PureDiff: Purifying Sparse-View Reconstructions with Distractors via One-Step Diffusion Model

Qirui Hu[1⋆] Jingjing Wang[1⋆] Chong Bao[1†] Xiyu Zhang[1] Zhaopeng Cui[1] Guofeng Zhang [1†]
[1] *State Key Lab of CAD&CG, Zhejiang University*

***Abstract*—Neural rendering from casual sparse-view captures with dynamic distractors is challenging. Existing sparse-view methods assume static scene and ignore distractors, thereby yielding artifacts, while dense-view distractor removal based on fitting residual assumption fails under sparse-view settings. We propose PureDiff, a novel one-step diffusion model enhanced with multi-view epipolar attention for high-quality, distractor-free reconstruction from casual sparse-view captures. PureDiff serves dual roles. As a Cleaner, it adopts a divide-and-conquer strategy with intra-set training and inter-set contrastive inference to effectively detect distractors. As a Refiner, it repairs severe rendering artifacts (e.g., holes and floaters) and feeds refined results back as additional supervision to improve reconstruction quality. To fit sparse-view settings, we further introduce an efficient few-shot fine-tuning scheme that significantly reduces training cost and data requirements. Experiments show that PureDiff consistently outperforms state-of-the-art methods in both rendering quality and distractor removal on casual sparse-views with dynamic distractors. Code will be public upon acceptance.**

***Index Terms*—neural rendering, sparse reconstruction, gaussian splatting, distractors, diffusion model**

I. Introduction

Neural rendering techniques, particularly Neural Radiance Fields [1] (NeRF) and 3D Gaussian Splatting [2] (3DGS), enable efficient high-fidelity 3D reconstruction and photo-realistic novel view synthesis from multi-view captures of static scenes. However, the inherent dynamism of the real world presents a substantial challenge. Casual captures often contain pedestrians, vehicles, or other transient objects. These view-dependent distractors violate the static scene assumption crucial to NeRF and 3DGS, resulting in notable degradation of reconstruction and rendering quality.

A common strategy for handling distractors relies on a "fitting residual" assumption [3]–[7]. This assumption posits that distractors, which break multi-view consistency, will exhibit high residuals and retarded optimization convergence. Its effectiveness depends on multi-view conflicts between dynamic distractors and the static scene content they occlude. Such conflicts necessitate exhaustive scene captures, often requiring hundreds of images, so that every occluded region is clearly observed from other viewpoints and accurately reconstructed to highlight distractors as high-residual areas. Unfortunately, casual captures in daily life rarely provide such detailed and exhaustive data.

This work was partially supported by NSF of China (No. 62425209).
⋆ indicates equal contribution. † indicates corresponding author.

Reconstructing 3D scenes from casual sparse-view captures with dynamic distractors therefore remains largely underexplored. The interplay between sparse-view data and distractors makes it more complicated. Existing studies either focus on the sparse-view problem for static scenes [8]–[11] or address distractors in dense-view captures [3]–[7]. Sparse-view methods [8]–[10] typically introduce geometric priors or heuristic regularization to compensate for missing observations, but the resulting reconstructions are often fragile and generalize poorly to novel viewpoints. Other methods [11] adopt fine-tuned image diffusion models with ControlNet [12] to refine rendered novel views, but they mainly handle minor textural issues (e.g., slight blur or noise) and cannot fix severe artifacts like missing regions or large floaters due to a lack of 3D awareness. While recent works [13] train multi-view diffusion models for view-consistent 3D refinement, they often require large-scale training data and generalize poorly to diverse real-world scenes, particularly those containing distractors. Notably, all the above methods assume static scenes, and the presence of distractors breaks multi-view consistency, weakening the effectiveness of their priors and regularization.

To address this problem, we introduce **PureDiff**, a novel one-step diffusion model designed to enable high-quality, distractor-free 3D reconstructions under casual sparse-view captures with distractors. PureDiff acts as both a **Cleaner**, to identify and remove distractors from the sparse-input images, and a **Refiner** to rectify severe artifacts in 3DGS renderings.

PureDiff is formulated as a rendering-repair diffusion model that processes distorted rendered inputs and yield refined images with enhanced consistency and fidelity. Our task is distinct from conventional denoising-based applications, such as image generation from Gaussian noise [14] or monocular depth estimation from clean images. The core challenge lies in our input: distorted renderings with inaccurate pixel values that do not conform to a stable distribution. To stabilize this challenging denoising process, we introduce a multi-view epipolar attention module that effectively incorporates geometric constraints across views, substantially improving the refinement of severe rendering artifacts such as holes and floaters. Furthermore, we leverage a scene-specific few-shot fine-tuning strategy. This enables PureDiff to acquire robust repair capabilities from sparse inputs in a few hours on a single RTX 4090 GPU.

