StratoSplat: Taming Layered Regularities for Sparse Aerial 3D Gaussian Splatting

Zihan Gao, Lingling Li*, Licheng Jiao, Fang Liu, Wenping Ma, Yuwei Guo, and Shuyuan Yang

School of Artificial Intelligence, Xidian University, Xi'an, China
z1han_gao@163.com, llli@xidian.edu.cn

Abstract. 3D Gaussian Splatting has revolutionized novel view synthesis with real-time rendering and photorealistic quality. However, standard 3DGS methods fail in aerial scenes due to sparse anisotropic UAV captures. Two critical issues emerge: incomplete triangulated points in textureless regions that destabilizes Gaussian initialization and view-aligned degeneracies where Gaussians overfit limited observations rather than respect true geometry. We present StratoSplat, a framework that exploits the inherent layered structure of aerial scenes for robust sparse-view 3DGS. Our key insight is that buildings, terrain, and roads naturally stratify along gravity, providing strong geometric priors to regularize this problem. We address point sparsity through layered memory guided initialization that enforces multi-view consistency. To prevent optimization degeneracies, we introduce neural multi-plane Gaussians where virtual primitives anchor to learned parallel planes for geometric regularization. Extensive experiments demonstrate state-of-the-art performance, surpassing the best baseline by over 3dB. Code available at https://github.com/keloee/StratoSplat.

Keywords: Aerial Photogrammetry · Sparse Aerial View Synthesis · Gaussian Splatting

1 Introduction

3D Gaussian Splatting (3DGS) [25] has emerged as one of the most promising solutions for novel view synthesis. It has drawn wide attention in aerial scenes due to its real-time rendering speed and ability to capture fine-grained scene details [13, 47]. Building upon 3DGS, numerous applications in aerial scenes have been developed, including urban planning, environmental monitoring, autonomous navigation, and large-scale 3D mapping [4, 21, 28, 61].

However, unmanned aerial vehicles (UAVs) may face constraints on flight time, viewing geometry, or weather, which hinder dense multi-view capture [18, 19, 48]. 3DGS, similar to other radiance field representations, suffers from degradation when trained with sparse input views [29, 64]. This vulnerability limits its use in real-world aerial scenes where dense acquisition maybe costly or infeasible, and motivates methods that remain reliable under sparse and anisotropic observations.

*Corresponding author.

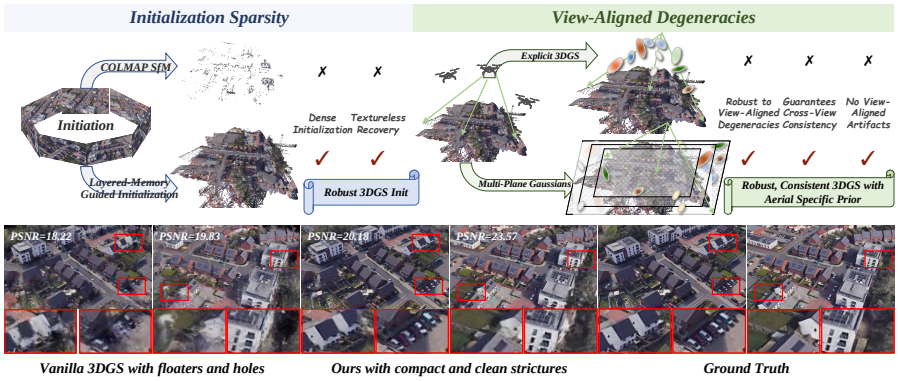

Fig. 1: Sparse aerial imagery poses two key challenges: incomplete sparse reconstruction / triangulation in textureless regions (**left**), and view-aligned minima from anisotropic observations (**right**). We introduce StratoSplat, which densifies geometry via LDI-guided completion and regularizes optimization with multi-plane priors, enabling robust reconstruction from sparse inputs.

Most existing sparse-view 3DGS studies focus on controlled indoor [10, 33, 39] or synthetic settings [24,34]. Their approaches can be grouped into three categories. Depth-based methods [9, 29, 49, 64] use pre-trained monocular depth networks to provide geometric supervision, but these models often fail to generalize to aerial viewpoints due to domain gaps. Regularization-based methods [6, 20, 51, 59] introduce self-supervised constraints to stabilize optimization, yet they offer little guidance when the inputs are extremely sparse. Data-driven approaches exploit learned priors from large-scale training, either through feed-forward prediction [7, 8, 40, 53, 58] or generative guidance [26, 27, 62], but both struggle in aerial scenes, where large-scale annotated training data is scarce and domain shift from ground-level datasets is severe. These limitations motivate a key question: *Can we achieve robust reconstruction by leveraging the geometric regularities of aerial scenes?*

To answer this question, we begin by analyzing the failure modes of 3DGS in aerial scenes. As illustrated in Fig. 1, sparse aerial inputs trigger two critical failure modes. First, extensive textureless regions cause the triangulation stage of SfM to produce incomplete point clouds with large holes that destabilize Gaussian initialization (Fig. 1, left). Note that this failure is specific to geometric triangulation, not pose estimation: UAV poses are typically available from onboard GNSS/IMU and can be assumed accurate. The challenge lies in the sparsity and incompleteness of triangulated geometry. Second, sparse anisotropic observations dominated by oblique views induce view-aligned degeneracies. Here Gaussians overfit the 2D appearance of training views by aligning with their view directions. They fit limited viewpoints rather than respect true scene geometry, producing floaters under novel views (Fig. 1, right).

These failures occur at different stages but share a common root cause: **standard 3DGS ignores the strong layered regularities inherent in aerial scenes.** While aerial inputs are sparse, the underlying scene geometry

is not arbitrary. Buildings, terrain, and roads stratify along gravity, forming distinct parallel depth layers that remain consistent across viewpoints. This inherent layered structure distinguishes aerial scenes from generic indoor or object-centric scenarios. Leveraging such regularities offers a promising path to stabilize optimization under limited views.

Building upon this insight, we propose StratoSplat, a framework that systematically exploits layered regularities in sparse aerial 3DGS. To tackle incomplete initialization from textureless regions, we perform test-time adaptation of a depth diffusion model guided by a layered memory implemented with Layered Depth Images (LDI). This memory-based guidance accumulates geometry across views with explicit depth ordering, enforcing multi-view constraints that drive diffusion toward dense and consistent depth predictions. To prevent view-aligned degeneracies during optimization, we introduce a hybrid representation combining explicit Gaussians for fine details with neural multi-plane Gaussians for geometric regularization. Virtual Gaussians anchor to learned parallel planes aligned with the stratification direction, imposing explicit geometric constraints that prevent collapse into view-aligned configurations. Crucially, the aerial-specific geometric prior stems purely from the structural regularity of aerial scenes and needs no large-scale aerial data. We adopt a generic pretrained depth model only as a backbone on which this prior operates. This makes StratoSplat particularly practical for real-world UAV deployments. The main contributions are summarized as follows:

1. We identify and analyze two critical failure modes of 3DGS in aerial scenes: incomplete triangulated sparse-point initialization from textureless regions and view-aligned degeneracies arising from sparse anisotropic observations.
2. We introduce a layered memory guided initialization scheme that performs sequential test-time depth adaptation with multi-view constraints, producing dense point clouds from incomplete sparse triangulation geometry.
3. We propose a hybrid neural multi-plane Gaussian representation that combines explicit Gaussians for fine detail with virtual Gaussians anchored to learned parallel planes, exploiting the inherent stratified structure of aerial scenes for geometric regularization.
4. We achieve state-of-the-art performance on LEVIR-NVS and a newly constructed 7-view benchmark based on 3DAS, with consistent improvements of over 3dB across varying scene types and view counts.

