinement network to generate a set of view-consistent images. Afterward, we fine-tune the 3D scene to get the inpainted scene. (b) illustrates the detailed structure and training of the multi-view refinement network. Basically, we take the pre-trained image inpainting model, incorporate additional space-time attention layers, and exploit the test-time adaptation to fine-tune it into a multi-view refinement model. To generate a fine-tuning dataset, we synthesize warped images under the original captured scene, which has the ground-truth captured images. In detail, reference views are selected randomly, and we set random masks over the images, apply geometry jittering, and warp the reference images in other views.

optimization to update p_m with respect to \mathcal{L}_1 loss between $\{\hat{M}\}_{n=1}^N$ and $\{M\}_{n=1}^N$. Notably, other parameters remain unchanged during this procedure. Finally, a pruning operation is performed to produce the pruned scene $\mathcal{G}_{\text{pruned}}$ by thresholding p_m with a threshold τ ($\tau = 0.4$).

4.2. Proposed 3D Inpainting Pipeline

Given the pruned GS scene $\mathcal{G}_{\text{pruned}}$ as well as inpainting masks $\{P_n\}_{n=1}^N$ indicating the regions to be inpainted for each view, we propose a novel 3D inpainting pipeline shown in Fig 2. We first render color images $\{I'_n\}_{n=1}^N$ from $\mathcal{G}_{\text{pruned}}$ for each view and select a reference view with image $I'_r \in \{I'_n\}_{n=1}^N$, which is then inpainted using a pre-trained inpainting model [22] to reference image I_r^* . We conduct geometry reconstruction to fill out the hole in 3D, which will be utilized to warp the inpainted contents in the reference image into the remaining views. We employ a multi-view refinement model to further refine the warped images, ensuring high fidelity and consistency, and then update the 3D scene. The model is fine-tuned from the pre-trained image inpainting model via a test-time adaptation strategy. The geometry reconstruction and multi-view refinement model are described as follows.

4.2.1 Geometry Reconstruction

Following [11], we render a set of depth maps $\{D_n\}_{n=1}^N$ from the pruned GS scene $\mathcal{G}_{\text{pruned}}$ and adopt a truncated signed distance function (TSDF) [6] technique to reconstruct the 3D mesh. However, $\{D_n\}_{n=1}^N$ only contributes

to the reconstruction of the background region, as the depth values within their inpainting mask areas are invalid. To fill the masked region, we aim to generate an all-valid depth map D_r^* of the reference view by filling the masked areas in D_r based on information from I_r^* . The filled depth map should meet two key requirements: 1) it must be consistent with the reference image I_r^* ; 2) it should seamlessly connect with the depth values outside the masked region.

We begin by exploiting a monocular depth estimator [37] to estimate depth \bar{D}_r from I_r^* , followed by a linear transformation to convert \bar{D}_r to scene scale \tilde{D}_r : $\tilde{D}_r = a * \bar{D}_r + b$, where the parameters a, b are computed in the neighborhood of P_r to minimize the difference between \bar{D}_r and D_r in advance, following [16]. We then develop a filling scheme according to \tilde{D}_r . Formally, the filled values of D_r^* are computed as follows:

$$D_r^*(x) = \arg \min_{D_r^*(x)} \sum_{x \in P_r} \|\nabla D_r^*(x) - \nabla \tilde{D}_r(x)\|^2 + \lambda \nabla D_r^*(x) \quad (2)$$

$$\text{s.t. } D_r^*(x) = D_r(x), \text{ where } P_r(x) = 0,$$

where ∇ denotes the Laplacian calculation and x represents the pixel coordinate. Intuitively, the first term encourages D_r^* to maintain a similar changing trend as \tilde{D}_r , while the second term regularizes D_r^* to be smooth and consistent with the depths outside the mask. As such, D_r^* can be used for the reconstruction of the NBS region.

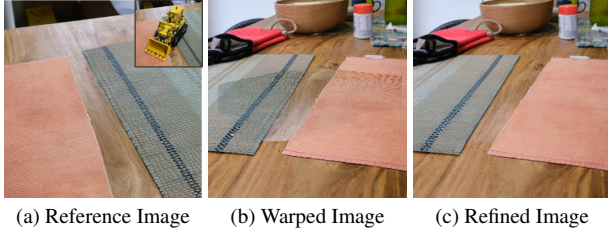


Figure 3. **Illustration of the importance of multi-view refinement.** The multi-view refinement model is proposed to tackle the warping artifacts with the example shown in (b), generating high-fidelity multi-view consistent images with one sampled view shown in (c).