As a Cleaner, PureDiff provides a novel distractor cleaning strategy without relying on the fitting residual assumption. We observe that a diffusion model tends to memorizes

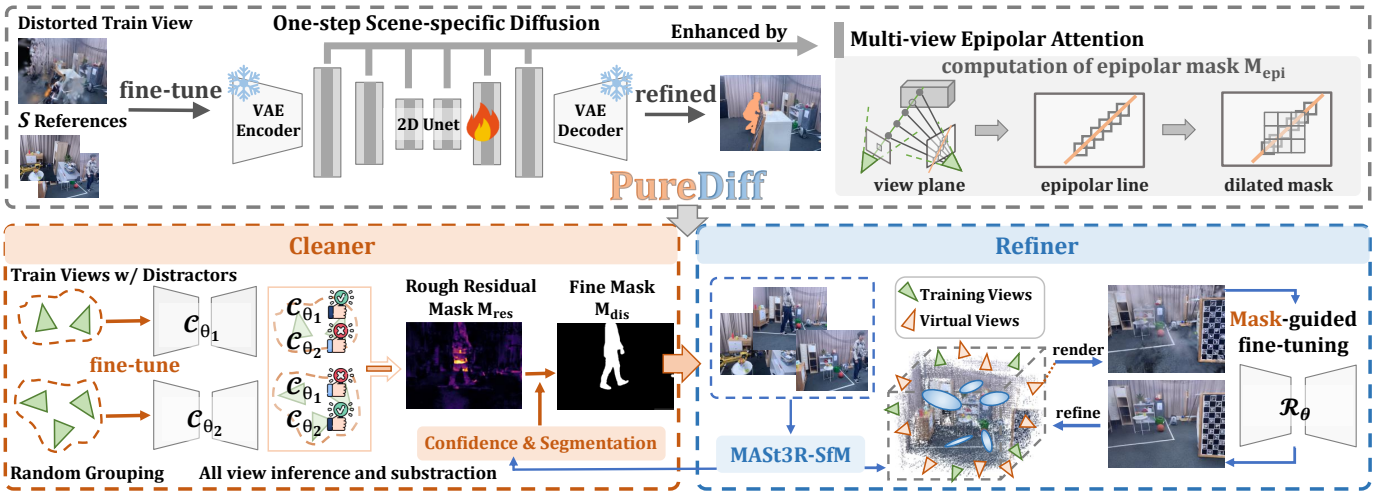

Fig. 1: **Overview.** We propose PureDiff, a unified one-step diffusion model with multi-view epipolar attention (top). Given sparse inputs with distractors, it is first instantiated as a Cleaner to expose distractors via prediction discrepancies (left), and then as a Refiner to refine distorted virtual-view renderings for 3DGS supervision (right).

distractors at the training viewpoints, but when applied to novel viewpoints with different layouts or content distribution, it naturally produces clean results because the distractor-free scene representation aligns better with its pre-trained distribution. Leveraging this property, we propose a divide-and-conquer cleaning strategy that performs intra-set training on subsets of views and inter-set contrastive inference across subsets to robustly detect distractors. As a **Refiner**, PureDiff repairs severe artifacts in 3DGS renderings and is seamlessly integrated into the 3DGS optimization loop. We sample virtual viewpoints during optimization and feed the repaired renderings back as additional supervision, enabling richer multi-view constraints and more accurate reconstruction.

Our main contributions are summarized as follows:

- We propose PureDiff, a novel one-step rendering-repair diffusion model that incorporates a multi-view epipolar attention mechanism and supports efficient scene-specific few-shot fine-tuning.
- We introduce a divede-and-conquer distractor cleaning strategy based on intra-set training and inter-set contrastive inference, eliminating the reliance on fitting residuals.
- We integrate PureDiff into the 3DGS training loop to achieve high-quality, distractor-free reconstructions under casual, sparse-view captures, outperforming existing methods both quantitatively and qualitatively.

II. Related Work

Sparse-View Novel View Synthesis. Sparse-view Novel View Synthesis(NVS) aims to generate unseen views from sparse inputs. Although NeRF [1] and 3DGS [2] produce high-quality results under dense views, their performance degrades in sparse settings. To address this, existing works either introduce regularization or structural designs [9], [10], exploit geometric priors [8], or adopt feed-forward designs that directly predict geometry or appearance for efficient sparse reconstruction [15]. However, these methods typically assume static scenes and struggle in the presence of transient distractors, limiting their applicability to real-world scenarios. Recently, diffusion models have introduced strong generative priors for sparse-view NVS. Some approaches incorporate pre-trained 2D diffusion models into 3D optimization via scoring functions or view refinement [11], [13]. Others lifts 2D diffusion to the video or 3D domain, using video diffusion models [16] and multi-view diffusion models [17] to directly synthesize view-consistent scenes. While improving fidelity and consistency, these methods often require large training data or multi-step inference , and generalize poorly to complex real-world scenes with distractors. In contrast, our approach enables efficient and robust one-step refinement using only sparse views.

Distractor-Free Novel View Synthesis. Casual real-world captures often violate the static scene assumption of the traditional NVS task. Some existing methods tackle this problem through robust optimization (e.g., uncertainty modeling, robust losses) [3], [4], [7], [18]; several leverage semantic features from pretrained models [14] to enhance spatial-temporal consistency. Others explicitly model dynamics using transient fields [19], per-view [6], or hybrid 2D/3D representations [20]. A third direction identifies distractors through (self-)supervised learning [21] or combines heuristic cues with SAM [22], [23]. However, under sparse-view settings, many of these methods struggle due to limited multi-view redundancy and insufficient cues for reliable distractor identification.