2 Related Work

2.1 Representations for View Synthesis

Early image-based rendering adopted Layered Depth Images (LDIs) [37], which store multiple depth and color samples per pixel with explicit depth ordering to handle occlusions across viewpoint changes. LDIs enabled early free-viewpoint navigation and real-time rendering systems [11, 32, 38, 65]. Building on layered

representations, Multi-Plane Images (MPIs) [63] organize scenes as stacked fronto-parallel RGBA planes for efficient view interpolation under narrow baselines, powering practical VR/AR applications [14, 22, 33, 42]. Neural Radiance Fields (NeRF) [34] shifted to continuous volumetric representations learned via differentiable volume rendering, achieving photorealistic quality. Follow-up work improved scene coverage and rendering speed through better sampling strategies [2, 3] and discrete acceleration structures including hash grids [35], tensor decompositions [5], voxels [16], octrees [54], and factorized planes [15]. More recently, 3D Gaussian Splatting (3DGS) [25] represents scenes with explicit anisotropic Gaussians rasterized via differentiable splatting for real-time rendering. Extensions handle aliasing [57], dynamic content [30, 46], and surface extraction [17], establishing 3DGS as a versatile framework for interactive graphics and robotics applications.

2.2 Sparse-view 3D Gaussian Splatting

Recent advances in sparse-view 3DGS explore diverse strategies to mitigate limited observations. Depth-based methods [9, 29, 49, 64] incorporate priors from pre-trained monocular estimators to regularize geometry during optimization. However, these approaches suffer from severe domain gaps in aerial imagery, as depth models trained on ground-level datasets fail to generalize to bird's-eye perspectives. Regularization-based methods [6, 20, 51, 59] impose constraints such as opacity decay, dropout, or co-adaptation suppression to prevent overfitting and encourage geometric consistency. While effective in moderately sparse settings, these generic priors offer limited guidance when observations are extremely sparse relative to scene complexity. Data-driven approaches [7, 8, 40, 53, 58] train feed-forward predictors on large-scale datasets to directly infer 3DGS from sparse inputs, but the scarcity of diverse aerial training data limits their practicality. Most existing methods focus on indoor or object-centric settings, leaving opportunities for aerial-specific solutions.

A few recent works address sparse-view synthesis in aerial contexts. MPNeRF [18] leverages multiplane priors to capture the layered structure of aerial scenes, while UPNeRF [19] introduces uncertainty-guided perceptual learning to mitigate overfitting under sparse supervision. Although these NeRF-based methods exploit aerial scene characteristics, they inherit NeRF's limitations in rendering speed and optimization cost, motivating aerial-specific 3DGS solutions.

3 Method

Given a sparse set of aerial images with camera poses assumed known from onboard GNSS/IMU, the goal is to reconstruct a photorealistic and geometrically consistent 3D scene for stable novel view synthesis. As analyzed in Sec. 1, aerial 3DGS faces two key challenges. First, SfM point clouds are sparse and uneven with large holes in textureless regions. Second, anisotropic viewpoints cause view-aligned minima during optimization. To address these issues, StratoSplat exploits

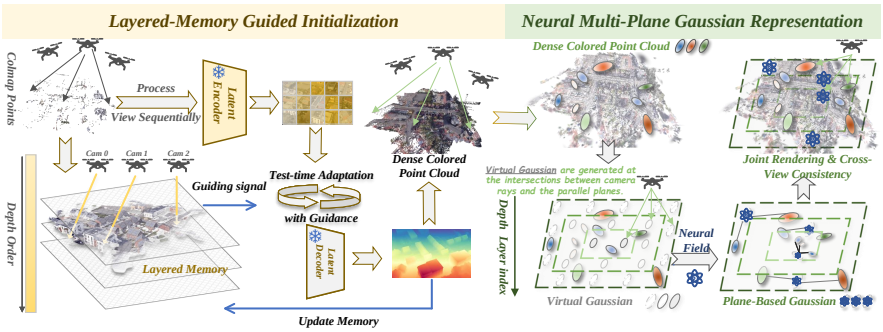

Fig. 2: Overview of StratoSplat. Left: Layered memory guided geometry initialization performs test-time adapted depth completion with layered depth images, transforming sparse COLMAP points into a dense, occlusion-aware colored point cloud. Right: neural multi-plane Gaussian representation augments explicit Gaussians with virtual Gaussians anchored on learned parallel planes via ray–plane intersections, enabling joint rendering and cross-view consistency that suppresses view-aligned degeneracies under sparse aerial views.

the inherent layered organization of aerial scenes through two components. Our Layered memory guided initialization (Sec. 3.2) sequentially adapts multi-view depths with a compact layered memory. This converts incomplete SfM geometry into a robust dense initialization $\mathcal{P}_{\text{dense}}$. Our hybrid Gaussian representation (Sec. 3.3) then couples explicit Gaussians from $\mathcal{P}_{\text{dense}}$ with virtual Gaussians anchored on learned parallel planes. This preserves geometry while regularizing against view-aligned collapse.

3.1 Preliminaries

3D Gaussian Splatting (3DGS) [25] represents a scene using N anisotropic 3D Gaussians $\mathcal{G} = \{(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \alpha_k, \boldsymbol{\phi}_k)\}_{k=1}^{N}$. Each Gaussian is parameterized by its center $\boldsymbol{\mu}_k \in \mathbb{R}^3$, covariance matrix $\boldsymbol{\Sigma}_k = \mathbf{R}_k \text{diag}(\mathbf{s}_k^2)\mathbf{R}_k^\top$ (decomposed into rotation $\mathbf{R}_k$ and scale $\mathbf{s}_k$), opacity $\alpha_k \in (0, 1)$, and spherical harmonic coefficients $\boldsymbol{\phi}_k$ for view-dependent color. For rendering from viewpoint i, each Gaussian is projected into screen space as a 2D Gaussian with center $\boldsymbol{\mu}_{ik}^{\text{img}}$ and covariance $\mathbf{S}_{ik}$. Let $\mathcal{N}_{ip}$ denote the set of Gaussians overlapping pixel p, ordered front-to-back by depth ($j \prec k$). To compute the pixel color, we apply α-compositing:

$$\hat{\mathbf{I}}_i(p) = \sum_{k \in \mathcal{N}_{ip}} \underbrace{\prod_{j \prec k} (1 - \alpha_{ijp})}_{T_{ikp}} \alpha_{ikp}\, \mathbf{C}(\boldsymbol{\phi}_k, \hat{\mathbf{v}}_i), \tag{1}$$

where $\alpha_{ijp} = \alpha_k \mathcal{N}(p; \boldsymbol{\mu}_{ik}^{\text{img}}, \mathbf{S}_{ik})$ is the pixel-wise opacity modulated by the Gaussian's 2D footprint, T_{ikp} the accumulated transmittance from front layers, and $\mathbf{C}(\boldsymbol{\phi}_k, \hat{\mathbf{v}}_i)$ the view-dependent color. The model is trained by minimizing photometric loss between rendered and observed images. Gaussians are initialized

from sparse triangulated points, with centers placed at point locations, covariances estimated from local neighborhoods, and opacities set to small values with colors sampled from input views.

3.2 Layered Memory Guided Initialization

Sparse triangulated point clouds from aerial imagery suffer from large holes in extensive textureless regions such as rooftops and roads, destabilizing Gaussian initialization. While pre-trained monocular depth estimators could potentially densify such geometry, they exhibit severe domain gaps under bird's-eye viewpoints. Models trained on ground-level datasets learn priors incompatible with aerial scenes, producing unreliable predictions.

To obtain stable initialization, we sequentially adapt a depth diffusion model at test time inspired by recent advances in controlled diffusion models [1, 43, 55]. Each view is conditioned on geometry accumulated from previously processed views. This requires a memory structure with three key properties: (1) efficient conversion from depth maps, (2) compact geometric storage, and (3) support for continuous updates as new views are processed. As each view is processed, the layered memory is incrementally densified and provides multi-view geometric constraints that guide the diffusion model toward consistent predictions. After all views are processed, we back-project the refined depth maps into 3D to produce a dense initialization $\mathcal{P}_{\text{dense}}$ for 3D Gaussians.

