This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DroneSplat: 3D Gaussian Splatting for Robust 3D Reconstruction from In-the-Wild Drone Imagery

Jiadong Tang[1] Yu Gao[1] Dianyi Yang[1] Liqi Yan[2] Yufeng Yue[1] Yi Yang[1,*]
[1]Beijing Institute of Technology [2]Hangzhou Dianzi University

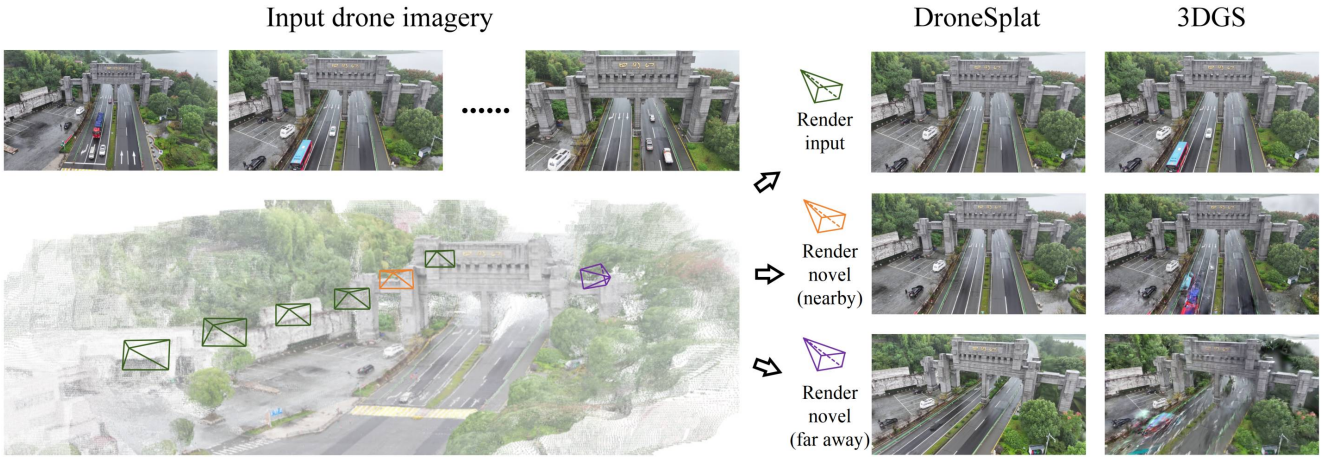


Figure 1. Given a set of drone imagery, our method effectively eliminates the impact of dynamic distractors on the static scenes (e.g., vehicles driving on the road). The right side of the figure shows the rendering results of our method compared to 3DGS under input view and novel view. Our method eliminates distractors while 3DGS generates artifacts in the corresponding regions. Moreover, our method reconstructs scenes with accurate geometry under limited viewpoints, demonstrating robustness to significant view variation (novel view).

Abstract

Drones have become essential tools for reconstructing wild scenes due to their outstanding maneuverability. Recent advances in radiance field methods have achieved remarkable rendering quality, providing a new avenue for 3D reconstruction from drone imagery. However, dynamic distractors in wild environments challenge the static scene assumption in radiance fields, while limited view constraints hinder the accurate capture of underlying scene geometry. To address these challenges, we introduce DroneSplat, a novel framework designed for robust 3D reconstruction from in-the-wild drone imagery. Our method adaptively adjusts masking thresholds by integrating local-global segmentation heuristics with statistical approaches, enabling precise identification and elimination of dynamic distractors in static scenes. We enhance 3D Gaussian Splatting with multi-view stereo predictions and a voxel-guided optimization strategy, supporting high-quality rendering under limited view constraints. For comprehensive evaluation, we provide a drone-captured 3D reconstruction dataset encompassing both dynamic and static scenes. Extensive experiments demonstrate that DroneSplat outperforms both 3DGS and NeRF baselines in handling in-the-wild drone imagery. Project page: https://bityia.github.io/DroneSplat/.

1. Introduction

3D reconstruction is pivotal in fields such as cultural heritage preservation [6], geological investigation [8] and urban surveying [19, 33]. In this context, drones have become valuable for these tasks due to their outstanding maneuverability. Capable of traversing obstacles like water and difficult terrain, drones enable extensive data acquisition from varied altitudes and angles.

Recently, radiance field methods, such as NeRF [23] and 3D Gaussian Splatting (3DGS) [11], have shown remarkable potential in 3D representation and novel view synthesis. Compared to traditional drone-based 3D reconstruction methods like oblique photogrammetry, NeRF and 3DGS can capture more realistic surface while reducing the train-

*Corresponding author: Yi Yang (yang_yi@bit.edu.cn).

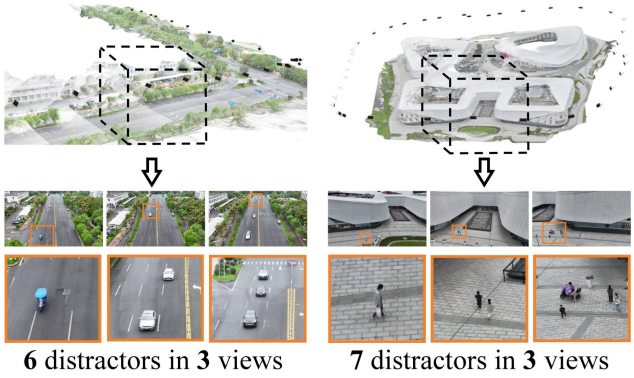


Figure 2. **The challenges within in-the-wild drone imagery.** Certain regions within the entire scene will face the challenge of dynamic distractors and limited view constraints.

ing time. However, applying NeRF or 3DGS to in-the-wild drone imagery presents several challenges for high-quality 3D reconstruction (Figure 2).

The primary challenge is *scene dynamics*. Drone-captured images in wild scenes often contain moving objects, or distractors, which violate the assumption of multi-view consistency in NeRF or 3DGS. To eliminate the impact of distractors, some works [28, 31–33] attempt to identify and mask specific moving objects by pre-trained semantic segmentation models. However, these methods rely on predefined categories of potential distractors and fail to distinguish static and dynamic objects of the same category, limiting their effectiveness in uncontrolled wild scenes. Other methods [1, 2, 12, 14, 16, 21, 29, 30, 45] utilize reconstruction residuals to predict distractors, which work well on controlled datasets, but struggle with smaller distractors in drone imagery, making accurate pixel-level mask prediction challenging. Additionally, existing approaches tend to apply hard thresholding to distinguish dynamic distractors, overlooking its limitations in adapting to variations across different scenes and training stages.