4.2.2 Test-Time Adaptation for Multi-view Refinement

We warp the inpainted contents from the reference view I_r to other views with the above restored geometry. Due to the imperfect geometry and the neglect of view-dependent effects, those warped contents contain artifacts, such as texture distortion and inconsistent appearance with the neighboring regions, especially when there are large view angle changes with respect to the reference view as shown in Fig 3. On the other hand, these warped contents have coarse-level consistency across views. To this end, we propose a multi-view refinement model that produces high-fidelity, consistent images conditioning on the warped images.

Network: Our model is built upon a pre-trained diffusion-based image inpainting model [22], which takes a masked image and its associated mask as the input, corresponding to the warped image and inpainting mask in our scenarios, respectively. To ensure multi-view consistency, we replace the self-attention layer in each transformer block with space-time attention to explore correlations across multiple views. Considering the quadratic complexity of global attention and the large number of warped images, space-time attention is implemented as the sparse space-time attention [10] that only attends to neighboring images. The attention layer is also allowed to attend to the reference image for alignment.

Training: Instead of relying on large datasets and extensive training, we exploit the test-time adaptation strategy for each scene to train the multi-view refinement network. Paired training sets are created for model adaptation. Specifically, given the original images $\{I_n\}_{n=1}^N$, at each training step, $L + 1$ images are randomly selected as ground truth. To simulate realistic warping scenarios, we randomly choose one image as the reference, generate a random mask, and warp both the reference image and mask to other views with geometry guidance from \mathcal{G} . In addition, to encourage cross-view learning as in [2], we also synthesize another set of paired images by applying image augmentations independently on each captured image, i.e., irregularly shaped masks that are randomly generated on

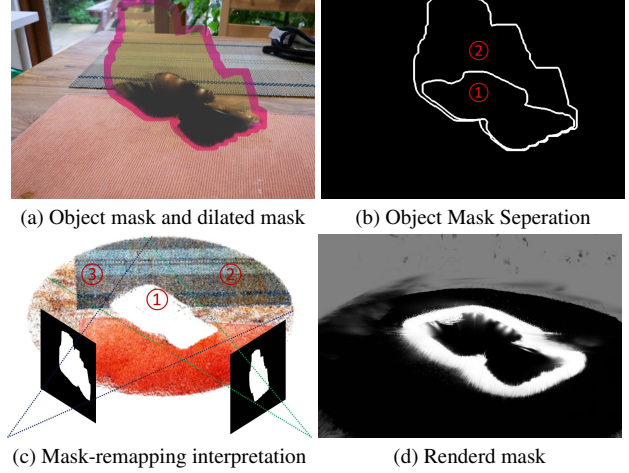


Figure 4. **Inpainting mask detection.** (a) shows the object mask (orange) along with the dilated mask (pink), which includes additional neighboring pixels; Each dilated mask consists of two parts: the actual inpainting mask and out-of-interest areas, marked as regions ① and ② in (b). These regions are mapped in 3D to the NBS and background areas, respectively, as shown in regions ① and ② in (c). During multi-view mapping optimization, the central NBS region is progressively refined, while background areas are suppressed by each other; (d) shows the rendered neighbor mask after GS scene optimization.

each image. Image-based augmentations—such as elastic transformations and color jittering, are then applied to the masked regions to simulate warping artifacts. We train the multi-view refinement model with those paired images, employing a simplified variational bound objective [8].

Inference: At inference, similar to [33], we randomly shuffle the order of the L warped images at each denoising step to break the sequential constraint of sparse attention, further enhancing cross-view consistency.

4.2.3 Fine-tune GS Scene

With the generated multi-view consistent images, we fine-tune $\mathcal{G}_{\text{pruned}}$ for $K_2(K_2 = 7000)$ steps with an L_1 loss and a SSIM loss as straightforward pixel-wise supervision. It is worth noting that other robust losses (e.g., perceptual loss) are functionally orthogonal to our approach and can be used to improve it further.

4.3. Inpainting Mask Detection

In the overall pipeline of 3D object removal and scene inpainting, identifying the occluded or NBS regions for inpainting is also critical, especially for 360° cases. In those cases, the object mask often covers a much larger area than the regions that require inpainting (as illustrated in Fig. 4). Therefore, a novel inpainting mask detection approach is proposed to identify the NBS region for each view.