Layered Memory. We adopt Layered Depth Images (LDI) [37] to store geometry in our layered memory. LDI encodes multiple depth samples per pixel with explicit front-to-back ordering, naturally fulfilling the above requirements. This design aligns well with aerial scenes where buildings, terrain, and roads stratify into distinct depth layers perpendicular to the ground plane. Anchored to a reference camera $\mathcal{C}_{\text{ref}}$, each pixel coordinate (u, v) maps to an ordered sequence of layers:

$$\text{LDI} : (u, v) \mapsto \{\ell_k\}_{k=1}^{K_{u,v}}, \quad \ell_k = (d_k, \mathbf{p}_k, \mathbf{c}_k), \tag{2}$$

where each layer ℓ_k stores depth d_k, world position $\mathbf{p}_k \in \mathbb{R}^3$, color $\mathbf{c}_k \in \mathbb{R}^3$. Layers are strictly ordered by increasing depth ($d_1 < d_2 < \cdots < d_{K_{u,v}}$), enabling efficient front-to-back rendering and explicit modeling of occlusions during adaptation.

The memory is initialized with sparse points and updated incrementally as each view is processed. For each new view, we first render the current state to produce guidance maps. After depth adaptation, we project the refined depth back into the LDI coordinate system. New depth samples are inserted into the ordered layer sequence at their corresponding depth positions. When a new sample falls within threshold ϵ_d of an existing layer, we merge them by averaging depth, position, and color. This update strategy progressively densifies the memory while preserving its stratified structure across views.

Sequential Depth Adaptation. Iterative refinement is performed at each denoising timestep. At timestep t, we obtain an intermediate depth prediction $\hat{D}_i \in [0, 1]$ by estimating the clean latent via the diffusion posterior [12] and

decoding through the pre-trained network. This prediction is aligned to scene scale via learnable scale-shift parameters: $\hat{D}_i^{(m)} = \hat{D}_i \cdot \hat{a} + \hat{b}$.

We enforce two complementary multi-view constraints. The sparse loss anchors predictions at triangulated SfM points, while the visibility loss enforces cross-view consistency in regions with prior memory coverage:

$$\mathcal{L}_{\text{sparse}} = \sum_{(u,v)\in\mathcal{P}_i^{\text{SfM}}} \left| \hat{D}_i^{(m)}(u,v) - d_{uv}^{\text{SfM}} \right|_1, \tag{3}$$

$$\mathcal{L}_{\text{vis}} = \left\| \left(\hat{D}_i^{(m)} - D_i^{\text{guide}} \right) \odot M_i^{\text{vis}} \right\|_1. \tag{4}$$

The combined objective $\mathcal{L} = \mathcal{L}_{\text{sparse}} + \lambda_{\text{vis}}\mathcal{L}_{\text{vis}}$ guides backpropagation through the decoder to update both the latent representation and scale-shift parameters $\hat{a}, \hat{b}$ at each timestep. After sequentially processing all N views through the complete denoising process, we unproject the predicted depths into world space to obtain a dense colored point cloud $\mathcal{P}_{\text{dense}}$ for 3DGS initialization.

Discussion. Our approach differs from prior depth-based methods in two aspects. Unlike DNGaussian [29] and FSGS [64] that enforce depth as hard training constraints, we use depth priors only for initialization, reducing sensitivity to prediction errors (refer to Sec. 4 for results). Our sequential adaptation further promotes multi-view consistency through explicit geometric guidance from the layered memory. A natural question arises: why use LDI rather than directly accumulating a 3D point cloud across views? The key difference lies in how new geometry is integrated. Direct point cloud accumulation naively stacks points from multiple views together, causing the representation to grow uncontrollably dense. In contrast, LDI's depth-based merging mechanism intrinsically controls memory growth while preserving geometric detail. For aerial scenes captured from downward-looking viewpoints, buildings, terrain, and roads naturally stratify along depth due to gravity-aligned structure. This depth-stratified organization makes LDI particularly well-suited, as its merging naturally aligns with scene geometry, whereas unstructured point clouds fail to exploit this regularity.

3.3 Neural Multi-Plane Gaussian Representation

While our initialization approach provides robust dense geometry, explicit 3D Gaussians alone remain vulnerable to view-aligned degeneracies under extreme sparsity. Insufficient multi-view supervision causes Gaussians to overfit limited viewpoints rather than respect true geometry. To address this, we introduce a hybrid representation combining explicit Gaussians $\mathcal{G}^{\text{exp}}$ and virtual Gaussians $\mathcal{G}^{\text{vir}}$ that exploits the inherent layered structure of aerial scenes. Explicit Gaussians, initialized from $\mathcal{P}_{\text{dense}}$, adapt freely through standard densification and pruning to capture local surface variations. Virtual Gaussians are neural multi-plane primitives anchored to learned parallel planes aligned with the dominant stratification direction. This planar anchoring imposes explicit geometric constraints that prevent view-aligned collapse, regularizing scene geometry to respect the layered structure characteristic of aerial scenes.

Plane-Based Gaussian Generation. We extract the dominant stratification direction from $\mathcal{P}_{\text{dense}}$ via SVD. Specifically, we select the direction with minimal singular value as plane normal $\mathbf{n} = V[:, \arg\min_j \Sigma_{jj}]$ where $U\Sigma V^\top = \text{SVD}(\mathcal{P}_{\text{dense}} - \bar{\mathbf{p}})$. This minimal variance direction corresponds to the stratification axis perpendicular to the ground plane. We uniformly distribute K parallel planes, each parameterized by $(\mathbf{n}, d_k)$ and defined as $\mathbf{n}^\top \mathbf{p} + d_k = 0$, covering the depth range $[d_{\min}, d_{\max}]$ of $\mathcal{P}_{\text{dense}}$. Although virtual Gaussians are generated per view through ray–plane intersection, the underlying planes are globally shared and fixed across views, so the plane set does not grow with the number of input views.

To enable efficient generation of virtual Gaussians, we parameterize their attributes via a compact neural field. We define $\Phi : \mathbb{R}^3 \to (\mathbb{R}^3, \mathbb{R}, \mathbb{R}, \mathbb{R}^4)$ that maps 3D positions to RGB color, opacity logit, scale, and rotation quaternion. Unlike discrete per-Gaussian parameters, this continuous neural parameterization provides inherent spatial smoothness, encouraging virtual Gaussians to capture low-frequency geometric regularities.

At each training iteration, we generate virtual Gaussians by intersecting camera rays with the plane stack. For each pixel, the ray from camera center $\mathbf{o}$ with direction $\mathbf{d}_p$ intersects plane π_k at position $\mathbf{p}_k = \mathbf{o} + t_k \mathbf{d}_p$, where $t_k = -\frac{\mathbf{n}^\top \mathbf{o} + d_k}{\mathbf{n}^\top \mathbf{d}_p}$. The neural field is queried to obtain attributes $(\mathbf{c}_k, \tilde{\alpha}_k, s_k, \mathbf{q}_k) = \Phi(\mathbf{p}_k)$ for RGB color, opacity logit, scale, and rotation. To prevent alpha compositing's early termination from large opacity near the camera and collapsing the multi-layer structure, we apply a opacity bias $\alpha_k = \sigma(\tilde{\alpha}_k + b(t_k))$ where bias function $b(t_k)$ encourages farther planes to maintain higher opacity.

Joint Training with Geometric Regularization. During training, explicit Gaussians $\mathcal{G}^{\text{exp}}$ initialized from $\mathcal{P}_{\text{dense}}$ and virtual Gaussians $\mathcal{G}^{\text{vir}}$ generated from plane intersections are rasterized jointly via alpha compositing. The photometric loss applies to both representations, but they follow distinct optimization strategies. Explicit Gaussians undergo standard adaptive densification and pruning to capture fine geometric details with view-dependent appearance. Virtual Gaussians remain anchored to their plane intersections as only the neural field Φ adapts to observations. Constrained by fixed planar positions and view-independent colors, virtual Gaussians naturally capture smooth low-frequency structure rather than overfitting to sparse observations. This asymmetric treatment establishes a clear distinction: explicit Gaussians provide flexibility for modeling high-frequency local variations, while virtual Gaussians enforce global geometric constraints through planar anchoring.