The second challenge is *viewpoint sparsity*. In a single flight, drone imagery oftens lacks comprehensive coverage of all viewpoints. For example, when flying over a street, the entire scene may include numerous images, but the viewpoints of certain regions remain limited (the black dashed box in Figure 2 is covered by only three views). Similar to challenges faced in sparse-view 3D reconstruction [15, 48], radiance fields may overfit to the inputs in limited view constraints, leading to poor rendering quality for novel views. Previous approaches attempt to address this by incorporating additional priors [4, 34, 37, 38, 47] or designing hand-crafted heuristics [24, 40, 41]. More recently, InstantSplat [5] propose a novel paradigm that initializes Gaussian primitives from multi-view stereo predictions to overcome sparsity. However, despite incorporating geometric priors, InstantSplat lacks corresponding optimization in 3DGS, undermining the abundant priors.

To address these challenges, we introduce DroneSplat, a robust 3D gaussian splatting framework tailored for in-the-wild drone imagery. For the issue of *scene dynamics*, we propose Adaptive Local-Global Masking, which integrates the strengths of segmentation heuristics and statistical approaches to enhance distractor identification and elimination. It can autonomously adjust the local masking threshold based on real-time residuals and pixel-level segmentation results, while tracking high residual candidates within the scene context as global masking. For the issue of *viewpoint sparsity*, our framework employs a multi-view stereo model to provide rich geometric priors by predicting dense 3D points. Based on these points, we implement a geometric-aware point sampling alongside a voxel-guided optimization strategy to effectively constrain 3DGS optimization.

Furthermore, to rigorously evaluate our approach in wild scenes, we provide a dataset of 24 drone-captured sequences, encompassing both dynamic and static scenes. The dynamic scenes are further categorized into three levels based on the number of dynamic distractors.

Our contributions are the following:

- We introduce a robust 3D Gaussian splatting framework tailored for in-the-wild drone imagery, which eliminates the impact of dynamic distractors on static scene reconstruction and addresses the issue of unconstrained optimization in 3DGS under limited viewpoints.
- We propose Adaptive Local-Global Masking that harmoniously integrates local-global segmentation heuristics and statistical approaches, combining local and global masking as collaborators to deliver robust and superior performance in dynamic distractor elimination.
- We enhance 3DGS with a geometric-aware point sampling method and a voxel-guided optimization strategy, effectively leveraging multi-view stereo's rich priors alongside 3DGS's powerful representational capacity.
- We provide a drone-captured 3D reconstruction dataset of 24 in-the-wild sequences, encompassing both dynamic and static scenes.

2. Related Work

Drone based 3D reconstruction. A typical method for 3D reconstruction with drones is oblique photography, where drones equipped with multiple cameras capture images from elevated angles. In traditional oblique photography, camera poses are estimated using a Structure-from-Motion (SfM) pipeline, followed by surface reconstruction through dense multi-view stereo [44]. Recently, radiance field methods [11, 23] have achieved realistic renderings. Several works [10, 22, 33, 44] utilize NeRF [23] to reconstruct scene from drone-captured images, and most prioritize the challenges associated with large-scale scene reconstruction.

DRAGON [7] optimizes a 3D Gaussian model [11] with drone imagery and ground building images. While previous work in radiance fields have advanced static scene reconstruction, these approaches encounter substantial challenges in Non-static environments.

Radiance fields in Non-static Scenes. There are roughly two approaches for reconstructing dynamic scenes: one aims to reconstruct the complete process of the scene changes from video sequences [9, 17, 42], while the other attempts to eliminate the influence of dynamic distractors to achieve static scene reconstruction. Our work aligns with the latter. Previous works in this realm have focused on how to accurately identify dynamic objects, or distractors, which can be roughly classified into two paradigms:

- Segmentation-based methods [1, 26, 28, 31–33] rely on pretrained semantic segmentation model to mask common dynamic objects like pedestrians. They require predefined classes for distractors, limiting their adaptability in real-world scenarios. Moreover, solely depending on semantic segmentation may lead to misclassification static objects as dynamic ones, such as parked cars along the roadside. Recent work NeRF-HuGS [1] enhances the traditional heuristic design by integrating the strengths of structure-from-motion and color residuals to obtain prompt points for SAM [13], thereby improving dynamic distractor segmentation. However, this approach requires manually tuning thresholds for each scene, and these threshold remain fixed during training, overlooking the residual variance at different training stages.
- Uncertainty-based methods [2, 12, 14, 16, 21, 29, 30, 45] employ color residuals and image feature to predict uncertainty (distractors), making them more adaptable in uncontrolled environments as they do not rely on priors of the scene. NeRF On-the-go [29] and WildGaussians [14] utilize DINOv2 [25] to extract features and design a network to predict the uncertainty. However, predicting pixel-level masks from low-resolution features results in blurred edge details and omission of small objects, posing challenges for accurately identification, especially in drone imagery where the distractors tend to be small.

Radiance fields under limited views. The performance of radiance fields substantially degrades when the input views become sparse. NeRF-based apporaches focus on incorporate regularization constraints [24, 40, 41] or introduce extra priors, such as depth estimation model [4, 34] and diffusion model [37, 38, 47]. 3DGS-based work [3, 15, 39, 48] typically integrate depth priors with hand-crafted heuristics. Recently, InstantSplat [5] has demonstrated notable achievement by leveraging the dense points predicted by DUSt3R [35] as geometric priors to initialize Gaussian primitives. However, Instantsplat lacks constraints for subsequent Gaussian optimization, leading to suboptimal performance despite the presence of rich priors.

3. Method

3.1. Preliminaries

We build our work upon 3DGS [11], which represents a 3D scene explicitly through a collection of 3D anisotropic Gaussian primitives. Specifically, a Gaussian primitive can be defined by the following set of parameters: a center point μ, a scaling matrix S, a rotation matrix R, an opacity parameter α and a color feature f. The influence of each Gaussian on a 3D point x can be described:

$$G(x) = e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \tag{1}$$

where Σ is a positive semi-definite covariance matrix which can be calculated from the scale matrix S and the rotation matrix R. To render 2D image, 3DGS first project Gaussian primitives onto the 2D plane, and then blending N ordered Gaussians overlapping each pixel to compute the color:

$$c = \sum_{i=1}^{N} c_i \tilde{\alpha}_i \prod_{j=1}^{i-1}(1-\tilde{\alpha}_j) \tag{2}$$

where c_i is the decoded color of feature f, $\tilde{\alpha}_i$ represents the opacity of the projected 2D Gaussians.

3DGS is optimized by a combination of D-SSIM [36] and L1 loss computed from the rendered color and the ground truth color:

$$\mathcal{L} = (1-\lambda_{dssim})\mathcal{L}_{\text{L1}} + \lambda_{dssim}\mathcal{L}_{\text{D-SSIM}} \tag{3}$$

where λ_{dssim} is a weighting factor.