III. Preliminary

Diffusion Models (DMs) [14] consist of a forward diffusion process and a reverse denoising process, both defined as Markov chains. The forward process gradually adds Gaussian noise $\epsilon \in \mathcal{N}(0, I)$ to real data $x_0 \sim p_{data}$ over time via:

$$\begin{aligned} q(x_t|x_{t-1}) &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\epsilon, \\ q(x_t|x_0) &= \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \end{aligned} \tag{1}$$

where t is the timestep uniformly sampled from $[1, T]$ (usually $T = 1000$), α_t controls the noise level at each step and $\bar{\alpha}_t =$

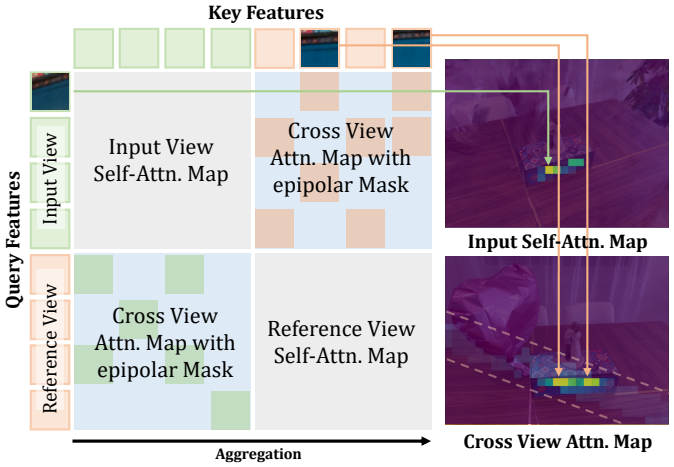

Fig. 2: **Illustration of the MEA.** It easily ignores the distractor area and focuses on the geometrically corresponding area.

$\prod_{i=1}^{t} \alpha_t$. DMs learn the reverse process $p_\theta(x_{t-1}|x_t)$, which denoises a random sample $x_T \in \mathcal{N}(0, I)$ to recover data from the original distribution. In this work, we use the velocity of "v-prediction" v_θ [24] as the learning target for p_θ. And the training objective is:

$$\begin{aligned} \mathcal{L} &= \mathbb{E}\left[\|\mathbf{v}_t^{\text{target}} - v_\theta(x_t, t)\|_2^2\right], \\ \text{where } \mathbf{v}_t^{\text{target}} &\triangleq \sqrt{\bar{\alpha}_t}\epsilon - \sqrt{1-\bar{\alpha}_t}x_0. \end{aligned} \tag{2}$$

Expectation $\mathbb{E}[\cdot]$ is over $t \sim \mathcal{U}(1, 1000)$ and $\epsilon \sim \mathcal{N}(0, I)$.

IV. Method

Task Formulation. Given casual sparse-view captures with visual distractors, our goal is to achieve photorealistic novel view synthesis from arbitrary viewpoints. We propose PureDiff, a scene-specific one-step diffusion model that serves two roles: 1) a **Cleaner** that identifies distractors masks, and 2) a **Refiner** that refines poorly reconstructed views to improve sparse-view reconstruction. The overview is shown in Fig. 1.

A. PureDiff: One-Step Geometry-aware Diffusion Model

We propose a scene-specific one-step image-to-image diffusion model to refine distorted 3DGS renderings under sparse-view supervision. Directly fine-tuning 2D diffusion models [14] is challenging due to 1) the lack of explicit multi-view geometric constraints, and 2) the high data demand and low efficiency of multi-step denoising. To address these challenges, PureDiff introduces geometry-aware multi-view epipolar attention and an efficient one-step diffusion formulation.

Multi-View Epipolar Attention. Given a distorted rendering $\hat{I}$, we condition the diffusion model on S nearby ground-truth images $\{I_{\text{ref}_i}\}_{i=1}^{S}$ as the reference set to provide multi-view cues. To explicitly enforce geometry consistency, we introduce Multi-view Epipolar Attention (MEA), which restricts cross-view feature interaction along epipolar correspondences. For each self-attention layer of the 2D UNet, we concatenate the latent features of the target view $\mathbf{z}_{\hat{I}} \in \mathbb{R}^{C\times H\times W}$ with those of S reference views $\{\mathbf{z}_{\text{ref}_i}\}_{i=1}^{S} \in \mathbb{R}^{S\times C\times H\times W}$, extending the spatial attention domain to $(S+1)\times H\times W$, where C denotes feature channels, and H, W the spatial resolution.

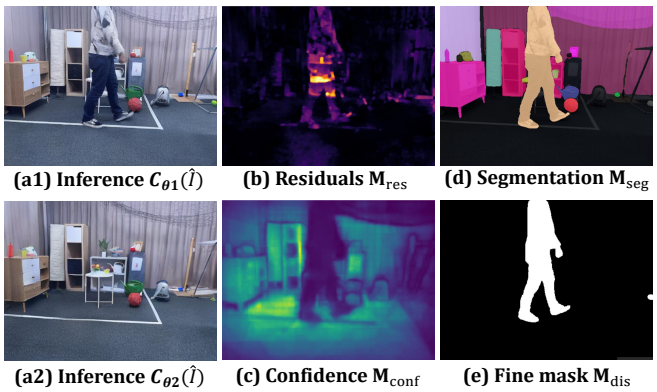

Fig. 3: Visualization of distractor mask generation pipeline.