An intuitive idea is to detect the NBS region in the

pruned GS scene and project it back into each view to obtain inpainting masks. However, since, by definition, the NBS region is occluded by the object, it usually lacks good geometry and meaningful Gaussians, making it challenging to detect in 3D. Here we propose to detect the inpainting masks by leveraging neighbors of NBS region. In detail, we first dilate the object masks to include neighboring pixels (Fig. 4(a)), and then map the dilated masks onto the pruned scene $\mathcal{G}_{\text{pruned}}$. Mapping is implemented as updating the learnable parameter $\{p_m\}_{m=1}^M$ of each Gaussian with respect to \mathcal{L}_1 loss between rendered masks of $\mathcal{G}_{\text{pruned}}$ and the dilated masks. The optimization process is the same as object removal (Sec. 4.1) but with the object mask replaced with dilated masks, and the optimization is performed over the pruned scene. We illustrate the interpretation of this process in Fig. 4(b-c). After the mapping process, we are able to get the neighbor mask by rendering $\{p_m\}_{m=1}^M$ from $\mathcal{G}_{\text{pruned}}$ as shown in Fig. 4(d). Finally, we extract the bounding box for the above rendered mask, which will be taken as input prompt for segment anything model [13] to get the final inpainting mask.

5. Experiments

Datasets: Most existing publicly available 3D inpainting datasets [17, 34] are limited to front-facing cases, which cannot be used to evaluate methods in more diverse or realistic situations. To address this, we construct a dataset that contains 20 diverse cases, ranging from indoor to outdoor, covering various camera trajectories-including front-facing, 90°, 180°, and 360°, and containing various interfacial surfaces such as single plane, multiple planes, curvature surfaces, and irregular surfaces. Each scene is recorded using a DJI Pocket 3 at 60fps with a resolution of 1920×1080 , from which we evenly extract 175 images and generate the corresponding camera poses and object masks. The first 50 images are used for evaluation purpose with the object removed, while the remaining 125 images are object-present and are used to reconstruct the original GS scene. Previews of our collected dataset are shown in supplementary materials. We use half resolution in the following experiments. In addition, we evaluate our method on SPINeRF dataset [17] to compare with state-of-the-art.

Implementation Details: StableDiffusionInpainting-Pipeline is utilized for image inpainting and our multi-view refinement model initialization. During training, we randomly select $L = 5$ images from training images and crop 512×512 patches to fine-tune the model over 1000 iterations with the learning rate and batch size set as 3×10^{-5} and 1, individually. At inference, directly refining all warped images, i.e., $L = 125$ would result in prohibitive memory usage and computational costs. To

Table 1. **Quantitative evaluation on the SPINeRF dataset and our dataset.** Numbers in bold indicate the best performance, and underscored numbers indicate the second best.

Method	Ours			SPINeRF		
	PSNR (\uparrow)	LPIPS (\downarrow)	FID (\downarrow)	PSNR (\uparrow)	LPIPS (\downarrow)	FID (\downarrow)
GaussianEditor	15.71	0.6163	375.03	14.41	0.6247	343.16
GScream	17.18	0.4431	290.63	16.96	0.3931	154.71
SPINeRF	<u>18.75</u>	<u>0.3519</u>	<u>206.43</u>	17.47	0.5740	239.99
MVIPNeRF	18.63	0.4332	278.99	17.67	0.5070	255.51
Ours	19.67	0.2685	149.52	<u>17.58</u>	<u>0.4513</u>	154.34

address this, we cluster all views into L groups and select the center view of each group as the anchor. Multi-view refinement is applied only to these L anchor views, while the remaining views in each group are warped from their respective anchors, as artifacts caused by warping between closely aligned views are negligible. L is set to 12 for 360° cases and linearly scaled down for other scenarios, such as 4 for 90° cases. We directly use object masks when evaluating on front-facing scenes and apply inpainting mask detection on others. All experiments are conducted on a single NVIDIA V100 GPU, with the inpainting of a 3D scene taking approximately one hour.