3.2. Adaptive Local-Global Masking

The predominant approach for eliminating distractors in static scene reconstruction is to identify masks $\mathcal{M}$ for dynamic distractors and incorporate them to Eq. 3. The training loss can be defined as follows:

$$\mathcal{L} = (1-\lambda_{dssim})\mathcal{M}\mathcal{L}_{\text{L1}} + \lambda_{dssim}\mathcal{M}\mathcal{L}_{\text{D-SSIM}} \tag{4}$$

The training loss (Eq. 4) indicates that higher mask accuracy leads to fewer artifacts and improved reconstruction quality.

Object-wise Average of Normalized Residuals. Inspired by RobustNeRF [30], we start with the residuals between the rendered image and the ground truth. Note that all residuals mentioned later are calculated independently and are not involved in the loss function. We normalize the DSSIM residual $\mathcal{R}_{\text{D-SSIM}}$ and L1 residual $\mathcal{R}_{\text{L1}}$ to $\tilde{\mathcal{R}}_{\text{D-SSIM}}$ and $\tilde{\mathcal{R}}_{\text{L1}}$, respectively, mitigating the impact of residual variations that could affect distractor identification. Then we can obtain the combined residuals $\tilde{\mathcal{R}}$ as:

$$\tilde{\mathcal{R}} = (1-\lambda_{dssim})\tilde{\mathcal{R}}_{\text{L1}} + \lambda_{dssim}\tilde{\mathcal{R}}_{\text{D-SSIM}} \tag{5}$$

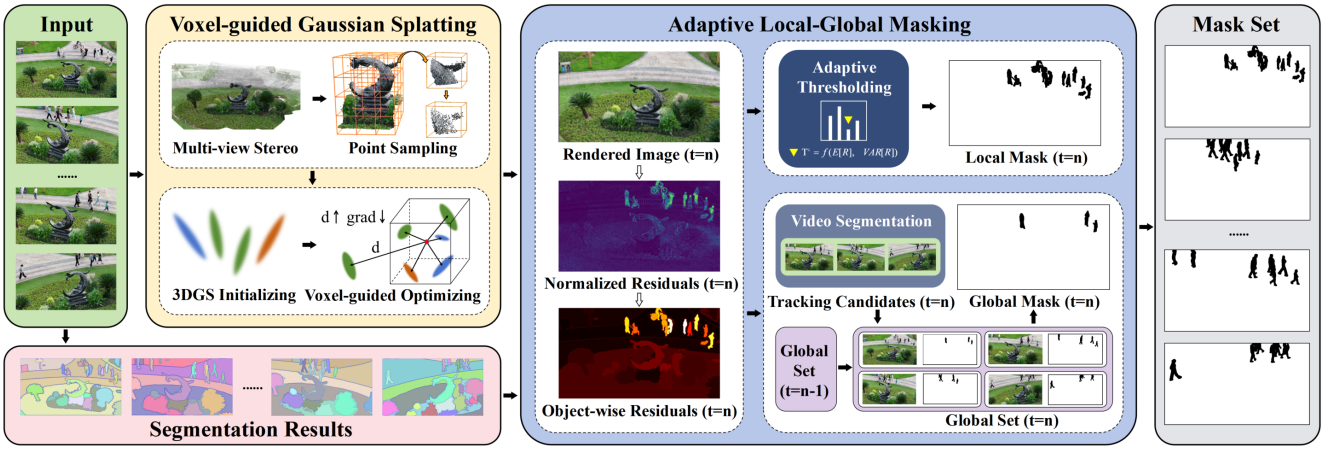


Figure 3. **The framework of DroneSplat.** Given a few posed drone imagery of a wild scene, our goal is to identify and eliminate dynamic distractors. We first predict a dense point cloud through a learning-based multi-view stereo method, followed by point sampling based on confidence and geometric features. The sampled point cloud is used to initialize Gaussian primitives, which is then optimized using a voxel-guided strategy. At iteration $t = n$, we calculate the normalized residual of the rendered image and combine it with the segmentation results to obtain the object-wise residuals. We adaptively adjust the threshold based on the object-wise residuals and statistical approaches to obtain local masks. Meanwhile, we mark objects with high residuals as tracking candidates, deriving the global set at $t = n$ by combining the global set at $t = n - 1$ with the tracking outcomes at $t = n$. After merging the local mask and the global mask retrieved from the global set, we can obtain the final mask at time $t = n$. The mask set illustrates the dynamic distractors we predicted.

To achieve pixel-level mask precision without reliance on predefined classes, we employ the latest Segment Anything Model v2 [27]. For each image I_i in the training set I, we use the segmentation model S to obtain $S(I_i) = \{m_i^1, m_i^2, \cdots, m_i^{N_i}\}$, where m_i^j represents the segmentation mask of the j-th object in image I_i and N_i denotes the number of masks of image I_i.

We assume that all pixels of the same object are treated jointly, sharing a unified residual value. The object-wise average residual $\mathcal{R}_i^j$ for a mask m_i^j can be calculated as:

$$\mathcal{R}_i^j(t) = \frac{\sum_{p \in m_i^j} \tilde{\mathcal{R}}(p,t)}{|m_i^j|} \tag{6}$$

where $p \in m_i^j$ indicates that p is one of the pixel positions contained within the mask m_i^j, t is the current iteration and $|m_i^j|$ is the area (number of pixels) of mask m_i^j.

Adaptive Local Masking. Since dynamic objects are more difficult to fit and often exhibit higher residuals, a straightforward approach is to set a threshold to identify distractors. However, this process can be tedious: a higher threshold may fail to remove large dynamic objects, while a lower threshold risks misclassifying static objects as distractors [1]. Previous studies [1, 26, 30] have not fully addressed this issue, instead applying a fixed threshold throughout training.

To establish an accurate and appropriate threshold across different scenarios and training stages, we propose an adaptive method to adjust threshold based on real-time residuals and statistical approaches. For the current training frame, we calculate the mathematical expectation and variance of the pixel-wise average residual:

$$\mathbb{E}[R_i](t) = \sum_{k=1}^{N} p_k \cdot r_k(t) \tag{7}$$

$$\mathrm{Var}[R_i](t) = \frac{1}{N} \sum_{k=1}^{N} (r_k(t) - \mathbb{E}[R_i](t))^2 \tag{8}$$

where N denotes the number of pixels in the current frame, t is the current iteration, $r_k(t) \in R_i(t)$ is residual of the k-th pixel, and p_k represents the probability of the k-th pixel, typically $p_k = \frac{1}{N}$ if all pixels are equally weighted.