To enforce geometry consistency, we precompute an epipolar attention mask $\mathbf{M}_{\text{epi}} \in \mathbb{B}^{S^+HW\times S^+HW}$ for all $S^+ = S+1$ views, using relative camera poses and intrinsics. Specifically, for each pixel (i, j) in the query view, we compute its epipolar line l_{ij} in each reference, formulated as $l_{ij} = \mathbf{F}\cdot\begin{bmatrix} i & j & 1\end{bmatrix}^\top$, where $\mathbf{F}$ is the fundamental matrix [25]. The epipolar line encodes the geometric correspondence prior that the projection of a query pixel onto a neighboring view must lie along this line. Pixels located on l_{ij} are marked in the epipolar mask, which is then dilated with a 3×3 kernel to account for discretization and minor misalignments. Repeating this process across all reference views and spatial locations yields $\mathbf{M}_{\text{epi}}$. Formally, the multi-view epipolar attention (MEA) is computed as:

$$\begin{aligned} &\mathbf{z} \leftarrow \text{rearrange}\left(\left[\mathbf{z}_{\hat{I}} \mid \{\mathbf{z}_{\text{ref}_i}\}_{i=1}^{S}\right], S^+CHW \rightarrow S^+(CHW)\right), \\ &\text{Attn}(\mathbf{z}) = \text{Softmax}\left(\frac{Q(\mathbf{z})\cdot K(\mathbf{z})^T}{\sqrt{d}} + (1-\mathbf{M}_{\text{epi}})\cdot(-\inf)\right), \\ &MEA(\mathbf{z}) = \text{Attn}(\mathbf{z})\cdot V(\mathbf{z}), \end{aligned} \tag{3}$$

where Q, K and V are learnable projections mapping input features into query, key and value. In addition to intra-view attention, our MEA mechanism enables inter-view attention constrained to projected epipolar lines, encouraging information exchange along geometrically consistent correspondences.

Efficient Few-Shot Fine-Tuning. To efficiently fine-tuning the diffusion model, we degenerate the formulation into a one-step mapping between the distorted rendering $\hat{I}$ and the target image I. This is achieved by setting the starting noise ϵ to the artifact-containing rendering $\hat{I}$ in Eq. 2, fixing alpha values $\{\alpha_t\}_{t=1}^{T}$ of all steps to zero, and defining x_0 as the refined target view I. This yields the following training objective:

$$\mathcal{L}_{\text{MSE}} = \mathbb{E}_{t=1000}\left[\| - I - \mathcal{R}_\theta(\hat{I}_t, \{I_{\text{ref}_i t}\}_{i=1}^{S})\|_2^2\right], \tag{4}$$

where the model always predicts the negative of the target view. We optimize the diffusion model at a fixed timestep $t = 1000$, enabling one-step training and inference while preserving diffusion priors. During fine-tuning, we freeze the VAE and update only the 2D Unet. The model is supervised in the pixel space after VAE decoding using both MSE and LPIPS [26] losses to enhance the visual fidelity [27]. The LPIPS loss encourages the model to learn high-level semantic features and global appearance from ground truth, while being tolerant to low-level spatial misalignments [28]. The final training loss is defined as $\mathcal{L}_{\mathcal{P}} = \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{LPIPS}}$.

Dataset	On-the-go												RobustNeRF			Tanks and Temples					
Method	Low Occlusion PSNR↑	SSIM↑	LPIPS↓	Medium Occlusion PSNR↑	SSIM↑	LPIPS↓	High Occlusion PSNR↑	SSIM↑	LPIPS↓	Average PSNR↑	SSIM↑	LPIPS↓	Average PSNR↑	SSIM↑	LPIPS↓	12 Views PSNR↑	SSIM↑	LPIPS↓	24 Views PSNR↑	SSIM↑	LPIPS↓
FSGS [8]	17.21	0.445	0.352	18.10	0.630	0.292	15.39	0.425	0.473	16.90	0.500	0.372	19.32	0.649	0.340	14.92	0.462	0.429	17.62	0.592	0.316
CoR-GS [9]	17.55	0.480	0.339	18.02	0.632	0.301	15.60	0.449	0.463	17.06	0.519	0.368	19.29	0.667	0.336	15.19	0.512	0.419	17.80	0.618	0.300
LMGaussian [11]	18.14	0.484	0.442	18.89	0.656	0.342	16.86	0.455	0.501	17.96	0.532	0.429	19.48	0.673	0.389	14.46	0.488	0.517	17.26	0.589	0.415
DropGaussian [10]	17.95	0.502	0.335	19.07	0.687	0.265	15.62	0.475	0.470	17.54	0.555	0.357	18.94	0.687	0.344	15.22	0.499	0.420	17.87	0.627	0.310
Difix3D+ [13]	17.68	0.493	0.349	19.15	0.667	0.279	17.55	0.475	0.431	18.13	0.545	0.353	19.35	0.653	0.339	14.60	0.469	0.436	17.56	0.623	0.314
NeRF On-the-go [4]	14.55	0.267	0.515	19.24	0.604	0.306	17.93	0.374	0.435	17.24	0.415	0.419	19.87	0.628	0.345	-	-	-	-	-	-
SLS-MLP [3]	16.82	0.417	0.366	19.58	0.651	0.270	16.08	0.409	0.453	17.49	0.492	0.363	19.81	0.642	0.339	-	-	-	-	-	-
WildGaussians [7]	16.00	0.423	0.401	17.64	0.600	0.333	16.40	0.438	0.429	16.68	0.487	0.388	18.23	0.617	0.384	-	-	-	-	-	-
DroneSplat [5]	15.84	0.394	0.380	19.28	0.639	0.273	15.83	0.412	0.452	16.98	0.482	0.368	18.68	0.616	0.349	-	-	-	-	-	-
DeSplat [6]	16.86	0.433	0.359	19.48	0.647	0.277	18.50	0.488	0.369	18.28	0.522	0.335	19.05	0.630	0.356	-	-	-	-	-	-
Ours-PureDiff	18.87	0.519	0.329	21.69	0.758	0.205	20.70	0.593	0.289	20.42	0.623	0.274	21.16	0.725	0.295	16.26	0.541	0.387	18.62	0.654	0.288

TABLE I: Quantitative results of sparse-view NVS w & w/o distractors. First, second and third best results are highlighted.

