SPECGUARD: A CAMERA-MANIFOLD GAUSSIAN ATLAS FOR RADIANCE FIELD RECONSTRUCTION

Anonymous authors
Paper under double-blind review

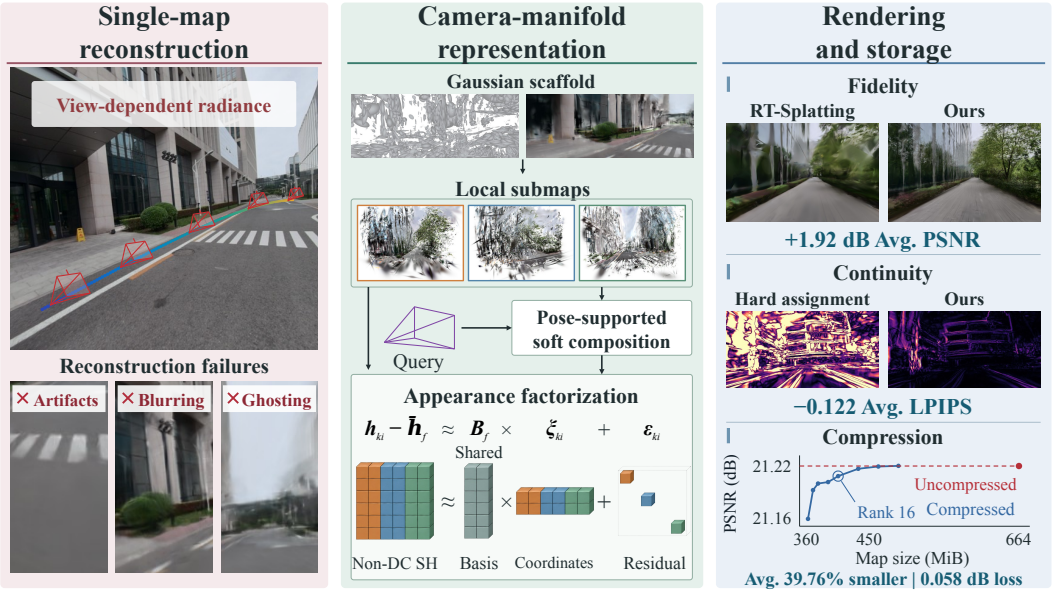


Figure 1: **SpecGuard** reconstructs scenes from posed images using a shared Gaussian scaffold to initialize local submaps that adapt to local viewing distributions. Pose-supported soft composition suppresses submap switching artifacts, while shared low-rank bases reduce storage. RGB examples and the compression curve illustrate SG-Robot Campus.

ABSTRACT

Radiance field reconstruction seeks to recover a renderable 3D scene from multi-view images. Monocular video captures changing views of scene structure and reflections, requiring a representation that preserves local detail while adapting to view-dependent appearance. We address this challenge with SpecGuard, a camera-manifold Gaussian atlas that combines scaffold-seeded local submaps with pose-supported soft composition. Given camera poses, a Gaussian scaffold seeds trajectory-local Gaussian submaps, each optimized for its local viewing distribution. To render across submaps, pose-supported soft composition suppresses submap switching artifacts, while rasterized opacity adjusts contributions according to ray-wise coverage. To reduce the storage overhead of overlapping submaps, shared low-rank bases and sparse residuals compress their non-DC spherical-harmonic coefficients. Full-session evaluations demonstrate improved aggregate reconstruction fidelity and reduced submap switching artifacts. The resulting representation reduces map storage with a small fidelity loss while supporting real-time rendering.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Photorealistic 3D maps support robot navigation and autonomous-driving simulation (Chen et al., 2024, Tonderski et al., 2024). For mobile platforms, monocular video provides a practical and accessible visual input owing to its simple setup and low hardware cost (Shu et al., 2021, Zhou et al., 2025a, Okawara et al., 2025). To reconstruct 3D scenes from such sequences, 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) combines explicit primitives with real-time rasterization and serves as a dense mapping engine in Gaussian SLAM (Matsuki et al., 2024, Zheng et al., 2025, Cheng et al., 2025). In monocular video, the camera samples different scene regions from different viewpoints, producing local viewing distributions along its path. Reflections on glass and other reflective surfaces further change with viewpoint, complicating appearance reconstruction (Shi et al., 2026, Zhou et al., 2025b). A single Gaussian map fits these observations with one set of primitives and spherical-harmonic (SH) coefficients. This motivates allocating representation capacity to local viewing distributions while retaining a shared spatial reference.

Local representations distribute reconstruction across subsets of observations. Spatial partitioning applies this principle to large scenes (Liu et al., 2024b, Lin et al., 2024, Chen & Lee, 2024, Wu et al., 2025), while trajectory-local radiance fields optimize over camera neighborhoods (Meuleman et al., 2023). Combining locally optimized representations into one renderable map raises three practical concerns. First, building independently initialized regions repeats geometric initialization. Second, switching between independently optimized regions can produce popping artifacts, while incomplete local coverage leaves visible holes. While recent blockwise methods enhance consistency via momentum guidance or occlusion-aware partitioning (Fan et al., 2025, Liu et al., 2025), smoothly arbitrating overlapping submaps across continuous camera paths remains challenging. Finally, maintaining independent SH coefficients across overlapping submaps duplicates appearance parameters, causing storage to scale linearly with the number of submaps.

Addressing these concerns requires organizing shared initialization and local view-dependent appearance across camera trajectories. A shared Gaussian scaffold provides a common initialization for local submaps, each optimized for its local viewing distribution. While recent works explore directional appearance factorization (Zhang et al., 2025) or attribute correlations (Shin et al., 2025, Wang et al., 2024a) for monolithic models, overlapping local submaps introduce redundancy across representations. Their overlapping observations motivate sharing low-rank bases across submaps while retaining local residuals.