It can be observed that, across various scenes and training stages, nearly all static objects fall within one standard deviation of the expectation. In other words, pixels with residuals exceeding the expectation plus the variance can be regarded as distractors. Considering the varying convergence rates of residuals across objects in the early stage of training, we relax the upper limit of the threshold. We set $\mathcal{T}_i^L$ as local threshold for image I_i:

$$\mathcal{T}_i^L(t) = \mathbb{E}[R_i](t) + \mathrm{Var}[R_i](t)(1 + \lambda_L \frac{T_{max} - t}{T_{max}}) \tag{9}$$

where T_{max} is the maximum iteration, t is the current iteration and λ_L is a weighting factor.

Let $\mathcal{M}_i^L(t)$ denote the set of local masks m_i^j in image I_i at iteration t whose object-wise average residual $\mathcal{R}_i^j(t)$ exceeds the threshold $\mathcal{T}_i^L(t)$:

$$\mathcal{M}_i^L(t) = \left\{ m_i^j \mid \mathcal{R}_i^j(t) > \mathcal{T}_i^L(t),\ j \in N_i \right\} \tag{10}$$

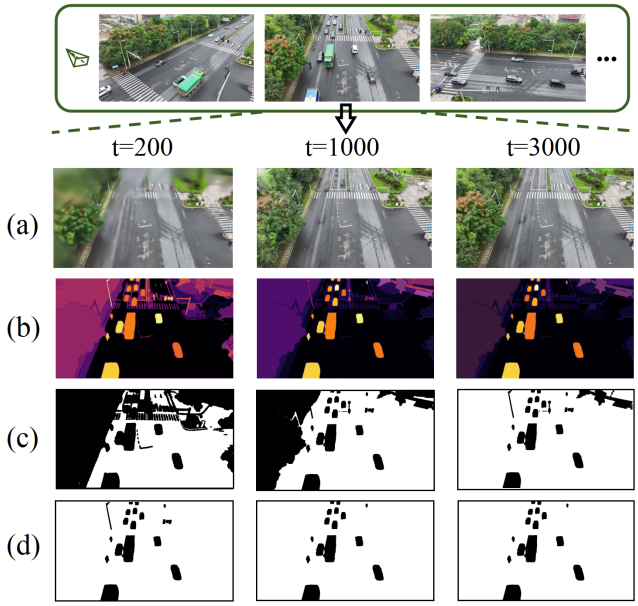


Figure 4. **The effect of Adaptive Local Masking.** (a) represent the renderings of the same frame across different iterations t, and (b) show the corresponding object-wise residuals. (c) are the masks obtained using a hard threshold, while (d) are the masks obtained by Adaptive Local Masking.

where N_i is the total number of masks (objects) in image I_i, and $\mathcal{R}_i^j(t)$ represents the object-wise average residual for mask m_i^j at iteration t. Figure 4 illustrates the effect of Adaptive Local Masking.

Complement Global Masking. While Adaptive Local Masking is effective in identifying distractors in each frame, it still has certain limitations when handling some specific scenes. For example, an object may remain stationary in several frames but still should be considered a distractor in the overall context, such as a moving vehicle stops at a red light at an intersection (Figure 5).

To address this, we employ the video segmentation capabilities of Segment Anything Model v2 [27]. We set $\mathcal{T}_i^G$ as global threshold for image I_i:

$$\mathcal{T}_i^G = \mathbb{E}[R_i](t) + \lambda_G \text{Var}[R_i](t) \tag{11}$$

where λ_G is a weighting factor which is larger than $1 + \lambda_L$.

The set of global masks $\mathcal{M}_i^{\text{G}}$ of image I_i is empty at start:

$$\mathcal{M}_i^G(0) = \{\emptyset\} \tag{12}$$

Once the residual $R_k^j(t)$ of m_k^j exceeding the threshold T_k^G, we mark it as a tracking candidate. Specifically, we select a center point and four edge points of m_k^j as point prompts, which are then input into Segment Anything Model v2 to initiate tracking. This allows us to obtain the corresponding masks of m_k^j in each frame:

$$\mathcal{M}_i^G(t) = \mathcal{M}_i^G(t-1) \cup \hat{m}_i^j \tag{13}$$

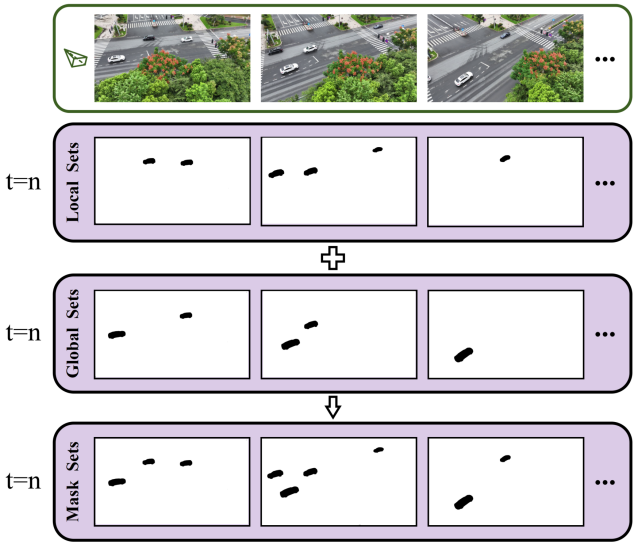


Figure 5. **The effect of Complement Global Masking.** At $t = n$, the white car waiting at a red light is not identified by the Adaptive Local Masking, but it is tracked through Complement Global Masking in other frames.

where $\hat{m}_i^j$ denotes the tracking results of m_k^j in frame I_i.

The final mask of image I_i at iteration t is:

$$\mathcal{M}_i(t) = \mathcal{M}_i^L(t) \cup \mathcal{M}_i^G(t) \tag{14}$$

3.3. Voxel-guided Gaussian Splatting

To reconstruct a 3D model with accurate geometry in regions with limited view constraints, we utilize a learning-based multi-view stereo method, DUSt3R [35], to obtain rich geometric priors. To integrate these priors within the 3DGS, we design a geometric-aware point sampling method and a voxel-guided Gaussian optimization strategy.

Geometric-aware Point Sampling. Given two unposed images, DUSt3R excels at predicting pairwise point maps and confidence maps, as well as recovering camera parameters. When processing more than two images, DUSt3R applies a point cloud alignment as post-processing to generate a dense point cloud. An intuitive idea is to use these dense points to initialize Gaussian primitives. However, the over-parameterized primitives, some of which exhibit positional deviations, make it challenging for 3DGS to achieve effective optimization. To address this issue, we sample dense points based on both confidence and geometric features, ensuring that the sampled points are well-suited for 3DGS initialization while preserving geometric priors.