B. PureDiff as a Cleaner

We first employ PureDiff as a *Cleaner* to identify transient distractors. This design is motivated by two observations: 1) distractors vary significantly across sparse views, leading to view-inconsistent artifacts in GS renderings; 2) a scene-specific diffusion model tends to memorize distractors in its training views but produces distractor-free outputs for unseen views. Based on this, we adopt a divide-and-conquer strategy by training two identical PureDiff-based cleaners, $\mathcal{C}_{\theta_1}$ and $\mathcal{C}_{\theta_2}$, on disjoint subsets of distractor-bearing views and detecting distractors via their prediction discrepancies. Given sparse ground-truth views $\{I_i\}_{i=1}^N$, the Cleaner $\mathcal{C}_{\theta_1,\theta_2}(\hat{I}, \{I_{\text{ref}_i}\}_{i=1}^S)$ produces distractor masks $\{\mathbf{M}_{\text{res}i}\}_{i=1}^N$.

Rough Residual Masks for Distractors. We randomly split $\mathcal{D}_{\text{train}}$ into two disjoint subsets, which naturally induces large view shifts between the two subsets under sparse-view settings. Each cleaner is fine-tuned on its own subset using $\mathcal{L}_{\mathcal{P}}$. For each view, we run inference with both cleaners and compute rough residual masks by thresholding their absolute differences:

$$\{\mathbf{M}_{\text{res}i}\}_{i=1}^N = \{|\, \mathcal{C}_{\theta_1}(\hat{I}_i) - \mathcal{C}_{\theta_2}(\hat{I}_i) \,| > \tau_{\text{res}}\}_{i=1}^N. \tag{5}$$

Since each cleaner tends to generalize to distractor-free outputs on unseen views (Fig. 3(a1)(a2)), their discrepancies mainly arise in distractor regions (Fig. 3(b)).

Fine Masks for Distractors. The rough residuals may include noise from underfitted background regions (Fig. 3(b)). To suppress it, we incorporate confidence maps $\{\mathbf{M}_{\text{conf}i}\}_{i=1}^N$ from Mast3R-SfM [29] (also used for initializing 3DGS point clouds in Sec. IV-C) with threshold τ_{conf}, which highlight view-consistent areas and allow us to exclude background regions not finely refined (Fig. 3(c)). Note that only using confidence maps will introduce extra areas without distractors. We further refine the masks using segmentation results $\{\mathbf{M}_{\text{seg}_i}\}_{i=1}^N$ from pre-trained models [23] to obtain accurate object boundaries (Fig. 3(d)), retaining regions whose intersection exceeds τ_{seg}. The final masks $\{\mathbf{M}_{\text{dis}i}\}_{i=1}^N$ are computed as:

$$\{\mathbf{M}_{\text{dis}i}\}_{i=1}^N = \{(\mathbf{M}_{\text{res}})_i \wedge \neg(\mathbf{M}_{\text{conf}})_i \wedge (\mathbf{M}_{\text{seg}})_i\}_{i=1}^N. \tag{6}$$

C. PureDiff as a Refiner

We further use PureDiff as a *Refiner*, $\mathcal{R}_\theta(\hat{I}, \{I_{\text{ref}_i}\}_{i=1}^S)$, to transform distorted 3DGS renderings $\hat{I}$ into photorealistic images $\tilde{I}$. The refined images are fed back into the 3DGS optimization as additional supervision, augmenting sparse-view reconstruction with richer multi-view cues.

Fine-tuning the Refiner. For each scene, we perform few-shot fine-tuning of $\mathcal{R}_\theta$ on $\mathcal{D}_{\text{train}}$. To suppress the influence of distractors in both references and ground-truth images, we apply mask-guided supervision using the final distractor masks $\mathbf{M}_{\text{dis}}$ extracted by the Cleaner (Sec. IV-B). The Refiner is optimized with the loss:

$$\mathcal{L}_{\text{R}} = \mathbf{M}_{\text{dis}} \odot \mathcal{L}_{\text{MSE}} + \mathbf{M}_{\text{dis}} \odot \mathcal{L}_{\text{LPIPS}}, \tag{7}$$

where pixel-wise LPIPS features are used to compute the masked perceptual loss.

Refiner-Assisted Virtual-View Augmentation for 3DGS. We initialize the 3DGS model $\mathcal{G}$ using MASt3R-SfM [29]. To improve scene coverage under sparse views, we sample virtual viewpoints via circular interpolation and 4-DoF random perturbations, filtering invalid viewpoints that cause poor scene visibility. After warming up $\mathcal{G}$ on the ground-truth image set $\mathcal{D}_{\text{gt}}$ for n_{warm} iterations, we render pseudo views at the sampled virtual viewpoints and refine them with pre-finetuned $\mathcal{R}_\theta$, yielding a distractor-free pseudo dataset $\mathcal{D}_{\text{pseudo}}$. $\mathcal{D}_{\text{gt}}$ and $\mathcal{D}_{\text{pseudo}}$ are jointly used for subsequent optimization.