Discussion. Our neural multi-plane Gaussians provide geometric regularization through fixed planar constraints. Virtual Gaussians differ from explicit Gaussians in two key aspects: view-independent colors prevent overfitting sparse observations, and neural field prediction enforces spatial smoothness. Crucially, their 3D positions are determined by ray-plane intersections with fixed parallel planes, preventing them from freely moving to fit arbitrary view-dependent configurations. This forces the representation to explain observations through consistent cross-view geometry rather than overfitting individual viewpoints. The

effectiveness relies on aerial scene structure. Buildings, terrain, and roads stratify into horizontal layers, naturally aligning with parallel planes. Downward-looking camera poses ensure these planes consistently intersect viewing rays across views. When scene geometry matches this planar assumption, fixed plane constraints effectively prevent view-aligned collapse by enforcing geometric consistency.

4 Experiment

4.1 Experimental Setups

Datasets and Setups. Following prior work [18], we evaluate on LEVIR-NVS [48], which provides diverse aerial scenes (urban buildings, mountains, stadiums, parks) captured by UAVs, using 3 training views per scene. To broaden the evaluation scope, we additionally construct a 7-view benchmark based on the 3DAS [41] dataset, covering 9 large-scale scenes across city, countryside, and port environments. While LEVIR-NVS offers scene diversity, its relatively small-scale captures and narrow baselines are complemented by 3DAS's wider viewing conditions and geometric complexity. We do not rely on COLMAP for pose estimation; instead, COLMAP is used only to obtain sparse triangulated points for initialization under these provided poses. This choice matches common UAV mapping workflows where camera poses are available from onboard GNSS and IMU. We report PSNR, SSIM, and LPIPS [60].

Baseline Methods. We compare against methods representing the spectrum of sparse-view radiance fields. For NeRF-based approaches, we evaluate dense-view NeRF [34], sparse-view variants DietNeRF [23], RegNeRF [36], and FreeNeRF [52], as well as the aerial-specific MPNeRF [18]. For 3DGS-based approaches, we evaluate vanilla 3DGS [25] and sparse-view extensions FSGS [64], DNGaussian [29], DropoutGS [51], and Guidedvd-3DGS [62]. We additionally include feed-forward methods NoPoSplat [53], MVSplat [7], DepthSplat [50], and TranSplat [58]. Among them, NoPoSplat operates in a pose-free setting and MVSplat is designed for two-view inputs. These are inherent design choices of the methods rather than controllable variables, and may confer trade-offs our method does not share. To compare as fairly as possible and reduce the influence of domain gap and pretraining, we evaluate these methods under a fine-tuned (ft) setting that jointly fine-tunes the predictor on the training views of all scenes. We further analyze feed-forward predictors under additional settings in Sec. 4.3. For LEVIR-NVS, we adopt results from the benchmark established in MPNeRF [18]. On 3DAS, we reproduce all baselines using official implementations to ensure fair evaluation.

Implementation Details. All experiments are conducted on a single NVIDIA 4090 GPU. All baselines are initialized from COLMAP sparse points reconstructed using poses provided by the dataset. The only exception is Guidedvd-3DGS, which uses DUSt3R [45] initialization. For ablation studies involving DUSt3R variants, we adopt Guidedvd-3DGS's initialization procedure for fair comparison. The layered memory initialization processes views sequentially in coverage-based order with approximately 12 seconds per view. The cross-view consistency weight

$\lambda_{\text{vis}} = 0.1$ and layer merging threshold $\epsilon_d = 0.02$ for 3DAS and 0.55 for LEVIR-NVS. Depth-dependent opacity bias $b(t_k)$ is normalized by ray distance with logits in $[-4.6, 1.4]$. Training converges in approximately 9 minutes. All other hyperparameters follow FSGS [64]. More details can be found in the supplementary.

4.2 Comparative Results

Table 1: Quantitative comparison with different baseline methods in 3 views in LEVIR-NVS dataset. The best, second-best, and third-best entries are marked in , , and , respectively.