We employ the Fast Point Feature Histogram(FPFH) Descriptor to extract geometric features from the dense points:

$$\text{FPFH}(p) = \text{SPFH}(p) + \frac{1}{K} \sum_{q \in \mathcal{N}(p)} \frac{1}{d(p,q)} \cdot \text{SPFH}(q) \tag{15}$$

where SPFH(p) denotes the Simplified Point Feature Histogram for point p, $\mathcal{N}(p)$ is the set of neighboring points surrounding point p while q is a neighbor point within the $\mathcal{N}(p)$. $d(p,q)$ is the Euclidean distance between points p and q and K is the number of neighbors considered.

The space occupied by the dense points is then divided into N voxels, with the voxel size adaptively determined by the scene's dimensions. In particular,the scene is enclosed within a rectangular cuboid, and the shortest edge of this cuboid is divided into N voxels. For each voxel, we calculate the scores for all points within it:

$$\text{Score}(p) = Conf(p) \cdot \tilde{\text{FPFH}}(p) \tag{16}$$

where $Conf(p)$ denotes the confidence of the point p and $\tilde{\text{FPFH}}(p)$ is the normalized FPFH feature of the point p.

In each voxel, the top k points with the highest scores are retained to initialize Gaussian primitives.

Voxel-guided Optimization. Although these sampled points provide valuable geometric priors for 3DGS, the original optimization strategy could undermine these advantages. Therefore, we additionally incorporate a voxel-guided optimization strategy to direct the optimization of 3DGS under limited view constraints.

For each initialized Gaussian within a voxel, as well as those split or cloned from it, strict constraints are imposed to keep them within the defined bounds of the voxel. Specifically, if a Gaussian g_i^j belonging to voxel V_i has its center or scale exceed defined limit, typically set to τ times the voxel length, it will be flagged as an unconstrained Gaussian. The gradient for such Gaussians decays exponentially with increasing distance from the voxel center. If the accumulated gradient of an unconstrained Gaussian reaches a preset threshold γ_1 and directs toward an empty voxel, it will be split or cloned, and the newly generated Gaussian will be assigned to that empty voxel.

To eliminate trivial voxels, we remove a voxel if it contains fewer Gaussians than a specified threshold γ_2 or if the average opacity falls below an acceptable level γ_3.

4. Experiments

4.1. Setups

DroneSplat Dataset. To rigorously evaluate our method in wild scenes, we introduce a novel drone-captured 3D reconstruction dataset. The dataset includes 24 in-the-wild sequences encompassing both dynamic and static scenes, and dynamic scenes containing varying numbers of dynamic distractors. For the experiments of dynamic distractor elimination, we select 6 scenes representing different levels of dynamics, each containing sufficient images to avoid the issue of sparsity; for limited view reconstruction, we select 2 static scenes with only 6 input views.

On-the-go Dataset. The On-the-go dataset [29] includes multiple casually captured scenes with varying ratios of occlusions. Following the baseline settings, we select 6 representative scenes for the distractor elimination experiments.

UrbanScene3D Dataset. UrbanScene3D [18] is a large-scale dataset designed for urban scene perception and reconstruction, including both synthetic data and real scenes captured by drones. We select two real scenes with six input views for limited view reconstruction experiments.

Baselines. For the experiments of distractor elimination, we compare our approach to three NeRF-based methods (RobustNeRF [30], NeRF-HuGS [1] and NeRF On-the-go [29]) and two 3DGS-based methods (GS-W [45] and WildGaussians [14]). In addition, we also include two well-known baselines, the vanilla 3DGS [11] and Mip-Splatting [43].

For the experiments of limited view reconstruction, we compare two NeRF-based methods (FreeNeRF [40] and DiffusioNeRF [38]) and three 3DGS-based methods (FSGS [48], DNGaussian [15] and InstantSplat [5]). Moreover, vanilla 3DGS [11] and a view-adaptive method, Scaffold-GS [20], are also included for comparison.

Implementations. All baselines are trained following their own default settings. For our method, we train the model with 7,000 iterations and all results are obtained using a NVIDIA A100 GPU. In the experiments of dynamic distractors elimination, we activate Adaptive Local-Global Masking at 500 iterations, and we set λ_L to 0.4 while λ_G to 2.8. For geometric-aware point sampling, a resolution of 512 is used for DUSt3R to predict dense points and we set N to 80 with k to 3. In Voxel-guided optimization, we set τ to 3.5, γ_1 to 0.003, γ_1 to 2 and γ_3 to 0.075.

Metrics. We adopt the widely used PSNR, SSIM [36] and LPIPS [46] to evaluate the novel view synthesis qualities.

4.2. Comparison

Distractor Elimination. Both the baselines and our method are trained on images with dynamic distractors and evaluated on images without distractors.

To further validate the effectiveness of Adaptive Local-Global Masking, we introduce a control group, denoted as *Ours (COLMAP)*. This group utilizes sparse point clouds generated by COLMAP for Gaussian initialization, ensuring consistency with other baseline methods, and employs the vanilla 3DGS for subsequent optimization.

As shown in Figure 6 and Figure 7, our approach outperforms all baseline method on both DroneSplat(dynamic) datatset and NeRF On-the-go dataset. In comparison, GS-W and WildGaussians struggle to eliminate the impact of distractors, leading to artifacts and floaters. While Robust-NeRF and NeRF On-the-go successfully remove distractors, they fail to retain fine details. Our method achieves the highest quantitative results, effectively eliminating dynamic distractors while preserving static details.

Method	Low Dynamic						Medium Dynamic						High Dynamic					
	Cultural Center			TangTian			Pavilion			Simingshan			Intersection			Sculpture		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
RobustNeRF [CVPR'23]	16.63	0.538	0.541	16.46	0.335	0.589	15.67	0.317	0.496	18.716	0.549	0.441	16.42	0.589	0.444	17.01	0.243	0.602
NeRF-HuGS [CVPR'24]	22.02	0.732	0.159	18.01	0.399	0.448	15.90	0.357	0.361	22.64	0.727	0.187	16.98	0.556	0.407	16.88	0.251	0.459
NeRF On-the-go [CVPR'24]	19.53	0.658	0.331	17.59	0.359	0.527	15.58	0.318	0.471	18.128	0.488	0.487	15.76	0.535	0.479	17.04	0.228	0.606
3DGS [SIGGRAPH'23]	22.43	0.730	0.152	17.31	0.430	0.384	17.04	0.438	0.228	22.14	0.749	0.127	18.11	0.570	0.317	17.11	0.396	0.256
Mip-Splatting [CVPR'24]	22.11	0.727	0.175	16.87	0.436	0.387	17.26	0.441	0.210	21.78	0.742	0.137	17.49	0.553	0.346	17.22	0.404	0.249
GS-W [ECCV'24]	23.29	0.759	0.151	18.55	0.419	0.528	17.44	0.416	0.424	22.41	0.706	0.243	19.06	0.624	0.382	17.09	0.414	0.270
WildGaussians [NIPS'24]	22.41	0.734	0.190	17.45	0.395	0.533	16.96	0.388	0.379	22.63	0.707	0.239	17.22	0.551	0.421	17.15	0.371	0.383
Ours(COLMAP)	24.56	0.773	0.142	19.53	0.459	0.365	17.89	0.462	0.217	23.91	0.792	0.098	19.47	0.681	0.304	19.51	0.528	0.197
Ours	24.53	0.785	0.149	19.68	0.472	0.331	17.79	0.469	0.214	24.17	0.808	0.093	19.86	0.657	0.322	19.64	0.517	0.192