During training, we supervise ground-truth images $\mathcal{D}_{\text{gt}}$ with L1 and SSIM losses following 3DGS [2], while masking distractor regions using $\mathbf{M}_{\text{dis}}$ and disabling gradient propagation within masked areas to ensure that distractors remain "invisible" to $\mathcal{G}$. For pseudo images, we adopt L1 and LPIPS losses to accommodate pixel-wise uncertainty and encourage perceptual consistency. To further enhance geometric fidelity, we incorporate monocular normal [30] and depth [31] priors: $\mathcal{L}_{\text{Reg}} = \lambda_{\text{N}}\mathcal{L}_{\text{Norm}} + \lambda_{\text{D}}\mathcal{L}_{\text{Depth}}$. The final loss is defined as:

$$\begin{aligned} \mathcal{L}_{\text{gt}} &= (1-\lambda_{\text{S}})\mathcal{L}_1 \circ \mathbf{M}_{\text{dis}} + \lambda_{\text{S}}\mathcal{L}_{\text{SSIM}} \circ \mathbf{M}_{\text{dis}} + \mathcal{L}_{\text{Reg}}, \\ \mathcal{L}_{\text{pseudo}} &= (1-\lambda_{\text{L}})\mathcal{L}_1 + \lambda_{\text{L}}\mathcal{L}_{\text{LPIPS}} + \mathcal{L}_{\text{Reg}}, \end{aligned} \tag{8}$$

where we set $\lambda_{\text{N}} = 0.1$, $\lambda_{\text{D}} = 0.2$, $\lambda_{\text{L}} = 0.9$, and $\lambda_{\text{S}} = 0.2$.

V. Experiments

Dataset. We evaluate PureDiff on two distractor-aware datasets, On-the-go [4] and RobustNeRF [18], as well as the standard NVS benchmark Tanks and Temples (T&T) [32]. On-the-go includes six scenes with low to high distractor ratios, where 24 views are selected for training. RobustNeRF contains four scenes with 16 selected training views. For both datasets, training views are selected by clustering camera centers using K-means ($k = 24$ or 16) and choosing the nearest image to each cluster center to ensure spatial coverage, while the original

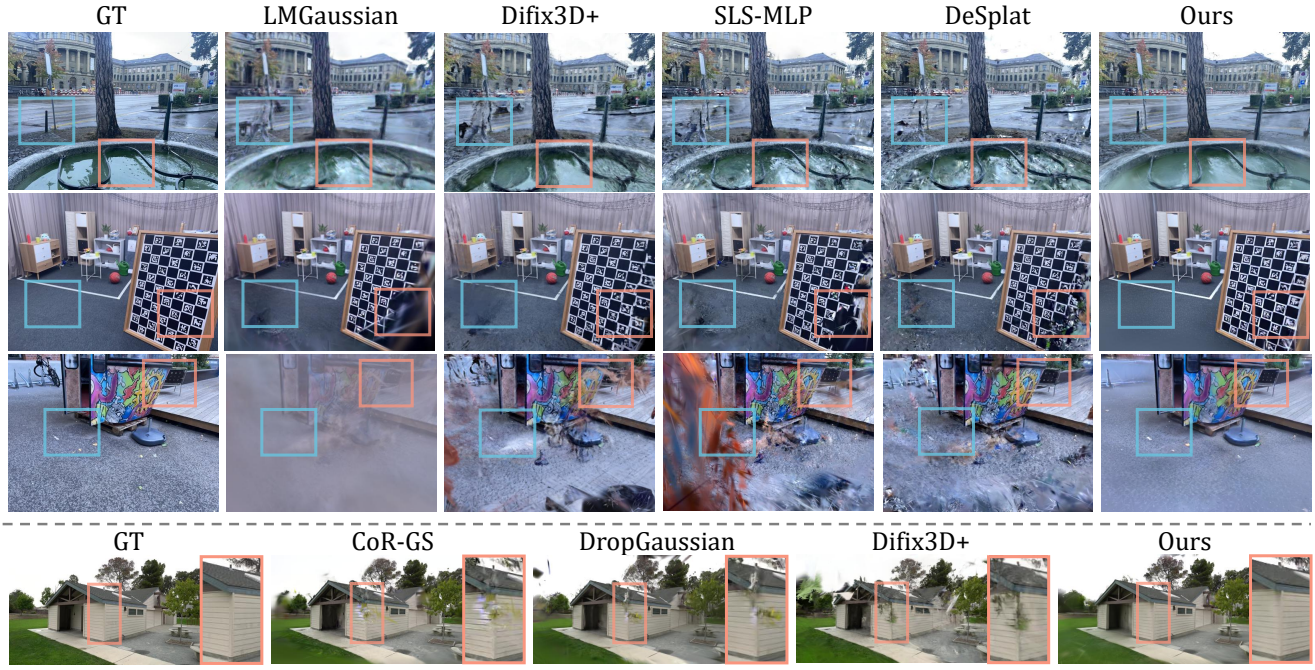

Fig. 4: Qualitative results of sparse-view NVS with distractors (top) and without distractors (bottom).