Guided by these principles, we present SpecGuard, a Gaussian mapping backend for continuous monocular video sessions. First, SpecGuard bootstraps trajectory-local submaps from a full-sequence Gaussian scaffold using a frustum-incidence score, eliminating repeated dense geometric frontends. At query time, viewpoint proximity retrieves a bounded set of local submaps, and pose-supported soft composition blends their renders with the global fallback to suppress boundary switching artifacts, while rasterized opacity refines per-ray spatial coverage. Finally, shared low-rank bases compress non-DC SH coefficients across submap families, with sparse residuals retaining coefficient information not captured by the bases.

Our contributions are:

- We propose SpecGuard, a camera-manifold Gaussian atlas for view-dependent radiance field reconstruction that uses a shared Gaussian scaffold to initialize trajectory-local Gaussian submaps, each optimized for its local viewing distribution.
- We develop pose-supported soft composition to suppress submap switching artifacts, with rasterized opacity refining ray-wise coverage. Query-aware appearance factorization compresses non-DC SH coefficients through shared low-rank bases and sparse residuals.
- Extensive evaluations across four benchmark datasets show that our framework improves observed-view reconstruction fidelity over existing baselines under shared poses, reduces visual discontinuities across submaps compared to hard assignment, and lowers storage overhead by 39.8% at a 0.058 dB fidelity cost.

2 Related Work

Gaussian rendering and mapping. 3DGS represents a scene with anisotropic Gaussian primitives and a visibility-aware rasterizer (Kerbl et al., 2023). MonoGS uses Gaussian rendering for incremental camera tracking and mapping (Matsuki et al., 2024). SplaTAM uses RGB-D observations and a rendered silhouette to guide tracking and map expansion (Keetha et al., 2024). Splat-SLAM adds global pose optimization and map deformation for RGB-only input (Sandström et al., 2024). WildGS-SLAM addresses dynamic environments, while S3PO-GS uses scale-consistent pointmaps for outdoor monocular mapping (Zheng et al., 2025, Cheng et al., 2025).

Reflective and transparent Gaussian rendering. Mirror-3DGS and MirrorGaussian use mirror geometry and virtual-camera reasoning (Meng et al., 2024, Liu et al., 2024a). GaussianShader introduces shading for reflective surfaces, while GS-IR estimates geometry, materials, and illumination for relighting (Jiang et al., 2024, Liang et al., 2024). RTR-GS and SpecTRe-GS further model complex reflections and near-field inter-reflections (Zhou et al., 2025b, Tang et al., 2025). TransparentGS, RT-Splatting, and RT-GS model transparent or reflection-transmission effects (Huang et al., 2025, Shi et al., 2026, Zeng et al., 2026). Alongside these optical models, Ref-GS factorizes spatial and directional features to represent view-dependent appearance in 2D Gaussian splatting (Zhang et al., 2025).

Local and scalable scene representations. Local radiance fields allocate multiple representations along a camera trajectory (Meuleman et al., 2023), whereas Block-NeRF and Mega-NeRF partition large scenes into separately trained fields (Tancik et al., 2022, Turki et al., 2022). VastGaussian extends spatial partitioning to Gaussian reconstruction, and CityGaussian combines blockwise training with level-of-detail rendering (Lin et al., 2024, Liu et al., 2024b). Storage is a related concern: ContextGS uses anchor-level context for entropy coding, while LocoGS exploits local correlations among Gaussian attributes (Wang et al., 2024a, Shin et al., 2025). For changing observations, NeRF-W learns appearance and transient embeddings (Martin-Brualla et al., 2021). K-Planes and dynamic Gaussian representations model scene variation over time (Fridovich-Keil et al., 2023, Luiten et al., 2024, Wu et al., 2024).

3 Method

3.1 Representation and Problem Setup

We consider a posed monocular RGB trajectory $(\mathbf{I}, \mathbf{T})$, where $\mathbf{I}$ contains calibrated, rectified observations and $\mathbf{T}$ their fixed poses. A VGGT-Omega point cloud (Wang et al., 2026) initializes the global primitives. Following 3DGS (Kerbl et al., 2023), each $g_i \in \mathcal{G}$ has center $\boldsymbol{\mu}_i$, covariance $\boldsymbol{\Sigma}_i$, opacity α_i, and SH coefficients $\mathbf{c}_i$. Tile-based rasterization $(\mathbf{C}, O) = \mathcal{R}(\mathcal{G}, \mathbf{T})$ outputs color and opacity through front-to-back alpha compositing.

As shown in Figure 2, the global map provides shared spatial anchors and seed centers and colors, while local submaps specialize radiance within bounded camera neighborhoods. Spatial Gaussian states remain unchanged during composition and compression. The equations below give the reference formulation. The supplementary material specifies the evaluated implementation and compares its seed scoring, pose support, and grouping choices with this formulation.

3.2 Scaffold-Seeded Submap Construction

We first optimize the global Gaussian map $\mathcal{G}$ over the full trajectory using the standard 3DGS photometric reconstruction objective (Kerbl et al., 2023), together with scale and opacity regularization denoted by $\mathcal{L}_{\text{rec}}$. We then divide the trajectory into contiguous observation windows and initialize one local submap for each window. The global primitives assigned to a window are determined by the following frustum-incidence score:

$$v_i^{(k)} = \alpha_i \frac{|\{t \in W_k \mid \boldsymbol{\mu}_i \in \mathcal{F}_t\}|}{|W_k|}, \tag{1}$$

where $\mathcal{F}_t$ denotes the viewing frustum of camera pose $\mathbf{T}_t$. The score combines the fraction of window views whose frusta contain the Gaussian center with its global opacity.