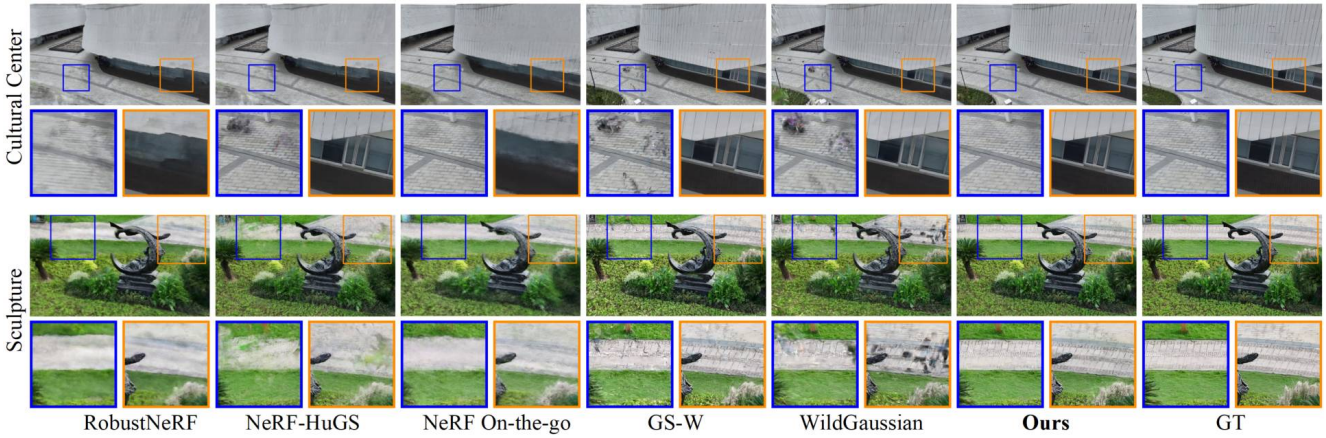


Figure 6. **Quantitative and qualitative results on DronSplat dataset (dynamic).** The 1st, 2nd and 3rd best results are highlighted. Our method outperforms baseline methods on scenes with various numbers of dynamic distractors, while *Ours(COLMAP)* leading the rest. Note that all scenes have a sufficient number of viewpoints, avoiding the issue of sparsity.

Method	Low Occlusion						Medium Occlusion						High Occlusion					
	Mountain			Fountain			Corner			Patio			Spot			Patio-High		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
RobustNeRF [CVPR'23]	17.38	0.402	0.523	15.64	0.322	0.561	23.12	0.741	0.186	20.41	0.685	0.197	20.46	0.488	0.458	20.55	0.426	0.407
NeRF-HuGS [CVPR'24]	20.21	0.659	0.160	20.45	0.663	0.167	23.07	0.748	0.191	17.56	0.611	0.261	20.33	0.540	0.361	16.65	0.479	0.397
NeRF On-the-go [CVPR'24]	20.07	0.635	0.286	20.23	0.624	0.297	24.02	0.803	0.184	20.65	0.723	0.224	23.31	0.762	0.198	21.44	0.709	0.258
3DGS [SIGGRAPH'23]	19.17	0.637	0.176	19.94	0.675	0.172	20.80	0.697	0.211	16.74	0.649	0.201	17.38	0.573	0.426	16.78	0.564	0.346
Mip-Splatting [CVPR'24]	18.83	0.616	0.195	19.92	0.677	0.169	20.26	0.657	0.238	16.19	0.635	0.232	15.59	0.499	0.535	16.47	0.539	0.397
GS-W [ECCV'24]	19.92	0.560	0.291	20.19	0.589	0.279	23.72	0.785	0.132	19.10	0.691	0.163	22.42	0.635	0.309	21.21	0.649	0.260
WildGaussians [NIPS'24]	20.43	0.652	0.213	20.83	0.671	0.185	24.19	0.816	0.105	21.48	0.806	0.111	23.76	0.794	0.081	22.27	0.732	0.165
Ours(COLMAP)	21.23	0.687	0.162	21.54	0.705	0.168	24.77	0.823	0.106	21.85	0.816	0.107	24.37	0.821	0.095	22.53	0.778	0.181
Ours	21.45	0.694	0.158	21.60	0.699	0.165	24.65	0.814	0.098	21.88	0.812	0.106	24.44	0.827	0.094	22.60	0.792	0.177

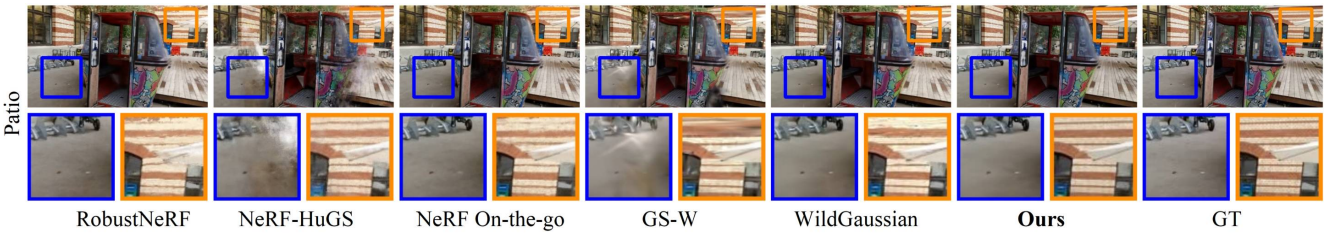


Figure 7. **Quantitative and qualitative results on NeRF On-the-go dataset.** The 1st, 2nd and 3rd best results are highlighted. Our method not only effectively eliminates dynamic distractors but also reconstructs fine details in the scene.

Limited-view Reconstruction. As shown in Figure 8 and Figure 9, our method demonstrates competitive results on both DroneSplat(static) dataset and UrbanScene3D dataset. NeRF-based methods struggle to recover accurate scene geometry. While 3DGS-based methods capture the overall shape, they still generate artifacts due to unconstrained optimization in areas with limited viewpoint overlap (the patches in orange frames). Despite not specifically designed to address the issue of limited view, Scaffold-GS demonstrates robustness due to its view-adaptive capability. However, when the novel view significantly differs from the input views, the results are still suboptimal.