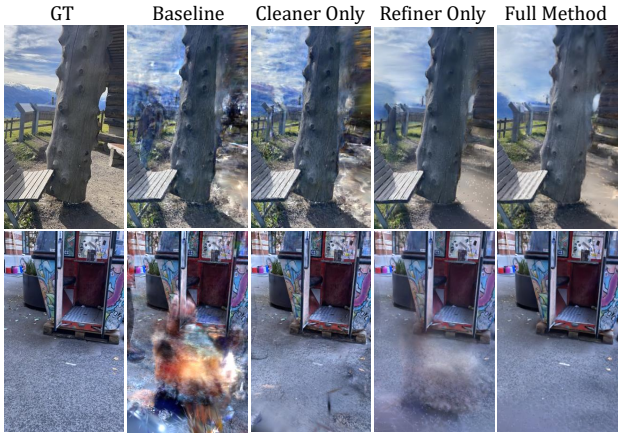

Fig. 5: Qualitative ablation results of the Cleaner and Refiner.

test sets are used. T&T includes seven scenes with 12 or 24 uniformly sampled training views, every 8th image for testing.

Implementation Details We use Stable Diffusion v2-1 [14] as the backbone for PureDiff. To construct $\mathcal{D}_{\text{train}}$, we divide the training set into $M = 4$ groups and train each 3DGS model for $n_{iter} = 1500$ iterations. For the Refiner, we fine-tune the model for 2000 iterations with a learning rate of 2×10^{-4}, and set $S = 2$ to balance memory consumption and performance. For On-the-go and RobustNeRF, the diffusion model is fine-tuned at a resolution of (376, 504), taking on average 5.5 hours. For T&T, the diffusion model is fine-tuned at a resolution of (272, 480), taking on average 2.5 hours. The Cleaner uses the same learning rate and is trained for 400 iterations. For binary mask generation, we set the thresholds $\tau_{\text{res}} = 0.3$, $\tau_{\text{conf}} = 10$, and $\tau_{\text{seg}} = 0.6$. When training the 3DGS model $\mathcal{G}$, we first optimize it with $\mathcal{D}_{\text{gt}}$ for $n_{\text{warm}} = 1500$ iterations, then continue training with both $\mathcal{D}_{\text{gt}}$ and $\mathcal{D}_{\text{pseudo}}$ up to 30K iterations. All

Method	On-the-go PSNR↑	SSIM↑	LPIPS↓
Difix3D+ [best few-shot NVS]	18.13	0.545	0.353
DeSplat [best distractor-free NVS]	18.28	0.522	0.335
Baseline	16.59	0.482	0.377
Baseline + Cleaner	18.72	0.562	0.285
Baseline + Refiner	19.45	0.595	0.313
Full method (PureDiff)	**20.42**	**0.623**	**0.274**

TABLE II: Quantitative ablation results. **Best** values are bolded.

baselines follow their default settings and are initialized with the same point clouds as ours.

A. Main Results

Sparse-View NVS with Distractors. Quantitative and qualitative comparisons with strong baselines are shown in Tab. I and Fig. 4. Our method achieves the best PSNR, SSIM, and LPIPS. Existing distractor-free NVS methods fail to remove distractors even under low occlusion due to their reliance on multi-view consistency (e.g., the pedestrian shadow in 1st row of Fig. 4), leading to floaters and artifacts that become more severe in high-occlusion cases (e.g., the 3rd rows of Fig. 4). Diffusion-based methods such as LMGaussian and Difix3D+ alleviate some artifacts but suffer from detail loss. In contrast, our method enables accurate missing-region completion and high-quality refinement by effectively leveraging self- and cross-view correlations.

Sparse-View NVS without Distractors. We further evaluate our method on the distractor-free T&T dataset. As shown in Tab. I and Fig. 4, our method consistently outperforms all baselines, effectively removing artifacts. These results demonstrate the strong generalization of our approach under sparse inputs, regardless of the presence of distractors.

B. Ablation Studies

We conduct incremental ablations on the 3DGS reconstruction pipeline on On-the-go and RobustNeRF, including the baseline, Cleaner only, Refiner only, and the full model. As shown in Tab. II and Fig. 5, both modules contribute substantially and are complementary. The Refiner alone outperforms the Cleaner, but struggles to fully suppress distractors under heavy occlusion (e.g., the bottom row of Fig. 5). In highly occluded scenes, the Refiner without explicit mask guidance leaves residual artifacts, despite improving view refinement. Conversely, the Cleaner alone cannot resolve sparse-view ambiguity. Combining both modules yields the best performance by jointly addressing distractors and view sparsity. Tab. II further shows that the Cleaner-only and Refiner-only variants already outperform DeSplat and Difix3D+, respectively, confirming the effectiveness of each component.

VI. Conclusion

We present PureDiff, a one-step diffusion model for sparse-view reconstruction with distractor removal. By effectively leveraging multi-view cues and diffusion priors, our method produces high-quality, distractor-free reconstructions from casual sparse-view captures, outperforming existing approaches. While effective, PureDiff still faces challenges under extremely sparse inputs or consistently present distractors that appear across most views, which we leave for future work.

References

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[2] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[3] S. Sabour, L. Goli, G. Kopanas, M. Matthews, D. Lagun, L. Guibas, A. Jacobson, D. Fleet, and A. Tagliasacchi, "Spotlesssplats: Ignoring distractors in 3d gaussian splatting," *ACM Transactions on Graphics*, vol. 44, no. 2, pp. 1–11, 2025.