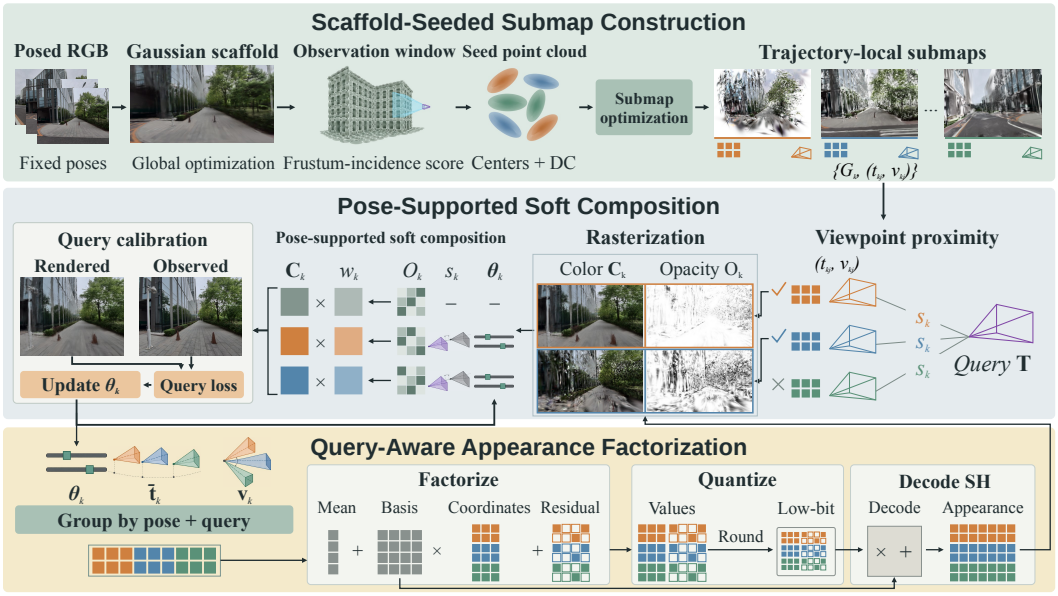


Figure 2: **Overview of SpecGuard.** A global Gaussian scaffold first seeds trajectory-local submaps via frustum-incidence scoring. For novel views, active submaps are retrieved by viewpoint proximity and composited using ray opacities with calibrated query weights. Finally, correlated submaps are clustered by pose and query states to factorize directional appearance into shared bases and sparse residuals.

For each observation window, the highest-scoring primitives provide their centers and DC colors as a local seed point cloud. The remaining Gaussian attributes are initialized independently, and the resulting submap is optimized over the observations in W_k:

$$\mathcal{G}_k^\star = \arg\min_{\mathcal{G}_k} \frac{1}{|W_k|} \sum_{t \in W_k} \mathcal{L}_{\text{rec}} \left(\mathcal{R}(\mathcal{G}_k, \mathbf{T}_t), \mathbf{I}_t \right). \tag{2}$$

This construction reuses one global initialization while allowing each submap to specialize to its local viewing distribution.

3.3 Pose-Supported Soft Composition

To access local submaps, we associate submap k with the pose set $\mathcal{P}_k$ along its observation window. For query pose $\mathbf{T}$ with position $\mathbf{t}$ and unit viewing direction $\mathbf{v}$, its viewpoint proximity is

$$s_k(\mathbf{T}) = \max_{\mathbf{T}_i \in \mathcal{P}_k} \exp\left(-\frac{\|\mathbf{t} - \mathbf{t}_i\|_2^2}{\sigma_t^2} - \frac{1 - \mathbf{v}^\top \mathbf{v}_i}{\sigma_v^2} \right), \tag{3}$$

where $(\mathbf{t}_i, \mathbf{v}_i)$ is the position and unit viewing direction of $\mathbf{T}_i$, and σ_t and σ_v control spatial and angular support. We retain a bounded set of submaps with the highest viewpoint proximity scores, denoted by $\mathcal{A}$. This bounds the rendered branch count, while total query cost also depends on retrieval and rasterization. For a fixed candidate set, the composition is continuous in the query pose, but hard bounded retrieval makes the overall mapping piecewise continuous at candidate-set transitions.

While viewpoint proximity identifies candidate submaps, rendering also requires local surface coverage. We therefore composite the final pixel color as a convex combination of the global map $\mathcal{G}$ and active submaps $\mathcal{A}$. We use index 0 for the global-map fallback branch and index k for a trajectory-local submap:

$$\widehat{\mathbf{C}} = w_0 \mathbf{C}_0 + \sum_{k \in \mathcal{A}} w_k \mathbf{C}_k, \tag{4}$$

The viewpoint proximity provides a prior for selecting and weighting local submaps. We introduce two learned scalar parameters, β_k and κ_k, to calibrate the contribution of submap k: β_k controls its

contribution, while κ_k controls how that contribution varies with viewpoint proximity. Given the rendered color and opacity $(\mathbf{C}_k, O_k)$ of each active submap, we define the normalized composition weights by

$$w_0 \propto O_0, \qquad w_k \propto O_k s_k \exp(\beta_k + \kappa_k s_k)\,. \tag{5}$$

Composition weights are normalized over the global fallback and active local submaps. Rasterized opacity reduces the contribution of a submap where it provides little rasterized coverage, allowing the global map to provide the fallback rendering.

With Gaussian parameters and poses fixed, we calibrate (β_k, κ_k) on the observed RGB trajectory, changing only composition weights.