5.1. Evaluation against State-of-the-Art Methods

We evaluate our method against the state-of-the-art methods on our dataset and SPINeRF dataset. The following state-of-the-art methods are reproduced on our datasets, including GaussianEditor [5], GScream [32], MVIPNeRF [3] and SPINeRF [17] using their official implementations. In particular, since the naive NeRF models used in MVIPNeRF and SPINeRF cannot handle complex unbounded scenes, we replace them with MeRF [21] considering both memory usage and reconstruction performance. On our dataset, we utilize the evaluation protocol and official implementation from [3] to compare methods. Specifically, we render color images at corresponding test views from the inpainted scenes and evaluate them against the ground truth using three metrics, namely PSNR, LPIPS, and FID. These metrics are calculated only within the bounding box defined by the object mask. For the SPINeRF dataset, we copy values from [3] for SPINeRF and MVIPNeRF, and we reproduce GaussianEditor and GScream to align with the metric computation protocol in [3].

As can be seen from Tab. 1, the proposed method consistently achieves the best performance on our dataset, with a particularly large margin that demonstrates its superiority in unconstrained realistic scenarios. On the SPINeRF dataset, our method achieves state-of-the-art results comparable to other approaches, given that this dataset is limited to front-facing cases and challenging to distinguish methods in handling complex cases.

In addition, we provide qualitative comparisons in Fig. 5, where our method successfully removes objects and seamlessly inpaints with appropriate content. Moreover, the in-



Figure 5. **Qualitative evaluation.** We show the visual comparisons against the state-of-the-art 3D inpainting algorithms, with two views displayed at the top and bottom for each method. Our method generates consistent, higher-quality results with rich texture details and fewer artifacts.

painted scene maintains 3D consistency, yielding visually more pleasing outcomes and a coherent geometric structure. In contrast, other methods often produce severe artifacts and fail to inpaint objects effectively in complex, realistic scenarios. We highly recommend viewing our video visualizations for a more comprehensive comparison.

5.2. Detection of Inpainting Masks

We then demonstrate the importance of doing inpainting mask detection and differentiating them from object masks. For our collected dataset, we manually annotate inpainting masks from rendered images of pruned scenes and compare them with the original object masks. Since our dataset is derived from continuous capture videos, we annotate one out of every 10 images in each scene, resulting in a total of 260 annotated masks. Our analysis shows that the inpainting masks, on average, occupy only 50.78% of the area covered by the object masks. This finding highlights that adopting entire object masks requires inpainting nearly twice the necessary area, complicating the task and leading to suboptimal results.

We also compare our detected inpainting masks with the manually labeled masks using the mean Intersection over Union (mIoU) metric. Our method achieves an 81.12% IoU score, validating the accuracy and robustness of the pro-

Table 2. **Quantitative results of ablation studies.**

Method	PSNR (\uparrow)	LPIPS (\downarrow)	FID (\downarrow)
w/o Warping	17.85	0.3215	198.24
w/o Refinement	18.90	0.3069	206.96
General Refinement	19.08	0.2719	165.80
Single-View Refinement	19.46	0.2725	154.33
Multi-View Refinement (Ours)	19.67	0.2685	149.52

posed inpainting mask detection strategy. In contrast, the method proposed in [5] achieves only 42.55%, which is significantly inferior to our approach.

5.3. Ablation Study

Our dataset is chosen for the ablation study due to its diversity and complexity.

Effectiveness of geometry-guided warping: With image rendering of the pruned scene and inpainting masks, we adopt a warping-refinement scheme rather than directly inpainting the masked areas. To demonstrate the effectiveness of our approach, we disable geometry reconstruction and warping operations, repurposing the multi-view refinement model as a multi-view inpainting model. Specifically, we zero out the masked regions in the input images and attempt to consistently inpaint the missing areas. The inpainting model is also fine-tuned for each given scene, with all other settings remaining unchanged.

We compare images generated with and without the warping process in Fig. 6. Despite the strength of multi-view architecture, direct inpainting fails to generate consistent output with the reference image in large view-change cases, which aligns with the observation in [2]. In contrast, our warping-refinement scheme provides a significant advantage, as the warped content offers strong guidance and regularizes the model to generate consistent results. We then use the generated images to fine-tune 3D scene. As shown in Tab. 2, the fine-tuned scene indeed shows inferior results to our complete method.