Metrics	Methods	Building-1	Church	College	Mountain-1	Mountain-2	Observation	Building-2	Town-1	Stadium	Town-2	Mountain-3	Town-3	Factory	Park	School	Downtown	Mean
PSNR	NeRF [34]	13.64	12.06	14.45	18.60	18.40	14.56	14.10	14.98	15.02	13.18	20.88	14.11	14.28	15.44	14.68	13.67	15.13
	DietNeRF [23]	13.44	12.20	14.86	19.34	18.67	15.27	13.73	15.78	16.68	14.21	20.66	14.73	16.73	16.55	14.97	13.61	15.71
	RegNeRF [36]	12.07	10.79	12.60	16.39	17.36	13.04	11.92	12.94	13.49	11.59	19.37	12.21	12.66	13.98	12.71	11.21	13.40
	FreeNeRF [52]	13.56	11.01	13.93	20.03	19.74	15.29	13.00	16.29	15.47	13.21	21.59	13.15	18.91	18.23	13.35	11.87	15.54
	MPNeRF [18]	18.81	17.93	20.71	25.50	24.92	19.56	18.64	21.59	22.08	21.20	28.57	21.41	22.61	23.57	20.71	19.73	21.72
	3DGS [25]	16.22	16.13	18.73	23.84	22.54	17.56	16.69	20.23	21.37	20.89	27.50	20.78	21.95	22.30	19.18	17.35	20.20
	FSGS [64]	17.94	17.59	20.00	24.29	24.13	18.12	18.11	21.06	22.12	19.84	25.67	20.22	22.99	23.82	19.64	19.30	20.93
	DNGaussian [29]	15.89	13.78	16.74	19.41	19.02	15.62	16.82	18.21	17.68	16.95	20.55	17.37	20.46	17.70	17.25	16.24	17.48
	DropoutGS [51]	17.54	15.37	18.44	22.11	19.96	15.62	18.17	20.55	19.86	19.88	22.55	18.18	21.69	22.83	18.45	17.84	19.31
	Guidedvd-3DGS [62]	15.47	14.31	16.66	23.97	21.61	12.87	14.26	17.48	17.29	14.99	23.88	18.64	15.22	16.95	15.47	13.77	17.05
	NoPoSplat ft [53]	15.72	16.84	20.10	22.27	21.99	17.86	16.94	19.91	21.21	18.72	25.99	19.19	22.09	20.70	18.58	17.96	19.75
	MVSplat ft [7]	14.36	13.95	16.22	20.07	20.00	15.54	14.65	17.91	18.51	16.54	21.80	18.37	18.97	18.48	16.74	14.63	17.30
	DepthSplat ft [50]	12.38	12.28	13.53	19.45	18.51	14.79	12.78	15.96	16.30	15.43	22.35	15.29	17.24	14.80	14.01	14.95	15.75
	TranSplat ft [58]	14.27	13.80	15.94	20.27	19.29	16.14	15.26	17.76	16.42	16.55	21.74	15.39	17.08	16.84	16.95	16.83	16.79
	Ours	20.37	19.88	22.21	29.35	27.58	21.14	21.77	24.22	23.25	24.72	33.15	24.43	27.33	27.64	23.22	21.40	24.48
SSIM	NeRF [34]	0.16	0.12	0.17	0.30	0.24	0.16	0.15	0.14	0.20	0.17	0.34	0.22	0.22	0.20	0.23	0.22	0.20
	DietNeRF [23]	0.16	0.15	0.23	0.34	0.24	0.21	0.17	0.18	0.28	0.21	0.35	0.26	0.36	0.26	0.28	0.19	0.24
	RegNeRF [36]	0.12	0.11	0.16	0.28	0.27	0.14	0.13	0.13	0.20	0.13	0.34	0.16	0.20	0.16	0.21	0.11	0.18
	FreeNeRF [52]	0.24	0.11	0.21	0.37	0.33	0.26	0.17	0.28	0.27	0.23	0.38	0.21	0.51	0.42	0.24	0.14	0.27
	MPNeRF [18]	0.73	0.72	0.79	0.82	0.81	0.73	0.71	0.81	0.80	0.84	0.89	0.84	0.86	0.85	0.79	0.76	0.80
	3DGS [25]	0.69	0.57	0.58	0.79	0.82	0.59	0.64	0.75	0.78	0.76	0.84	0.71	0.86	0.87	0.64	0.64	0.72
	FSGS [64]	0.69	0.62	0.63	0.78	0.78	0.59	0.65	0.73	0.78	0.72	0.77	0.69	0.83	0.84	0.65	0.70	0.71
	DNGaussian [29]	0.54	0.30	0.44	0.48	0.44	0.38	0.53	0.53	0.49	0.47	0.45	0.53	0.70	0.52	0.53	0.51	0.49
	DropoutGS [51]	0.57	0.43	0.55	0.55	0.41	0.20	0.58	0.60	0.58	0.65	0.51	0.57	0.73	0.74	0.61	0.61	0.56
	Guidedvd-3DGS [62]	0.44	0.38	0.49	0.69	0.56	0.20	0.33	0.53	0.53	0.44	0.61	0.59	0.51	0.46	0.48	0.28	0.47
	NoPoSplat ft [53]	0.47	0.53	0.60	0.48	0.55	0.52	0.51	0.58	0.70	0.57	0.64	0.54	0.80	0.63	0.51	0.57	0.57
	MVSplat ft [7]	0.34	0.27	0.30	0.36	0.40	0.27	0.25	0.42	0.48	0.39	0.36	0.47	0.63	0.51	0.39	0.33	0.39
	DepthSplat ft [50]	0.18	0.19	0.23	0.36	0.30	0.26	0.18	0.28	0.32	0.32	0.44	0.35	0.47	0.30	0.30	0.30	0.30
	TranSplat ft [58]	0.19	0.19	0.26	0.34	0.30	0.24	0.23	0.33	0.24	0.32	0.35	0.23	0.39	0.37	0.32	0.32	0.29
	Ours	0.78	0.74	0.76	0.88	0.86	0.74	0.79	0.84	0.83	0.86	0.92	0.84	0.92	0.91	0.78	0.78	0.83
LPIPS	NeRF [34]	0.59	0.62	0.60	0.56	0.58	0.58	0.61	0.59	0.60	0.59	0.53	0.59	0.53	0.57	0.55	0.66	0.58
	DietNeRF [23]	0.59	0.61	0.59	0.56	0.59	0.56	0.61	0.58	0.52	0.56	0.56	0.58	0.48	0.54	0.57	0.59	0.57
	RegNeRF [36]	0.65	0.65	0.67	0.61	0.62	0.63	0.63	0.62	0.63	0.64	0.59	0.65	0.58	0.62	0.64	0.66	0.63
	FreeNeRF [52]	0.61	0.67	0.64	0.55	0.58	0.58	0.63	0.58	0.61	0.61	0.57	0.64	0.48	0.54	0.61	0.65	0.60
	MPNeRF [18]	0.21	0.24	0.18	0.20	0.18	0.25	0.24	0.18	0.20	0.16	0.12	0.17	0.12	0.14	0.20	0.19	0.19
	3DGS [25]	0.24	0.33	0.33	0.20	0.17	0.30	0.28	0.21	0.19	0.21	0.17	0.25	0.12	0.13	0.30	0.30	0.23
	FSGS [64]	0.27	0.30	0.33	0.27	0.29	0.33	0.29	0.24	0.22	0.26	0.32	0.28	0.20	0.19	0.32	0.27	0.27
	DNGaussian [29]	0.39	0.51	0.47	0.52	0.54	0.54	0.39	0.37	0.42	0.44	0.56	0.41	0.30	0.41	0.42	0.41	0.44
	DropoutGS [51]	0.42	0.45	0.43	0.56	0.46	0.66	0.38	0.36	0.42	0.36	0.62	0.41	0.31	0.35	0.41	0.37	0.43
	Guidedvd-3DGS [62]	0.49	0.53	0.47	0.28	0.42	0.67	0.61	0.39	0.49	0.54	0.36	0.39	0.49	0.51	0.54	0.62	0.49
	NoPoSplat ft [53]	0.30	0.28	0.24	0.28	0.24	0.28	0.27	0.22	0.16	0.23	0.20	0.24	0.13	0.19	0.25	0.26	0.24
	MVSplat ft [7]	0.46	0.48	0.48	0.48	0.45	0.49	0.49	0.37	0.34	0.43	0.50	0.38	0.30	0.35	0.44	0.47	0.43
	DepthSplat ft [50]	0.64	0.60	0.64	0.51	0.52	0.53	0.61	0.52	0.52	0.45	0.45	0.52	0.39	0.55	0.54	0.54	0.53
	TranSplat ft [58]	0.61	0.59	0.57	0.56	0.57	0.56	0.58	0.51	0.58	0.54	0.54	0.59	0.48	0.57	0.57	0.54	0.56
	Ours	0.17	0.20	0.19	0.12	0.14	0.21	0.16	0.14	0.15	0.13	0.09	0.14	0.08	0.09	0.20	0.19	0.15

Tab. 1 demonstrates StratoSplat's consistent superiority across all metrics on the LEVIR-NVS benchmark. StratoSplat achieves 24.48dB PSNR from only 3 training views, surpassing the strongest 3DGS baseline FSGS by 3.5dB and the previous best MPNeRF by 2.76dB. Similar improvements hold on the larger-scale

Table 2: Quantitative comparison with different baseline methods in 7 views in 3DAS dataset. The best, second-best, and third-best entries are marked in , , and , respectively.

Method	City									Country									Port								
	Scene 0			Scene 1			Scene 2			Scene 0			Scene 1			Scene 2			Scene 0			Scene 1			Scene 2		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
NeRF [34]	13.26	0.15	0.65	14.85	0.20	0.64	15.11	0.23	0.72	15.46	0.33	0.71	17.52	0.47	0.64	16.42	0.42	0.70	15.63	0.47	0.50	14.84	0.22	0.65	15.49	0.32	0.61
FreeNeRF [52]	13.52	0.12	0.58	14.28	0.15	0.56	14.31	0.17	0.61	15.52	0.27	0.58	18.30	0.42	0.52	15.78	0.32	0.55	18.98	0.59	0.31	15.04	0.23	0.54	15.89	0.35	0.48
RegNeRF [36]	13.41	0.08	0.62	15.46	0.16	0.57	15.54	0.18	0.59	16.31	0.25	0.58	18.43	0.37	0.55	16.83	0.33	0.58	17.74	0.54	0.37	15.43	0.20	0.60	16.27	0.31	0.54
3DGS [25]	20.66	0.67	0.24	20.62	0.52	0.37	19.43	0.41	0.45	19.89	0.42	0.45	24.35	0.65	0.32	19.43	0.42	0.45	23.67	0.88	0.21	24.03	0.84	0.13	24.41	0.87	0.12
FSGS [64]	19.67	0.64	0.26	18.37	0.47	0.38	14.93	0.23	0.59	16.88	0.36	0.55	22.87	0.66	0.35	22.86	0.69	0.35	23.81	0.87	0.15	24.02	0.84	0.14	24.07	0.86	0.14
DNGaussian [29]	17.59	0.49	0.38	14.62	0.19	0.59	13.33	0.22	0.68	15.31	0.34	0.65	16.95	0.48	0.58	19.98	0.60	0.57	21.62	0.80	0.20	19.31	0.61	0.30	20.86	0.75	0.22
DropoutGS [51]	19.18	0.58	0.33	18.37	0.47	0.38	14.79	0.24	0.62	15.86	0.34	0.60	17.93	0.47	0.52	16.55	0.43	0.61	21.92	0.81	0.19	21.55	0.72	0.21	22.28	0.78	0.19
Guidedvd-3DGS [62]	14.60	0.25	0.59	16.49	0.28	0.53	16.63	0.29	0.57	17.59	0.37	0.51	19.94	0.55	0.47	17.61	0.45	0.54	21.88	0.81	0.16	17.28	0.43	0.50	21.89	0.73	0.22
Ours	23.21	0.79	0.15	23.68	0.74	0.20	21.02	0.56	0.29	21.37	0.53	0.35	27.48	0.76	0.20	25.71	0.80	0.18	24.28	0.90	0.09	25.25	0.89	0.09	25.19	0.90	0.09

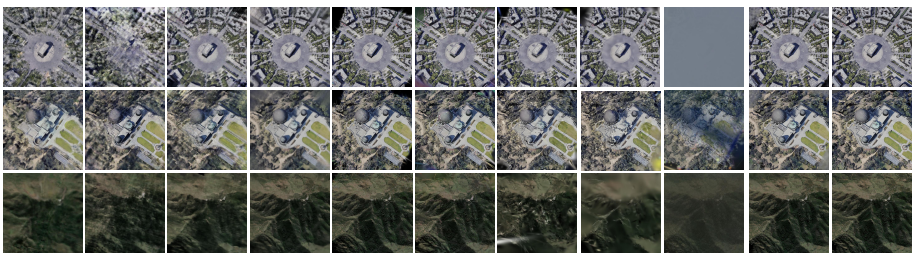

Fig. 3: Visual comparisons on three representative scenes in LEVIR dataset. StratoSplat achieves sharper textures, more complete geometry, and fewer artifacts than existing sparse-view baselines.