Method	Exhibition Hall			Plaza		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
FreeNeRF [CVPR'23]	11.53	0.244	0.683	12.30	0.273	0.706
DiffusioNeRF [CVPR'23]	11.82	0.267	0.656	12.41	0.308	0.695
3DGS [SIGGRAPH'23]	15.64	0.404	0.374	16.45	0.438	0.343
Scaffold-GS [CVPR'24]	16.55	0.445	0.343	17.01	0.461	0.323
FSGS [ECCV'24]	15.92	0.424	0.411	16.41	0.469	0.378
DNGaussian [CVPR'24]	12.61	0.281	0.558	12.41	0.222	0.595
InstantSplat	17.37	0.393	0.398	16.61	0.384	0.410
Ours	18.09	0.496	0.352	18.17	0.534	0.335

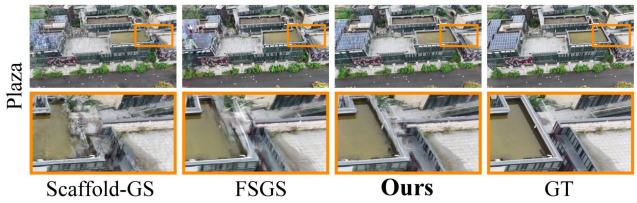


Figure 8. **Quantitative and qualitative results on DroneSplat dataset(static).** The 1st, 2nd and 3rd best results are highlighted. Our method reconstructs the static scenes with correct geometry even under limited view constraints.

Method	PolyTech			ArtSci		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
FreeNeRF [CVPR'23]	11.56	0.165	0.692	13.42	0.264	0.578
DiffusioNeRF [CVPR'23]	10.97	0.142	0.769	18.23	0.627	0.235
3DGS [SIGGRAPH'23]	14.05	0.424	0.279	19.01	0.595	0.201
Scaffold-GS [CVPR'24]	14.61	0.479	0.253	19.69	0.618	0.183
FSGS [ECCV'24]	14.82	0.473	0.307	20.32	0.651	0.240
DNGaussian [CVPR'24]	12.99	0.347	0.297	15.43	0.439	0.382
InstantSplat	14.51	0.365	0.337	16.16	0.325	0.364
Ours	15.62	0.516	0.288	20.47	0.639	0.215



Figure 9. **Quantitative and qualitative results on UrbanScene3D dataset.** The 1st, 2nd and 3rd best results are highlighted. Our approach shows competitive results compared to state-of-the-art sparse-view reconstruction methods.

4.3. Ablation Study

We divide our method into three modules to analyze their contributions. As shown in Figure 10, Adaptive Local Masking effectively identifies and eliminates distractors in dynamic scenes (b) while Complementary Global Masking serves as an additional check for any missed distractors (c). Incorporating Voxel-guided 3DGS further enhances performance (d). This improvement is attributed to our voxel-guided optimization, which constrains the spread of distractors and removes voxels with a high proportion of distractors by calculating the average opacity within each voxel.

To further study the Voxel-guided 3DGS, we remove different components to verify their effect. As shown in Figure 11, each component plays a critical role. Notably, despite leveraging geometric priors from multi-view stereo, the reconstruction quality does not improve as expected when using vanilla 3DGS, indicating the vanilla 3DGS struggles to optimize the over-parameterization initial primitives (b). The sampled points retain the prior of scene geometry, yet subsequent Gaussian optimization undermines the prior (c). The complete method that combines the prior and voxel-guided optimization strategy achieves the best results (d).

	Local Masking	Global Masking	Voxel-guided 3DGS	DroneSplat(dynamic)			NeRF On-the-go		
				PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
(a)	✗	✗	✗	19.02	0.552	0.244	18.46	0.632	0.255
(b)	✓	✗	✗	20.71	0.609	0.223	22.64	0.764	0.140
(c)	✓	✓	✗	20.74	0.616	0.221	22.71	0.772	0.136
(d)	✓	✓	✓	20.86	0.618	0.217	22.77	0.773	0.133

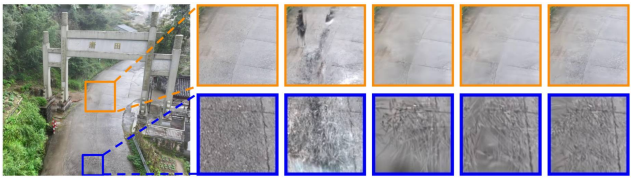


Figure 10. **The ablations of our method on the DroneSplat(dynamic) and NeRF On-the-go dataset.** The 1st, 2nd and 3rd best results are highlighted.

	Geometric Priors	Point Sampling	Gaussian Optimization	DroneSplat(static)			UrbanScene3D		
				PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
(a)	✗	✗	✗	16.01	0.421	0.358	16.53	0.510	0.241
(b)	✓	✗	✗	16.99	0.388	0.404	15.33	0.345	0.351
(c)	✓	✓	✗	17.46	0.432	0.359	16.97	0.536	0.289
(d)	✓	✓	✓	18.13	0.515	0.343	18.05	0.578	0.252

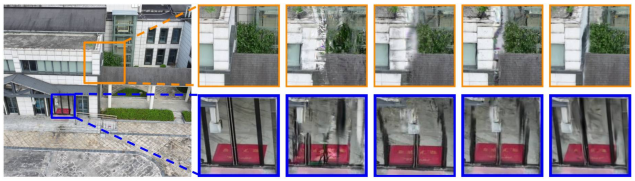


Figure 11. **The ablations of Voxel-guided 3DGS on the DroneSplat(static) and UrbanScene3D dataset.** The 1st, 2nd and 3rd best results are highlighted.

5. Conclusions

We present DroneSplat, a novel framework for robust 3D reconstruction from in-the-wild drone imagery. We integrate local-global segmentation heuristics and statistical approaches to precisely identify and eliminate dynamic distractors. Furthermore, We enhance 3D Gaussian Splatting through multi-view stereo predictions and a voxel-guided optimization strategy, enabling accurate scene reconstruction under limited view constraints. Experimental evaluations across diverse datasets demonstrate the superiority and robustness of our approach over previous methods.

Acknowledgements

This work was partly supported by National Natural Science Foundation of China (Grant No. NSFC 62233002) and National Key R&D Program of China (2022YFC2603600). The authors would like to thank Tianji Jiang, Xihan Wang, and all other members of ININ Lab of Beijing Institute of Technology for their contribution to this work.