For a temperature-scaled normalized exponential formulation, the calibration derivatives are:

$$\frac{\partial \widehat{\mathbf{C}}}{\partial \beta_k} = \frac{\partial \widehat{\mathbf{C}}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\eta}} \frac{\partial \boldsymbol{\eta}}{\partial \beta_k} = \frac{w_k}{\tau}(\mathbf{C}_k - \widehat{\mathbf{C}}), \qquad \frac{\partial \widehat{\mathbf{C}}}{\partial \kappa_k} = \frac{\partial \widehat{\mathbf{C}}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial \boldsymbol{\eta}} \frac{\partial \boldsymbol{\eta}}{\partial \kappa_k} = s_k \frac{w_k}{\tau}(\mathbf{C}_k - \widehat{\mathbf{C}}). \tag{6}$$

where $\boldsymbol{\eta}$ denotes branch logits. The branch-composite difference determines color sensitivity. The observed-RGB loss gradient determines the update direction, with s_k modulating the effect of κ_k. Unit temperature recovers the weighting rule above. The supplementary material specifies the evaluated calibration rule.

3.4 Query-Aware Appearance Factorization

We reduce local-submap storage by sharing low-rank bases for non-DC SH coefficients across submaps with similar viewpoints and calibration parameters. Geometry, primitive opacities, and DC coefficients remain unchanged.

We group submaps into families according to their pose statistics and calibration parameters. To measure the similarity between two submaps k and ℓ, we use a multi-cue distance:

$$d(k, \ell) = \lambda_t \|\bar{\mathbf{t}}_k - \bar{\mathbf{t}}_\ell\|_2 + \lambda_v (1 - \bar{\mathbf{v}}_k^\top \bar{\mathbf{v}}_\ell) + \lambda_\theta \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_\ell\|_2, \tag{7}$$

where $\boldsymbol{\theta}_k = [\beta_k, \kappa_k]^\top$ denotes the calibration parameters of submap k. The three terms measure camera center separation, viewing-axis deviation, and calibration-parameter discrepancy, respectively. We then group the local submaps into families according to this distance using agglomerative clustering. For family f, we denote its member submaps by $\mathcal{F}_f$.

For each family f, principal component analysis of its pooled non-DC SH coefficients $\mathbf{h}_{ki}$ gives a rank-r orthonormal basis $\mathbf{B}_f$. We encode each centered vector as $\boldsymbol{\xi}_{ki} = \mathbf{B}_f^\top(\mathbf{h}_{ki} - \bar{\mathbf{h}}_f)$, where $\bar{\mathbf{h}}_f$ is the family mean.

We model directional appearance with a low-rank component and a sparse residual that preserves information not captured by the shared basis:

$$\mathbf{h}_{ki} = \bar{\mathbf{h}}_f + \mathbf{B}_f \boldsymbol{\xi}_{ki} + \boldsymbol{\varepsilon}_{ki}, \tag{8}$$

where $\boldsymbol{\varepsilon}_{ki}$ retains information outside the shared basis. We uniformly quantize the coordinates and retained residual entries for offline storage and decode the approximate coefficients before composition.

4 Experiments

4.1 Experimental Setup

Benchmarks and Baselines. We evaluate DL3DV (Ling et al., 2024), GSIM (Wang et al., 2024b), Wild-SLAM (Zheng et al., 2025), and SG-Robot: *Facade* follows glass facades along a straight path, while *Campus* spans wider viewpoints. MonoGS (Matsuki et al., 2024), S3PO-GS (Cheng et al., 2025), Splat-SLAM (Sandström et al., 2024), and WildGS-SLAM use native poses. MirrorGS (an unofficial Mirror-3DGS implementation) (Meng et al., 2024), RTR-GS (Zhou et al., 2025b), RT-Splatting (Shi et al., 2026), and SpecGuard share VGGT-Omega poses, intrinsics, and frontend points. Global 3DGS (Kerbl et al., 2023) is an internal control. Native-resolution PSNR (Wang & Bovik, 2009), SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018) are averaged first within sequences, then across sequences. Runtime includes synchronized retrieval, rasterization, and composition.

Implementation Details. PyTorch/`gsplat` optimizes the scaffold for 30k iterations and each $L = 20$-frame submap for 7k, capped at 200k Gaussians. Defaults are $M = 3$ active submaps, rank 16, and 4% non-DC SH residuals. Table 2 uses uncompressed submaps. Ablations fix schedules, seeds, poses, and evaluation views except for the tested component. The main comparison and default compression results use overflow-safe maps. Earlier ablations, qualitative figures, and timings retain their original serialization, with the precision check reported in the supplementary material.

4.2 Reconstruction Quality

Table 1: **Full-session reconstruction results across four datasets.** Average weights sequences equally. Best and second-best results within each pose-source group are bold and underlined.