3DAS dataset (Tab. 2), where StratoSplat maintains state-of-the-art performance across diverse environments. Qualitative results in Figs. 3 and 4 confirm these gains, showing sharper textures and more complete geometry. This consistent advantage across different scene types, datasets, and sparsity levels validates that exploiting inherent geometric structure provides more robust guidance than generic regularization or misaligned external priors.

Methods without sparse-view adaptations such as NeRF and vanilla 3DGS, lack mechanisms to handle limited supervision. Depth-based methods provide stronger baselines by incorporating geometric priors, but their effectiveness depends critically on prior quality. DNGaussian imposes hard depth constraints from pretrained monocular estimators, making it highly sensitive to prediction errors and frequently underperforming vanilla 3DGS. FSGS improves robustness

Fig. 4: Visual comparisons on three representative scenes in 3DAS dataset. Compared to state-of-the-art baselines, StratoSplat preserves fine-scale structures and removes view-aligned artifacts, yielding sharper and more consistent renderings

through Pearson-correlation-based soft supervision, achieving the strongest 3DGS baseline. DropoutGS adds regularization via Gaussian dropout but fails to exploit aerial-specific structure. Guidedvd-3DGS combines video diffusion priors from ViewCrafter [56] with DUSt3R [45] initialization, but both components trained on non-aerial data lead to poor generalization, producing floating Gaussians that occlude content (Fig. 3). Similarly, feed-forward methods including NoPoSplat and MVSplat struggle to generalize to aerial scenes in the zero-shot setting, and fine-tuning provides only marginal improvement due to the scarcity of large-scale in-domain aerial training data. MPNeRF leverages multiplane representations to encode planar structure and achieves strong results among NeRF-based methods. However, its implicit volumetric field remains harder to constrain under sparse observations than our explicit Gaussians with fixed planar anchoring. Our approach directly embeds aerial scene structure through layered initialization and planar constraints, providing more effective regularization than generic priors or domain-shifted external models.

4.3 Ablation Studies and Further Analyses

We conduct systematic ablation studies to validate each component of StratoSplat. All experiments are performed on LEVIR-NVS with 3 training views, reporting average metrics across 16 scenes. Results are summarized in Tab. 3. More experiments and discussions can be found in supplementary.

Table 3: Ablation study. We progressively examine initialization, representation, and key components.

Method	PSNR↑	SSIM↑	LPIPS↓
Vanilla 3DGS (SfM, Explicit)	20.20	0.72	0.23
Initialization variants			
+ DUSt3R init	18.65	0.56	0.34
+ VGGT init	16.43	0.30	0.46
+ DA3 init	20.49	0.71	0.23
+ COLMAP MVS init	20.45	0.75	0.21
+ Marigold init	20.31	0.73	0.23
+ Layered Memory guided init	21.00	0.77	0.18
Representation variants			
Explicit 3DGS	21.00	0.77	0.18
Fixed multi-plane GS	21.81	0.79	0.16
w/o depth-dependent opacity bias	23.64	0.79	0.18
StratoSplat (Full)	**24.48**	**0.83**	**0.15**

Initialization Methods. To isolate initialization effects, all variants adopt explicit Gaussians with identical training. DUSt3R [45] yields dense stereo but degrades performance, highlighting severe domain gaps when ground-level priors are transferred to aerial views. Native multi-view predictors VGGT [44] and DA3 [31] avoid such pairwise stitching yet still fall below our init (16.43 and 20.49 dB), confirming that the bottleneck is the aerial domain gap rather than stitching. COLMAP MVS offers a slight improvement yet remains unreliable in textureless regions. Marigold-based monocular depth reaches 20.31 dB, indicating that independent per-view priors cannot resolve multi-view ambiguity under sparse coverage. In contrast, our initialization attains 21.00 dB by enforcing a layered structure and occlusion-aware multi-view consistency aligned with aerial stratification, providing more stable geometry than both generic learned priors and traditional stereo.

Representation Designs. Two representation strategies are evaluated under identical initialization. Fixed multi-plane arranges Gaussians on parallel planes but lacks spatial continuity due to discrete parameterization, with each Gaussian

Table 4: View ordering sensitivity analysis. We assess sensitivity to view processing order across different strategies.

Ordering Strategy	PSNR↑	SSIM↑	LPIPS↓
Random ordering (5 trials)	21.10±0.15	0.73±0.01	0.21±0.01
Order by distance	21.00	0.73	0.21
Order by coverage (Ours)	**21.14**	**0.74**	**0.21**

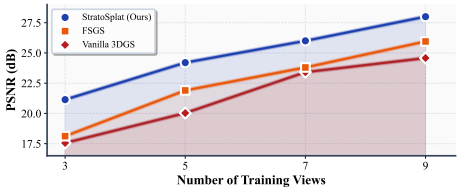

Fig. 5: Scalability to view count. StratoSplat consistently outperforms baselines across 3-9 training views.

Table 5: Ablation on memory structure design. We compare point cloud accumulation against our layered memory.

Memory Structure	Final Quality		Efficiency	
	PSNR↑	SSIM↑	Points ↓	Time(s)↓
Point Cloud Accum.	22.99	0.76	778646	98
Layered Memory (Ours)	**23.21**	**0.79**	**215607**	**84**

Table 6: Pose perturbation study (PSNR↑).

Noise Level	Completion-only	Train-only	All
Clean	24.48		
L1	24.01	24.32	24.00
L2	23.21	23.76	22.45
L3	21.88	21.19	19.11

optimized independently without inter-primitive coupling. Neural multi-plane without depth-dependent bias achieves 23.64 dB through continuous field representation, where shared network parameters enforce spatial coherence across virtual Gaussians. Full StratoSplat reaches 24.48 dB, where the depth-dependent opacity bias prevents multi-layer collapse.

View Ordering Sensitivity. To assess the robustness of sequential depth adaptation to view order, we conduct experiments on the LEVIR-NVS *Observation* scene with 3 views, comparing random orderings against two deterministic strategies: order by distance, from nearest to farthest from the scene center, and order by coverage, from highest to lowest SfM point density. Tab. 4 shows all orderings achieve similar performance. This indicates the layered memory integrates multi-view geometry reliably regardless of sequence. Coverage-based order yields slightly better metrics. This suggests that initializing the memory with geometry-rich views offers stronger guidance. The small gap reflects the broad spatial coverage typical of aerial captures.

Memory Structure. We compare layered memory against point cloud accumulation on 3DAS *City Scene 0*. As shown in Tab. 5, our layered memory achieves 0.32 dB PSNR gain with 3.6× fewer points and 14% less computation time. View-aligned layering enables efficient conflict resolution through depth ordering, avoiding uncontrolled growth from unstructured accumulation.

Scalability to View Count. To evaluate performance across varying views, we test StratoSplat with 3, 5, 7, and 9 training views on the LEVIR-NVS *Building-1* scene. Fig. 5 shows that our method consistently outperforms baselines across all view counts, maintaining advantage even as observations increase, demonstrating that our layered geometric priors remain beneficial across different sparsity.