References

[1] Jiahao Chen, Yipeng Qin, Lingjie Liu, Jiangbo Lu, and Guanbin 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*, pages 19436–19446, 2024. 2, 3, 4, 6

[2] Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12943–12952, 2022. 2, 3

[3] Jaeyoung Chung, Jeongtaek Oh, and Kyoung Mu Lee. Depth-regularized optimization for 3d gaussian splatting in few-shot images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 811–820, 2024. 3

[4] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022. 2, 3

[5] Zhiwen Fan, Wenyan Cong, Kairun Wen, Kevin Wang, Jian Zhang, Xinghao Ding, Danfei Xu, Boris Ivanovic, Marco Pavone, Georgios Pavlakos, et al. Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds. *arXiv preprint arXiv:2403.20309*, 2024. 2, 3, 6

[6] Leonardo Gomes, Olga Regina Pereira Bellon, and Luciano Silva. 3d reconstruction methods for digital preservation of cultural heritage: A survey. *Pattern Recognition Letters*, 50: 3–14, 2014. Depth Image Analysis. 1

[7] Yujin Ham, Mateusz Michalkiewicz, and Guha Balakrishnan. Dragon: Drone and ground gaussian splatting for 3d building reconstruction. In *2024 IEEE International Conference on Computational Photography (ICCP)*, pages 1–12. IEEE, 2024. 3

[8] Jianning Hao, Xiuli Zhang, Chengtang Wang, Hao Wang, and Haibin Wang. Application of uav digital photogrammetry in geological investigation and stability evaluation of high-steep mine rock slope. *Drones*, 7(3), 2023. 1

[9] Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4220–4230, 2024. 3

[10] Zhihao Jia, Bing Wang, and Changhao Chen. Drone-nerf: Efficient nerf based 3d scene reconstruction for large-scale drone survey. *Image and Vision Computing*, 143:104920, 2024. 2

[11] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 1, 2, 3, 6

[12] Injae Kim, Minhyuk Choi, and Hyunwoo J Kim. Up-nerf: Unconstrained pose prior-free neural radiance field. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3

[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. 3

[14] Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024. 2, 3, 6

[15] Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. 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*, pages 20775–20785, 2024. 2, 3, 6

[16] Peihao Li, Shaohui Wang, Chen Yang, Bingbing Liu, Weichao Qiu, and Haoqian Wang. Nerf-ms: Neural radiance fields with multi-sequence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18591–18600, 2023. 2, 3

[17] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8508–8520, 2024. 3

[18] Liqiang Lin, Yilin Liu, Yue Hu, Xingguang Yan, Ke Xie, and Hui Huang. Capturing, reconstructing, and simulating: the urbanscene3d dataset. In *European Conference on Computer Vision*, pages 93–109. Springer, 2022. 6

[19] Yang Liu, Chuanchen Luo, Lue Fan, Naiyan Wang, Junran Peng, and Zhaoxiang Zhang. Citygaussian: Real-time high-quality large-scale scene rendering with gaussians. In *European Conference on Computer Vision*, pages 265–282. Springer, 2025. 1

[20] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20654–20664, 2024. 6

[21] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7210–7219, 2021. 2, 3

[22] Christopher Maxey, Jaehoon Choi, Hyungtae Lee, Dinesh Manocha, and Heesung Kwon. Uav-sim: Nerf-based synthetic data generation for uav-based perception. In *2024*

IEEE International Conference on Robotics and Automation (ICRA), pages 5323–5329. IEEE, 2024. 2

[23] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Computer Vision – ECCV 2020*, pages 405–421. Springer International Publishing, 2020. 1, 2

[24] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5480–5490, 2022. 2, 3

[25] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3

[26] Takashi Otonari, Satoshi Ikehata, and Kiyoharu Aizawa. Entity-nerf: Detecting and removing moving entities in urban scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20892–20901, 2024. 3, 4

[27] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 4, 5

[28] Konstantinos Rematas, Andrew Liu, Pratul P Srinivasan, Jonathan T Barron, Andrea Tagliasacchi, Thomas Funkhouser, and Vittorio Ferrari. Urban radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12932–12942, 2022. 2, 3

[29] Weining Ren, Zihan Zhu, Boyang Sun, Jiaqi Chen, Marc Pollefeys, and Songyou 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*, pages 8931–8940, 2024. 2, 3, 6

[30] Sara Sabour, Suhani Vora, Daniel Duckworth, Ivan Krasin, David J Fleet, and Andrea Tagliasacchi. Robustnerf: Ignoring distractors with robust losses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20626–20636, 2023. 2, 3, 4, 6

[31] Jiaming Sun, Xi Chen, Qianqian Wang, Zhengqi Li, Hadar Averbuch-Elor, Xiaowei Zhou, and Noah Snavely. Neural 3d reconstruction in the wild. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–9, 2022. 2, 3

[32] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8258, 2022.

[33] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12922–12931, 2022. 1, 2, 3

[34] Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9065–9076, 2023. 2, 3

[35] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 3, 5

[36] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 3, 6

[37] Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor Verbin, Jonathan T Barron, Ben Poole, et al. Reconfusion: 3d reconstruction with diffusion priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21551–21561, 2024. 2, 3

[38] Jamie Wynn and Daniyar Turmukhambetov. Diffusionerf: Regularizing neural radiance fields with denoising diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4180–4189, 2023. 2, 3, 6

[39] Haolin Xiong, Sairisheek Muttukuru, Rishi Upadhyay, Pradyumna Chari, and Achuta Kadambi. Sparsegs: Real-time 360 {\deg} sparse view synthesis using gaussian splatting. *arXiv preprint arXiv:2312.00206*, 2023. 3

[40] Jiawei Yang, Marco Pavone, and Yue Wang. Freenerf: Improving few-shot neural rendering with free frequency regularization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8254–8263, 2023. 2, 3, 6

[41] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4578–4587, 2021. 2, 3

[42] Heng Yu, Joel Julin, Zoltán Á Milacski, Koichiro Niinuma, and László A Jeni. Cogs: Controllable gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21624–21633, 2024. 3

[43] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19447–19456, 2024. 6

[44] Xiaoyi Zeng, Kaiwen Song, Leyuan Yang, Bailin Deng, and Juyong Zhang. Oblique-merf: Revisiting and improving merf for oblique photography. *arXiv preprint arXiv:2404.09531*, 2024. 2

[45] Dongbin Zhang, Chuming Wang, Weitao Wang, Peihao Li, Minghan Qin, and Haoqian Wang. Gaussian in the wild: 3d gaussian splatting for unconstrained image collections. *arXiv preprint arXiv:2403.15704*, 2024. 2, 3, 6

[46] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[47] Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12588–12597, 2023. 2, 3

[48] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. In *European Conference on Computer Vision*, pages 145–163. Springer, 2025. 2, 3, 6