Method	Metric	SG-Robot		DL3DV		GSIM		Wild-SLAM		Average
		Facade	Campus	Mean	Med.	Mean	Med.	Mean	Med.	
Self-estimated poses										
MonoGS (Matsuki et al., 2024)	PSNR ↑	17.44	<u>17.55</u>	18.71	17.92	20.46	21.11	23.89	25.62	20.40
	SSIM ↑	0.431	<u>0.499</u>	0.502	0.430	0.675	0.677	0.812	0.857	0.632
	LPIPS ↓	0.821	<u>0.795</u>	0.573	0.667	0.541	0.566	0.353	0.305	0.538
S3PO-GS (Cheng et al., 2025)	PSNR ↑	**20.63**	**18.91**	**21.53**	**21.11**	26.10	25.93	26.31	**28.67**	<u>24.06</u>
	SSIM ↑	**0.553**	**0.538**	<u>0.642</u>	<u>0.687</u>	0.807	0.818	0.892	**0.932**	0.748
	LPIPS ↓	<u>0.536</u>	**0.505**	0.361	0.316	0.264	0.231	0.202	0.176	0.308
Splat-SLAM (Sandström et al., 2024)	PSNR ↑	<u>18.57</u>	9.64	<u>20.44</u>	20.53	**29.44**	**30.75**	**26.75**	<u>28.60</u>	**24.35**
	SSIM ↑	0.491	0.225	**0.682**	**0.696**	**0.887**	**0.918**	<u>0.901</u>	**0.932**	**0.768**
	LPIPS ↓	**0.508**	0.920	**0.267**	**0.268**	**0.149**	**0.084**	**0.160**	**0.129**	**0.253**
WildGS-SLAM (Zheng et al., 2025)	PSNR ↑	18.50	12.92	19.13	<u>20.65</u>	<u>27.96</u>	<u>27.10</u>	<u>26.33</u>	27.27	23.54
	SSIM ↑	<u>0.548</u>	0.451	0.613	0.671	<u>0.867</u>	<u>0.869</u>	**0.912**	<u>0.919</u>	<u>0.760</u>
	LPIPS ↓	0.538	0.842	<u>0.326</u>	<u>0.279</u>	<u>0.186</u>	<u>0.126</u>	<u>0.161</u>	<u>0.172</u>	<u>0.280</u>
Shared VGGT-Omega poses										
MirrorGS (Meng et al., 2024)	PSNR ↑	12.92	13.68	14.70	14.68	17.27	16.92	17.33	17.92	16.06
	SSIM ↑	0.367	0.422	0.506	0.487	0.744	0.743	0.762	0.766	0.637
	LPIPS ↓	0.727	0.695	0.466	0.494	0.318	0.317	0.308	0.314	0.405
RTR-GS (Zhou et al., 2025b)	PSNR ↑	16.62	16.25	17.64	18.60	17.58	17.85	19.46	20.43	17.90
	SSIM ↑	0.444	0.519	0.553	0.561	0.702	0.698	0.813	0.840	0.658
	LPIPS ↓	0.794	0.737	0.381	0.430	0.409	0.401	0.329	0.281	0.424
RT-Splatting (Shi et al., 2026)	PSNR ↑	<u>20.43</u>	<u>21.16</u>	<u>24.11</u>	<u>23.38</u>	<u>31.69</u>	**32.18**	<u>28.45</u>	<u>30.75</u>	<u>27.41</u>
	SSIM ↑	<u>0.537</u>	<u>0.610</u>	<u>0.772</u>	<u>0.796</u>	<u>0.908</u>	**0.919**	<u>0.925</u>	<u>0.947</u>	<u>0.833</u>
	LPIPS ↓	<u>0.523</u>	<u>0.502</u>	<u>0.161</u>	<u>0.162</u>	**0.072**	**0.066**	<u>0.105</u>	<u>0.082</u>	<u>0.158</u>
SpecGuard	PSNR ↑	**22.90**	**21.21**	**26.30**	**26.52**	**32.26**	<u>32.08</u>	**32.37**	**33.81**	**29.33**
	SSIM ↑	**0.695**	**0.634**	**0.855**	**0.868**	**0.912**	<u>0.918</u>	**0.950**	**0.963**	**0.875**
	LPIPS ↓	**0.297**	**0.493**	**0.113**	**0.107**	<u>0.082</u>	<u>0.079</u>	**0.084**	**0.071**	**0.128**

Table 1 reports the highest aggregate fidelity for SpecGuard. Under shared VGGT-Omega poses, it exceeds RT-Splatting by 1.92 dB PSNR and 0.042 SSIM while reducing LPIPS by 0.030. Native-pose SLAM results are separate. Supplementary Tables S1–S4 give per-scene results.

PSNR improves over RT-Splatting by 2.47, 2.62, and 6.27 dB on SG-Robot Facade, DL3DV Business, and Wild Crowd. GSIM Office gains 2.89 dB, although WildGS-SLAM has higher PSNR under native poses. Conversely, RT-Splatting retains higher PSNR on GSIM Discussion and Lounge and lower GSIM mean LPIPS. These scene-dependent trade-offs motivate the controlled local-submap ablation in Section 4.3.

Figure 3 shows clearer canopy lettering in DL3DV Business and distinct window mullions on SG-Robot Facade, where baseline renders exhibit blur or scattered artifacts. Differences are subtler in GSIM Office blinds and Wild-SLAM Static 1 box labels. All SpecGuard views use the delivered compressed map.

4.3 Ablation Studies

Local submaps gain 1.00 dB over the strongest tested global control (Table 2a), led by Wild-SLAM (4.08 dB) and SG-Robot (1.70 dB). DL3DV gains 0.44 dB, while GSIM loses 0.81 dB, showing



Figure 3: **Qualitative comparison of full-session reconstruction.** SpecGuard more clearly preserves canopy lettering in DL3DV Business and window mullions on SG-Robot Facade. Splat-SLAM uses native poses, while RT-Splatting and Ours share VGGT-Omega poses.

Table 2: **Ablations of local submaps, scaffold initialization, and composition.** In (a), initialization variants use equal final Gaussian counts. In (b), LPIPS/L1 report scene-averaged 95th-percentile differences between adjacent renders, while PSNR measures observed-view fidelity.