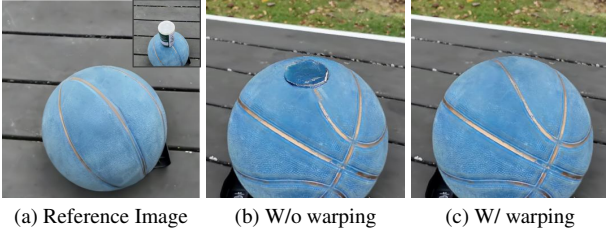


Figure 6. **Multi-view refinement with and without conditioning on the warped image.** (a) shows the inpainted reference with the originally captured image shown in the top right corner. (c) and (b) are the images from an alternate view generated with and without having the warped image as a condition, respectively.

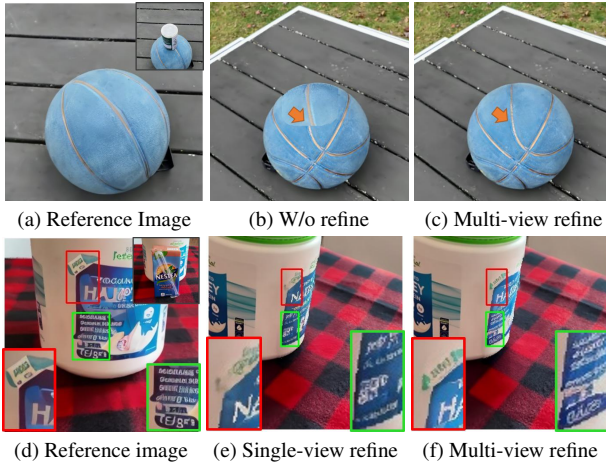


Figure 7. **Ablation on multi-view refinement.** (a) and (d) show the inpainted reference images with the originally captured images shown in the top right corner; (b) and (c) are the images from an alternate view generated without and with the refinement network, respectively; (e) and (f) are the images generated by the single-view and multi-view refinement model, respectively.

Effectiveness of multi-view refinement model: We design the following two variants to illustrate the importance of the multi-view refinement model. a) w/o refinement: the refinement model is removed, and the warped images are directly used to fine-tune the pruned scene. b) single view: we replace the space-time attention with the original self-attention layer, performing refinement independently for

each view without leveraging correlations across different views. When fine-tuning the single-view model for a given scene, only random image pairs are used.

Similarly, we compare images generated by these variants in Fig. 7. As seen in (a), (b), and (c), warping indeed introduces artifacts and texture mismatches, while the multi-view refinement model effectively addresses them. In (d), (e), and (f), the single-view refinement model fails to generate results consistent with the reference image, whereas the multi-view model achieves consistency. We further fine-tune the 3D scene with corresponding images. As illustrated in Tab. 2, our completed method achieves the best result.

Necessity of test-time adaption: Our pipeline requires a per-scene fine-tuned refinement model to generate images. To demonstrate its necessity, we also train a generalized refinement model on the DI3DV dataset, a scene-level multi-view dataset, where we randomly selected two views as the reference and target, synthesized warping artifacts on one, and trained the model to restore. However, as shown in Tab. 2, its performance is inferior to our proposed method due to the limited size of the training set and domain gaps between the training set and real-world cases, such as differences in the severity of artifacts/mismatch. In addition, we use an existing inpainting method, LeftRefill [1], to directly inpaint images based on the reference image. However, as shown in Fig. 8, LeftRefill can only ensures overall visual similarity but lacks pixel-level consistency.

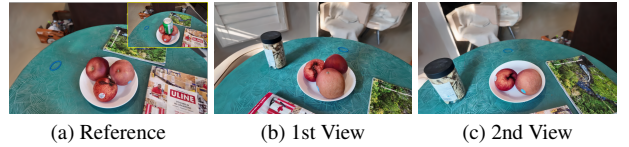


Figure 8. Inpainting results by LeftRefill.

6. Conclusion and Limitation

We present a novel 3D inpainting pipeline that achieves state-of-the-art results across both front-facing and unconstrained benchmarks, with particularly strong performance on the latter. We achieve this by utilizing geometry priors and a multi-view refinement network to generate multi-view consistent inpainted images. Additionally, a novel inpainting mask detection technique is developed to boost performance in unconstrained scenes. The effectiveness of our method is demonstrated through extensive experiments and ablation studies.

Limitations: The multi-view refinement network requires customization for each scene, which may take considerable time. Future advancements in parameter-efficient fine-tuning and the personalization of diffusion models may help to accelerate and further optimize our approach. In addition, our framework relies on monocular depth estimation and image segmentation modules. The accuracy and robustness of these modules can affect the overall performance.