Pose Sensitivity. We inject controlled SE(3) perturbations at three levels—L1 (rot=0.5°, trans=0.001·extent), L2 (rot=1.0°, trans=0.002·extent), L3 (rot=2.0°, trans=0.005·extent)—into the completion stage, training, or both. Performance

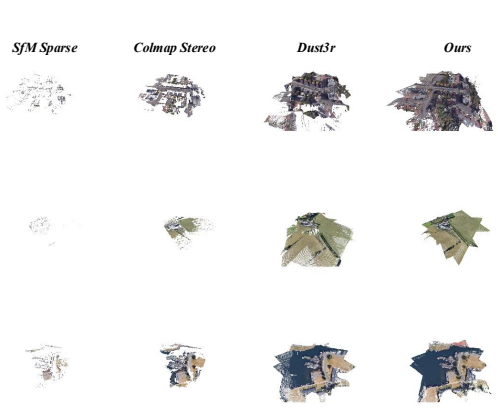

Fig. 6: Point cloud initialization comparison. From left to right: COLMAP sparse, COLMAP MVS, DUSt3R, and ours.

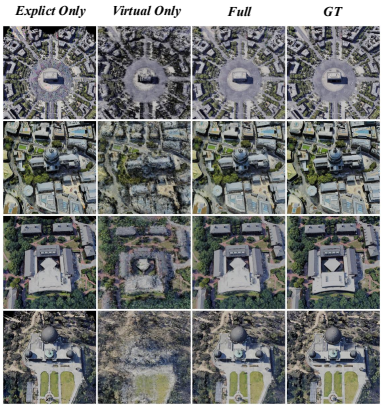

Fig. 7: Rendering contribution visualization. Columns (left to right): Explicit Only, Virtual Only, Full (Ours), Ground Truth.

degrades with noise strength (Tab. 6). Completion-only noise causes larger drops than Train-only, since pose errors degrade cross-view consistency during initialization and subsequently destabilize optimization. Overall, the results demonstrate that our pipeline remains stable under mild-to-moderate perturbations. In practice, UAV onboard GNSS/IMU typically provides accurate pose, making this a manageable limitation. Joint pose refinement remains an avenue for future work.

Initialization Method Comparison. Fig. 6 compares point clouds from different initialization methods. SfM sparse points leave severe holes in textureless regions, while COLMAP MVS improves density but remains unreliable in texture-poor areas. DUSt3R generates dense predictions but suffers from domain gap at aerial viewpoints, producing geometrically inconsistent artifacts. In contrast, our LDI-guided initialization achieves both density and multi-view consistency through sequential depth adaptation, yielding complete and coherent point clouds for robust 3DGS initialization.

Rendering Contribution Visualization. Fig. 7 visualizes rendering contributions of explicit and virtual Gaussians. Virtual Gaussians provide geometric regularization through plane-based anchoring but lack high-frequency details due to view-independent colors and neural field smoothness. Explicit Gaussians compensate with fine surface variations. The full hybrid combines both, achieving complete coverage with accurate geometry. This validates our design: virtual Gaussians enforce stratified structure while explicit Gaussians preserve local detail.

Feed-forward overfitting analysis. We further analyze why feed-forward predictors underperform on aerial scenes. We evaluate DepthSplat and TranSplat under zero-shot, ft, and ft+opt settings (Tab. 7). Across all settings, their novel-view quality stays far below ours, and per-scene 3DGS optimization does not close the gap. Notably, TranSplat reaches 49.76 dB on training views but only 17.31 dB on novel views. This large train–novel gap is exactly the view-aligned

Table 7: Feed-forward baselines under different settings on LEVIR-NVS (3 views, 16-scene mean). *ft* jointly fine-tunes the predictor on the training views of all scenes; *ft+opt* additionally runs per-scene 3DGS optimization on top.

Method	Setting	Novel View PSNR↑	Novel View SSIM↑	Novel View LPIPS↓	Train View PSNR
DepthSplat [50]	zero-shot	7.24	0.01	0.74	–
	ft	15.75	0.30	0.53	17.58
	ft+opt	15.60	0.33	0.51	32.03
TranSplat [58]	zero-shot	7.82	0.02	0.79	–
	ft	16.79	0.29	0.56	18.81
	ft+opt	17.31	0.41	0.47	49.76
Ours	–	24.48	0.83	0.15	–

overfitting our method targets, where Gaussians fit the few observed views rather than recover consistent geometry, reinforcing the need for aerial-specific priors.

5 Conclusion

We present StratoSplat, a framework for robust sparse-view aerial 3DGS that exploits the layered regularities of aerial scenes. Standard 3DGS methods fail under sparse aerial captures due to incomplete SfM initialization from textureless regions and view-aligned degeneracies from anisotropic observations. We address these challenges by embedding layered geometric priors tailored to aerial scene structure. Our layered memory guided initialization accumulates multi-view geometry in ordered layers, providing occlusion-aware constraints for dense and consistent initialization. Our neural multi-plane Gaussians anchor virtual Gaussians to learned parallel planes, enforcing geometric regularization that prevents view-aligned collapse. Experiments demonstrate state-of-the-art performance, surpassing the best baseline by 3 dB. Our work validates that exploiting scene-specific geometric structure provides robust guidance for sparse aerial 3DGS.

Acknowledgements

This work was supported in part by the Joint Funds of the National Natural Science Foundation of China (U22B2054), the National Natural Science Foundation of China (62076192, 62276199, 62431020 and 62276201), the 111 Project, the Program for Cheung Kong Scholars and Innovative Research Team in University (IRT 15R53), the Science and Technology Innovation Project from the Chinese Ministry of Education. the National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, Xi'an Jiaotong University (HMHAI-202404and HMHAI-202405).

References

1. Bansal, A., Chu, H.M., Schwarzschild, A., Sengupta, S., Goldblum, M., Geiping, J., Goldstein, T.: Universal guidance for diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 843–852 (2023)
2. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5855–5864 (2021)
3. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5470–5479 (2022)
4. Chabot, F., Granger, N., Lapouge, G.: Gaussianbev: 3d gaussian representation meets perception models for bev segmentation. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 2250–2259. IEEE (2025)
5. Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H.: Tensorf: Tensorial radiance fields. In: European conference on computer vision. pp. 333–350. Springer (2022)
6. Chen, K., Zhong, Y., Li, Z., Lin, J., Chen, Y., Qin, M., Wang, H.: Quantifying and alleviating co-adaptation in sparse-view 3d gaussian splatting. arXiv preprint arXiv:2508.12720 (2025)
7. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In: European Conference on Computer Vision. pp. 370–386. Springer (2024)
8. Chen, Y., Zheng, C., Xu, H., Zhuang, B., Vedaldi, A., Cham, T.J., Cai, J.: Mvsplat360: Feed-forward 360 scene synthesis from sparse views. Advances in Neural Information Processing Systems **37**, 107064–107086 (2024)
9. Chung, J., Oh, J., Lee, K.M.: Depth-regularized optimization for 3d gaussian splatting in few-shot images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 811–820 (2024)
10. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proc. Computer Vision and Pattern Recognition (CVPR), IEEE (2017)
11. Dhamo, H., Tateno, K., Laina, I., Navab, N., Tombari, F.: Peeking behind objects: Layered depth prediction from a single image. Pattern Recognition Letters **125**, 333–340 (2019)
12. Efron, B.: Tweedie's formula and selection bias. Journal of the American Statistical Association **106**(496), 1602–1614 (2011)
13. Fei, B., Xu, J., Zhang, R., Zhou, Q., Yang, W., He, Y.: 3d gaussian as a new vision era: A survey. CoRR (2024)
14. Flynn, J., Broxton, M., Debevec, P., DuVall, M., Fyffe, G., Overbeck, R., Snavely, N., Tucker, R.: Deepview: View synthesis with learned gradient descent. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2367–2376 (2019)
15. Fridovich-Keil, S., Meanti, G., Warburg, F.R., Recht, B., Kanazawa, A.: K-planes: Explicit radiance fields in space, time, and appearance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12479–12488 (2023)
16. Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5501–5510 (2022)