(a) Local submaps and scaffold initialization

Dataset	PSNR ↑			ΔPSNR ↑		Seed red. (%) ↑
	Global	Window	**Ours**	vs. Global	vs. Window	
SG-Robot	20.37	21.80	**22.07**	+1.70	+0.27	57.95
DL3DV	25.90	25.09	**26.34**	+0.44	+1.25	0.41
GSIM	**33.08**	31.97	32.27	-0.81	+0.30	7.17
Wild-SLAM	28.43	30.10	**32.51**	+4.08	+2.41	0.00
Avg.	28.38	28.31	**29.38**	+1.00	+1.07	9.47

(b) Composition ablation

Dataset	Hard assignment			Pose blend			**Ours**		
	LPIPS ↓	L1 ↓	PSNR ↑	LPIPS ↓	L1 ↓	PSNR ↑	LPIPS ↓	L1 ↓	PSNR ↑
SG-Robot	0.332	0.086	**22.13**	**0.114**	0.049	22.07	**0.114**	**0.048**	22.07
DL3DV	0.148	0.065	26.33	0.066	**0.049**	26.27	**0.065**	**0.049**	**26.34**
GSIM	0.213	0.089	**33.02**	**0.198**	**0.079**	32.11	**0.198**	**0.079**	32.27
Wild-SLAM	0.081	0.023	**32.70**	**0.010**	**0.008**	32.43	**0.010**	**0.008**	32.51
Avg.	0.177	0.066	**29.70**	**0.105**	**0.050**	29.29	**0.105**	**0.050**	29.38

a scene-dependent trade-off between global parameter sharing and local specialization. The global controls vary Gaussian capacity and optimization steps, selecting the strongest tested setting separately for each scene. This comparison tests whether local submaps improve on additional resources within a single map, rather than only on its default budget. Across sequences outside Wild-SLAM, the mean gain is 0.059 dB: improvements on SG-Robot and DL3DV are largely offset by GSIM losses. The aggregate gain therefore describes a nonuniform benefit of local specialization, rather than a uniform advantage across scenes.

Figure 4 locates these differences along SG-Robot Campus and DL3DV Art Galleries, whose mean gains are 1.31 and 1.54 dB. Curves show per-view fidelity along each frozen-map trajectory. Matched crops expose facade and entrance detail blurred in the global render, complementing the averages.

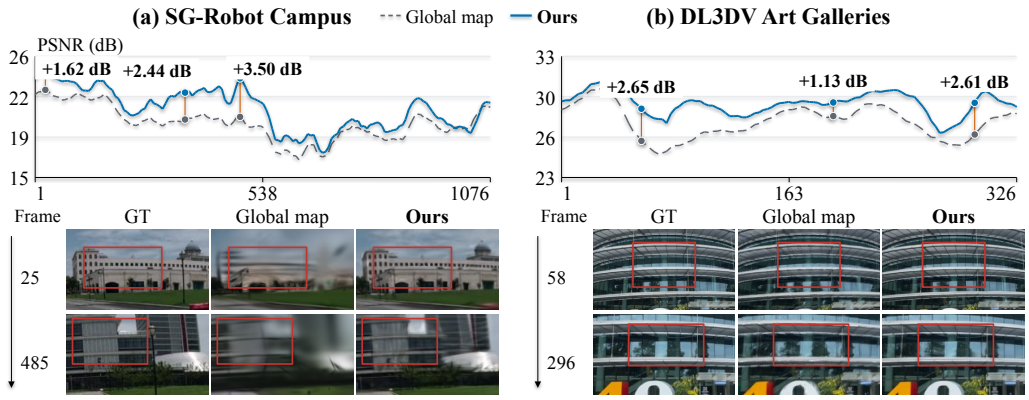


Figure 4: **Local submaps versus the strongest tested global Gaussian map.** On SG-Robot Campus and DL3DV Art Galleries, Ours achieves higher mean observed-view PSNR and preserves facade and entrance details blurred in the global render. Curves show 31-frame moving averages.

At equal final Gaussian counts, scaffold initialization improves PSNR by 1.07 dB over window seeding (Table 2a) while reducing seed counts by 9.47% on average. Wild-SLAM gains fidelity without fewer seeds, so the benefit cannot be attributed solely to seed reduction. Supplementary per-scene results retain memory, training time, and the quality cost on SG-Robot Campus. Selecting seed centers and DC coefficients from the optimized scaffold reuses information from the full session before window-wise optimization. The initialization gains on DL3DV and Wild-SLAM are 1.25 and 2.41 dB, respectively, at equal final Gaussian counts. This separates the quality comparison from changes in the number of rendered primitives.

With the maps frozen, Table 2(b) distinguishes transition continuity from observed-view fidelity. Hard assignment renders the highest-proximity local submap and gains 2.63 dB over the default global map, but changes branches abruptly at submap boundaries. Pose-supported soft composition reduces aggregate adjacent-render LPIPS from 0.177 to 0.105 and L1 from 0.066 to 0.050. Adding ray-wise coverage leaves both metrics unchanged at the reported precision, with no systematic additional continuity gain. Figure 5 links these aggregate results to the visible rendering change across a GSIM Discussion boundary.

Ray-wise coverage downweights a local submap where rasterized opacity is low. Replacing image-mean opacity with rasterized opacity improves PSNR by 0.09 dB overall and 0.156 dB near boundaries. Hard assignment still exceeds the full composition by 0.32 dB at the displayed precision on observed views, exposing a fidelity–continuity trade-off. Thus pose-supported soft composition supplies the principal continuity benefit, while ray-wise coverage recovers a smaller part of the observed-view fidelity. The continuity evaluation uses interpolated cameras between observation windows, while the fidelity evaluation compares renders with observed RGB. Together, these measurements distinguish smoother transitions from more accurate reconstruction.