[4] W. Ren, Z. Zhu, B. Sun, J. Chen, M. Pollefeys, and S. Peng, "Nerf on-the-go: Exploiting uncertainty for distractor-free nerfs in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8931–8940.

[5] J. Tang, Y. Gao, D. Yang, L. Yan, Y. Yue, and Y. Yang, "Dronesplat: 3d gaussian splatting for robust 3d reconstruction from in-the-wild drone imagery," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 833–843.

[6] Y. Wang, M. Klasson, M. Turkulainen, S. Wang, J. Kannala, and A. Solin, "Desplat: Decomposed gaussian splatting for distractor-free rendering," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 722–732.

[7] J. Kulhanek, S. Peng, Z. Kukelova, M. Pollefeys, and T. Sattler, "Wildgaussians: 3d gaussian splatting in the wild," *arXiv preprint arXiv:2407.08447*, 2024.

[8] Z. Zhu, Z. Fan, Y. Jiang, and Z. Wang, "Fsgs: Real-time few-shot view synthesis using gaussian splatting," in *European conference on computer vision*. Springer, 2024, pp. 145–163.

[9] J. Zhang, J. Li, X. Yu, L. Huang, L. Gu, J. Zheng, and X. Bai, "Cor-gs: sparse-view 3d gaussian splatting via co-regularization," in *European Conference on Computer Vision*. Springer, 2024, pp. 335–352.

[10] H. Park, G. Ryu, and W. Kim, "Dropgaussian: Structural regularization for sparse-view gaussian splatting," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 21 600–21 609.

[11] H. Yu, X. Long, and P. Tan, "Lm-gaussian: Boost sparse-view 3d gaussian splatting with large model priors," *arXiv preprint arXiv:2409.03456*, 2024.

[12] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," 2023.

[13] J. Z. Wu, Y. Zhang, H. Turki, X. Ren, J. Gao, M. Z. Shou, S. Fidler, Z. Gojcic, and H. Ling, "Difix3d+: Improving 3d reconstructions with single-step diffusion models," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 26 024–26 035.

[14] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[15] J. Xu, S. Gao, and Y. Shan, "Freesplatter: Pose-free gaussian splatting for sparse-view 3d reconstruction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 25 442–25 452.

[16] C. Bao, X. Zhang, Z. Yu, J. Shi, G. Zhang, S. Peng, and Z. Cui, "Free360: Layered gaussian splatting for unbounded 360-degree view synthesis from extremely sparse and unposed views," in *The IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[17] M. Liu, R. Shi, L. Chen, Z. Zhang, C. Xu, X. Wei, H. Chen, C. Zeng, J. Gu, and H. Su, "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 (CVPR)*, June 2024, pp. 10 072–10 083.

[18] S. Sabour, S. Vora, D. Duckworth, I. Krasin, D. J. Fleet, and A. Tagliasacchi, "Robustnerf: Ignoring distractors with robust losses," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 20 626–20 636.

[19] W. Park, M. Nam, S. Kim, S. Jo, and S. Lee, "Forestsplats: Deformable transient field for gaussian splatting in the wild," *arXiv preprint arXiv:2503.06179*, 2025.

[20] J. Lin, J. Gu, L. Fan, B. Wu, Y. Lou, R. Chen, L. Liu, and J. Ye, "Hybridgs: Decoupling transients and statics with 2d and 3d gaussian splatting," *arXiv preprint arXiv:2412.03844*, 2024.

[21] R. Wang, Q. Lohmeyer, M. Meboldt, and S. Tang, "Degauss: Dynamic-static decomposition with gaussian splatting for distractor-free 3d reconstruction," in *ICCV 2025 Workshop on Wild 3D: 3D Modeling, Reconstruction, and Generation in the Wild*, 2025.

[22] J. Chen, Y. Qin, L. Liu, J. Lu, and G. Li, "Nerf-hugs: Improved neural radiance fields in non-static scenes using heuristics-guided segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 436–19 446.

[23] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[24] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=TIdIXIpzhoI

[25] M. Suhail, C. Esteves, L. Sigal, and A. Makadia, "Generalizable patch-based neural rendering," in *European Conference on Computer Vision*. Springer, 2022, pp. 156–174.

[26] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[27] N. Elata, B. Kawar, Y. Ostrovsky-Berman, M. Farber, and R. Sokolovsky, "Novel view synthesis with pixel-space diffusion models," 2025.

[28] R. Gao, A. Holynski, P. Henzler, A. Brussee, R. Martin-Brualla, P. Srinivasan, J. T. Barron, and B. Poole, "Cat3d: Create anything in 3d with multi-view diffusion models," *arXiv preprint arXiv:2405.10314*, 2024.

[29] B. Duisterhof, L. Zust, P. Weinzaepfel, V. Leroy, Y. Cabon, and J. Revaud, "Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion," *arXiv preprint arXiv:2409.19152*, 2024.

[30] C. Ye, L. Qiu, X. Gu, Q. Zuo, Y. Wu, Z. Dong, L. Bo, Y. Xiu, and X. Han, "Stablenormal: Reducing diffusion variance for stable and sharp normal," *ACM Transactions on Graphics (TOG)*, vol. 43, no. 6, pp. 1–18, 2024.

[31] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth anything v2," *Advances in Neural Information Processing Systems*, vol. 37, pp. 21 875–21 911, 2024.

[32] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–13, 2017.