17. Gao, L., Yang, J., Zhang, B.t., Sun, J.m., Yuan, Y.j., Fu, H., Lai, Y.k.: Real-time large-scale deformation of gaussian splatting. ACM Transactions on Graphics (TOG) **43**(6), 1–17 (2024)
18. Gao, Z., Jiao, L., Li, L., Liu, X., Liu, F., Chen, P., Guo, Y.: Multiplane prior guided few-shot aerial scene rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5009–5019 (2024)
19. Gao, Z., Li, L., Liu, X., Jiao, L., Liu, F., Yang, S.: Uncertainty guided progressive few-shot learning perception for aerial view synthesis. IEEE Transactions on Multimedia (2024)
20. Han, L., Zhou, J., Liu, Y.S., Han, Z.: Binocular-guided 3d gaussian splatting with view consistency for sparse view synthesis. Advances in Neural Information Processing Systems **37**, 68595–68621 (2024)
21. Han, X., Jia, Z., Li, B., Wang, Y., Ivanovic, B., You, Y., Liu, L., Wang, Y., Pavone, M., Feng, C., et al.: Extrapolated urban view synthesis benchmark. arXiv preprint arXiv:2412.05256 (2024)
22. Han, Y., Wang, R., Yang, J.: Single-view view synthesis in the wild with learned adaptive multiplane images. In: ACM SIGGRAPH 2022 Conference Proceedings. pp. 1–8 (2022)
23. Jain, A., Tancik, M., Abbeel, P.: Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5885–5894 (2021)
24. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 406–413 (2014)
25. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
26. Kong, H., Yang, X., Wang, X.: Generative sparse-view gaussian splatting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26745–26755 (2025)
27. Kong, H., Yang, X., Wang, X.: Rogsplat: Robust gaussian splatting via generative priors. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 25735–25745 (2025)
28. Li, H., Gao, Y., Peng, H., Wu, C., Ye, W., Zhan, Y., Zhao, C., Zhang, D., Wang, J., Han, J.: Dgtr: Distributed gaussian turbo-reconstruction for sparse-view vast scenes. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 207–213. IEEE (2025)
29. Li, J., Zhang, J., Bai, X., Zheng, J., Ning, X., Zhou, J., Gu, L.: Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20775–20785 (2024)
30. Li, Z., Chen, Z., Li, Z., Xu, Y.: Spacetime gaussian feature splatting for real-time dynamic view synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8508–8520 (2024)
31. Lin, H., Chen, S., Liew, J., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
32. Liu, C., Kohli, P., Furukawa, Y.: Layered scene decomposition via the occlusion-crf. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 165–173 (2016)

33. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (ToG) **38**(4), 1–14 (2019)
34. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
35. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM transactions on graphics (TOG) **41**(4), 1–15 (2022)
36. Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S., Geiger, A., Radwan, N.: Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5480–5490 (2022)
37. Shade, J., Gortler, S., He, L.w., Szeliski, R.: Layered depth images. In: Proceedings of the 25th annual conference on Computer graphics and interactive techniques. pp. 231–242 (1998)
38. Shih, M.L., Su, S.Y., Kopf, J., Huang, J.B.: 3d photography using context-aware layered depth inpainting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8028–8038 (2020)
39. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., Clarkson, A., Yan, M., Budge, B., Yan, Y., Pan, X., Yon, J., Zou, Y., Leon, K., Carter, N., Briales, J., Gillingham, T., Mueggler, E., Pesqueira, L., Savva, M., Batra, D., Strasdat, H.M., Nardi, R.D., Goesele, M., Lovegrove, S., Newcombe, R.: The Replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
40. Szymanowicz, S., Insafutdinov, E., Zheng, C., Campbell, D., Henriques, J.F., Rupprecht, C., Vedaldi, A.: Flash3d: Feed-forward generalisable 3d scene reconstruction from a single image. In: 2025 International Conference on 3D Vision (3DV). pp. 670–681. IEEE (2025)
41. Tang, X., Jia, J., Wang, Y., Ma, J., Zhang, X.: Semantic-aware dropsplat: Adaptive pruning of redundant gaussians for 3d aerial-view segmentation. arXiv preprint arXiv:2508.09626 (2025)
42. Tucker, R., Snavely, N.: Single-view view synthesis with multiplane images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 551–560 (2020)
43. Viola, M., Qu, K., Metzger, N., Ke, B., Becker, A., Schindler, K., Obukhov, A.: Marigold-dc: Zero-shot monocular depth completion with guided diffusion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5359–5370 (2025)
44. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5294–5306 (2025)
45. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20697–20709 (2024)
46. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Wang, X.: 4d gaussian splatting for real-time dynamic scene rendering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20310–20320 (2024)

47. Wu, T., Yuan, Y.J., Zhang, L.X., Yang, J., Cao, Y.P., Yan, L.Q., Gao, L.: Recent advances in 3d gaussian splatting. Computational Visual Media **10**(4), 613–642 (2024)
48. Wu, Y., Zou, Z., Shi, Z.: Remote sensing novel view synthesis with implicit multiplane representations. IEEE Transactions on Geoscience and Remote Sensing **60**, 1–13 (2022)
49. Xiong, H., Muttukuru, S., Upadhyay, R., Chari, P., Kadambi, A.: Sparsegs: Real-time 360 {\deg} sparse view synthesis using gaussian splatting. arXiv preprint arXiv:2312.00206 (2023)
50. Xu, H., Peng, S., Wang, F., Blum, H., Barath, D., Geiger, A., Pollefeys, M.: Depthsplat: Connecting gaussian splatting and depth. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16453–16463 (2025)
51. Xu, Y., Wang, L., Chen, M., Ao, S., Li, L., Guo, Y.: Dropoutgs: Dropping out gaussians for better sparse-view rendering. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 701–710 (2025)
52. Yang, J., Pavone, M., Wang, Y.: Freenerf: Improving few-shot neural rendering with free frequency regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8254–8263 (2023)
53. Ye, B., Liu, S., Xu, H., Li, X., Pollefeys, M., Yang, M.H., Peng, S.: No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. arXiv preprint arXiv:2410.24207 (2024)
54. Yu, A., Li, R., Tancik, M., Li, H., Ng, R., Kanazawa, A.: Plenoctrees for real-time rendering of neural radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5752–5761 (2021)
55. Yu, H.X., Duan, H., Herrmann, C., Freeman, W.T., Wu, J.: Wonderworld: Interactive 3d scene generation from a single image. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5916–5926 (2025)
56. Yu, W., Xing, J., Yuan, L., Hu, W., Li, X., Huang, Z., Gao, X., Wong, T.T., Shan, Y., Tian, Y.: Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. arXiv preprint arXiv:2409.02048 (2024)
57. Yu, Z., Chen, A., Huang, B., Sattler, T., Geiger, A.: Mip-splatting: Alias-free 3d gaussian splatting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19447–19456 (2024)
58. Zhang, C., Zou, Y., Li, Z., Yi, M., Wang, H.: Transplat: Generalizable 3d gaussian splatting from sparse multi-view images with transformers. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 9869–9877 (2025)
59. Zhang, J., Li, J., Yu, X., Huang, L., Gu, L., Zheng, J., Bai, X.: Cor-gs: sparse-view 3d gaussian splatting via co-regularization. In: European Conference on Computer Vision. pp. 335–352. Springer (2024)
60. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)
61. Zhang, S., Ye, B., Chen, X., Chen, Y., Zhang, Z., Peng, C., Shi, Y., Zhao, H.: Drone-assisted road gaussian splatting with cross-view uncertainty. arXiv preprint arXiv:2408.15242 (2024)
62. Zhong, Y., Li, Z., Chen, D.Z., Hong, L., Xu, D.: Taming video diffusion prior with scene-grounding guidance for 3d gaussian splatting from sparse inputs. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6133–6143 (2025)
63. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: Learning view synthesis using multiplane images. arXiv preprint arXiv:1805.09817 (2018)

64. Zhu, Z., Fan, Z., Jiang, Y., Wang, Z.: Fsgs: Real-time few-shot view synthesis using gaussian splatting. In: European conference on computer vision. pp. 145–163. Springer (2024)
65. Zitnick, C.L., Kang, S.B., Uyttendaele, M., Winder, S., Szeliski, R.: High-quality video view interpolation using a layered representation. ACM transactions on graphics (TOG) **23**(3), 600–608 (2004)