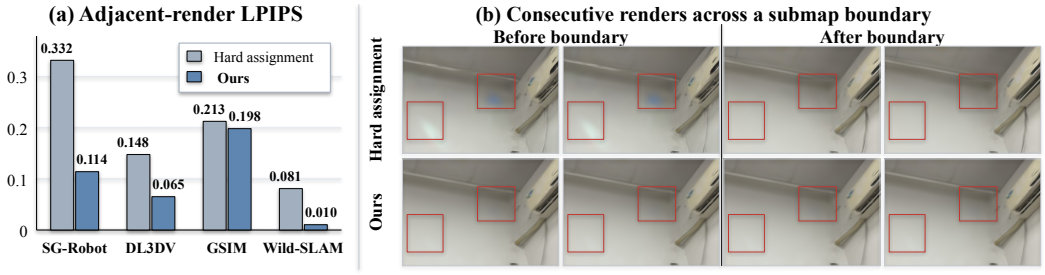


Figure 5: **Submap switching with hard assignment and pose-supported soft composition.** Soft composition reduces adjacent-render differences and suppresses the abrupt appearance change shown in GSIM Discussion. Bars average scene-level 95th-percentile adjacent-render LPIPS, and crops show consecutive interpolated views across one submap boundary.

Table 3: **Rendering efficiency and appearance compression.** We report dataset-level efficiency and rank/residual sensitivity. Storage reduction and metric differences are measured against the uncompressed atlas.

(a) Delivered system

Dataset	Map size (MiB) ↓	Reduction (%) ↑	ΔPSNR (dB) ↑	FPS ↑	Load (s) ↓	GPU (GiB) ↓
SG-Robot	434.4	40.78	-0.015	71.2	2.23	1.52
DL3DV	387.2	37.60	-0.039	82.3	1.99	1.29
GSIM	363.9	35.32	-0.015	103.9	1.60	1.16
Wild-SLAM	681.3	48.61	-0.168	62.5	3.10	2.66
Avg.	453.8	39.76	-0.058	84.0	2.14	1.59

(b) Rank and residual sensitivity (original serialization)

Configuration	Size (MiB) ↓	Reduction (%) ↑	ΔPSNR ↑	ΔSSIM ↑	ΔLPIPS ↓
Uncompressed SH	782.1	0.00	**0.000**	**+0.00000**	**+0.00000**
Rank 16, res. 0%	444.4	40.87	-0.162	-0.00075	+0.00200
Rank 16, res. 4% (default)	453.3	39.78	-0.058	-0.00044	+0.00105
Rank 4, res. 4%	**398.6**	**46.41**	-0.322	-0.00298	+0.00380
Rank 32, res. 4%	526.4	30.94	-0.010	-0.00007	+0.00025

4.4 Efficiency Analysis

Appearance factorization reduces storage by 39.8% with a 0.058 dB mean PSNR decrease (Table 3a), including the scaffold, submaps, calibration, and bases in map size. Wild-SLAM has both the largest reduction and loss (0.168 dB versus at most 0.039 dB elsewhere), revealing a scene-dependent quality–storage trade-off.

Raising rank from 4 to 32 lowers the PSNR decrease from 0.322 to 0.010 dB but reduces storage reduction from 46.41% to 30.94% (Table 3b). At rank 16, the residual recovers 0.104 dB for 8.9 MiB. Family grouping organizes shared bases but adds no fidelity gain over single shared basis or pose-only controls at the default setting (supplementary factorization-control table). The residual also reduces the LPIPS increase from 0.00200 to 0.00105 and the SSIM decrease from 0.00075 to 0.00044. Thus its reconstruction benefit extends beyond PSNR to structural and perceptual similarity.

With $M = 3$ active local submaps and the global fallback, the renderer averages 84.0 FPS for retrieval, rasterization, and composition. Cost remains scene-dependent: Wild-SLAM is slower and uses more memory than GSIM. The supplementary active-submap sweep isolates the effect of the active-local-submap limit on renderer throughput. The decoded atlas remains resident during rendering, with mean peak GPU allocation of 1.59 GiB. Loading and decoding take 2.14 s on average.

5 Conclusion and Discussion

We presented SpecGuard, a camera-manifold Gaussian atlas for view-dependent radiance field reconstruction. A shared Gaussian scaffold initializes trajectory-local submaps, allowing each submap to specialize to its local viewing distribution. Pose-supported soft composition coordinates their rendering, while query-aware appearance factorization reduces the storage of their non-DC SH coefficients. Full-session evaluation demonstrates improved aggregate reconstruction fidelity, reduced submap switching artifacts, and lower storage with a small reconstruction-quality cost.

Limitations. SpecGuard uses externally supplied poses, and its evaluation focuses on observed views and nearby trajectory interpolation. Local submaps do not uniformly outperform the strongest global controls, with several GSIM sequences favoring a single map. Soft composition trades some observed-view fidelity for smoother transitions, while scaffold initialization and appearance compression introduce additional quality–resource trade-offs. These findings motivate adaptive submap allocation and composition, together with evaluation under pose uncertainty and wider viewpoint changes.

AI USE STATEMENT

Generative AI tools were used to assist with code scaffolding, experiment automation, and language editing. The authors reviewed all scientific claims, experimental configurations, manuscript text, and generated artifacts. No generative AI tool was used to generate or alter target RGB images, metric values, or qualitative evidence. The authors take responsibility for the final content.

ETHICS STATEMENT

The SG-Robot sequences are private outdoor captures. Any public release requires appropriate licensing and privacy review. Otherwise, they must be replaced by a public reflective loop sequence. The method represents observational appearance and should not be used as a safety guarantee for robot navigation without independent geometric and collision-sensing safeguards.