References

- [1] Chenjie Cao, Yunuo Cai, Qiaole Dong, Yikai Wang, and Yanwei Fu. Leftrefill: Filling right canvas based on left reference through generalized text-to-image diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7705–7715, 2024.
- [2] Chenjie Cao, Chaohui Yu, Yanwei Fu, Fan Wang, and Xiangyang Xue. Mvinpainter: Learning multi-view consistent inpainting to bridge 2d and 3d editing. *arXiv preprint arXiv:2408.08000*, 2024.
- [3] Honghua Chen, Chen Change Loy, and Xingang Pan. Mvipnerf: Multi-view 3d inpainting on nerf scenes via diffusion prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5344–5353, 2024.
- [4] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6593–6602, 2024.
- [5] Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21476–21485, 2024.
- [6] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996.
- [7] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.
- [8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [9] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.
- [10] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.
- [11] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.
- [12] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.
- [13] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.
- [14] Chieh Hubert Lin, Changil Kim, Jia-Bin Huang, Qinbo Li, Chih-Yao Ma, Johannes Kopf, Ming-Hsuan Yang, and Hung-Yu Tseng. Taming latent diffusion model for neural radiance field inpainting. In *European Conference on Computer Vision*, pages 149–165. Springer, 2025.
- [15] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023.
- [16] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Marcus A Brubaker, Jonathan Kelly, Alex Levinshtein, Konstantinos G Derpanis, and Igor Gilitschenski. Reference-guided controllable inpainting of neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17815–17825, 2023.
- [17] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G Derpanis, Jonathan Kelly, Marcus A Brubaker, Igor Gilitschenski, and Alex Levinshtein. Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20669–20679, 2023.
- [18] Ashkan Mirzaei, Riccardo De Lutio, Seung Wook Kim, David Acuna, Jonathan Kelly, Sanja Fidler, Igor Gilitschenski, and Zan Gojcic. Reffusion: Reference adapted diffusion models for 3d scene inpainting. *arXiv preprint arXiv:2404.10765*, 2024.
- [19] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4296–4304, 2024.
- [20] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.
- [21] Christian Reiser, Rick Szeliski, Dor Verbin, Pratul Srivivasan, Ben Mildenhall, Andreas Geiger, Jon Barron, and Peter Hedman. Merf: Memory-efficient radiance fields for real-time view synthesis in unbounded scenes. *ACM Transactions on Graphics (TOG)*, 42(4):1–12, 2023.
- [22] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.
- [23] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023.
- [24] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023.
- [25] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023.

- [26] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.
- [27] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.
- [28] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2149–2159, 2022.
- [29] Luming Tang, Nataniel Ruiz, Qinghao Chu, Yuanzhen Li, Aleksander Holynski, David E Jacobs, Bharath Hariharan, Yael Pritch, Neal Wadhwa, Kfir Aberman, et al. Realfill: Reference-driven generation for authentic image completion. *ACM Transactions on Graphics (TOG)*, 43(4):1–12, 2024.
- [30] Dongqing Wang, Tong Zhang, Alaa Abboud, and Sabine Süsstrunk. Innerf360: Text-guided 3d-consistent object inpainting on 360-degree neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12677–12686, 2024.
- [31] Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv preprint arXiv:2312.02201*, 2023.
- [32] Yuxin Wang, Qianyi Wu, Guofeng Zhang, and Dan Xu. Gstream: Learning 3d geometry and feature consistent gaussian splatting for object removal. *arXiv preprint arXiv:2404.13679*, 2024.
- [33] Ethan Weber, Aleksander Holynski, Varun Jampani, Saurabh Saxena, Noah Snavely, Abhishek Kar, and Angjoo Kanazawa. Nerfiller: Completing scenes via generative 3d inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20731–20741, 2024.
- [34] Silvan Weder, Guillermo Garcia-Hernando, Aron Monszpart, Marc Pollefeys, Gabriel J Brostow, Michael Firman, and Sara Vicente. Removing objects from neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16528–16538, 2023.
- [35] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7623–7633, 2023.
- [36] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18381–18391, 2023.
- [37] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024.
- [38] Hu Ye, Jun Zhang, Sibio Liu, Xiao Han, and Wei Yang. Ip-adapt: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.
- [39] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.
- [40] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022.