REPRODUCIBILITY STATEMENT

The supplementary material specifies the fixed-pose protocol, data boundary, local-submap construction, pose-supported soft composition, hyperparameters, and ablations. The repository provides scripts to regenerate temporary local submaps, materialize the atlas, and render the Gaussian tensors used by the evaluator. Section 4 defines the full-session metric protocol.

REFERENCES

Timothy Chen, Ola Shorinwa, Joseph Bruno, Aiden Swann, Javier Yu, Weijia Zeng, Keiko Nagami, Philip Dames, and Mac Schwager. Splat-Nav: Safe real-time robot navigation in gaussian splatting maps, 2024. URL https://arxiv.org/abs/2403.02751.

Yu Chen and Gim Hee Lee. DOGS: Distributed-oriented gaussian splatting for large-scale 3D reconstruction via gaussian consensus. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper_files/paper/2024/hash/3d03b03197666b19c6a6e69812dd3e34-Abstract-Conference.html.

Chong Cheng, Sicheng Yu, Zijian Wang, Yifan Zhou, and Hao Wang. Outdoor monocular SLAM with global scale-consistent 3D gaussian pointmaps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Jixuan Fan, Wanhua Li, Yifei Han, Tianru Dai, and Yansong Tang. Momentum-GS: Momentum gaussian self-distillation for high-quality large scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 25250–25260, 2025. URL https://arxiv.org/abs/2412.04887.

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. 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.

Letian Huang, Dongwei Ye, Jialin Dan, Chengzhi Tao, Huiwen Liu, Kun Zhou, Bo Ren, Yuanqi Li, Yanwen Guo, and Jie Guo. TransparentGS: Fast inverse rendering of transparent objects with gaussians. *ACM Transactions on Graphics*, 44(4):1–17, 2025.

Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. GaussianShader: 3D gaussian splatting with shading functions for reflective surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. SplaTAM: Splat, track & map 3D gaussians for dense RGB-D SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://spla-tam.github.io/.

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.

Zhihao Liang, Qi Zhang, Ying Feng, Ying Shan, and Kui Jia. GS-IR: 3D gaussian splatting for inverse rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21644–21653, 2024. URL https://arxiv.org/abs/2311.16473.

Jiaqi Lin, Zhihao Li, Xiao Tang, Jianzhuang Liu, Shiyong Liu, Jiayue Liu, Yangdi Lu, Xiaofei Wu, Songcen Xu, Youliang Yan, and Wenming Yang. VastGaussian: Vast 3D gaussians for large scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5166–5175, 2024.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, Xuanmao Li, Xingpeng Sun, Rohan Ashok, Aniruddha Mukherjee, Hao Kang, Xiangrui Kong, Gang Hua, Tianyi Zhang, Bedrich Benes, and Aniket Bera. DL3DV-10K: A large-scale scene dataset for deep learning-based 3D vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Jiayue Liu, Xiao Tang, Freeman Cheng, Roy Yang, Zhihao Li, Jianzhuang Liu, Yi Huang, Jiaqi Lin, Shiyong Liu, Xiaofei Wu, Songcen Xu, and Chun Yuan. MirrorGaussian: Reflecting 3D gaussians for reconstructing mirror reflections. In *European Conference on Computer Vision*, 2024a.

Shiyong Liu, Xiao Tang, Zhihao Li, Yingfan He, Chongjie Ye, Jianzhuang Liu, Binxiao Huang, Shunbo Zhou, and Xiaofei Wu. OccluGaussian: Occlusion-aware gaussian splatting for large scene reconstruction and rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 26643–26652, 2025. URL https://arxiv.org/abs/2503.16177.

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*, 2024b. doi: 10.1007/978-3-031-72640-8_15. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/papers/02472.pdf.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3D gaussians: Tracking by persistent dynamic view synthesis. In *International Conference on 3D Vision*, 2024.

Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. 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*, pp. 7210–7219, 2021.

Hidenobu Matsuki, Riku Murai, Paul H. J. Kelly, and Andrew J. Davison. Gaussian splatting SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18039–18048, 2024.

Jiarui Meng, Haijie Li, Yanmin Wu, Qiankun Gao, Shuzhou Yang, Jian Zhang, and Siwei Ma. Mirror-3DGS: Incorporating mirror reflections into 3D gaussian splatting. In *IEEE International Conference on Visual Communications and Image Processing*, 2024. URL https://arxiv.org/abs/2404.01168.

Andréas Meuleman, Yu-Lun Liu, Chen Gao, Jia-Bin Huang, Changil Kim, Min H. Kim, and Johannes Kopf. Progressively optimized local radiance fields for robust view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16539–16548, 2023.

Taku Okawara, Ryo Nishibe, Mao Kasano, Kentaro Uno, and Kazuya Yoshida. 3D mapping using a lightweight and low-power monocular camera embedded inside a gripper of limbed climbing robots, 2025. URL https://arxiv.org/abs/2511.05816.

Erik Sandström, Keisuke Tateno, Michael Oechsle, Michael Niemeyer, Luc Van Gool, Martin R. Oswald, and Federico Tombari. Splat-SLAM: Globally optimized RGB-only SLAM with 3D gaussians. *arXiv preprint arXiv:2405.16544*, 2024.

Ji Shi, Xianghua Ying, Bowei Xing, Ruohao Guo, and Wenzhen Yue. RT-Splatting: Joint reflection-transmission modeling with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Seungjoo Shin, Jaesik Park, and Sunghyun Cho. Locality-aware gaussian compression for fast and high-quality rendering. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=dHYwfV2KeP.

Fangwen Shu, Paul Lesur, Yaxu Xie, Alain Pagani, and Didier Stricker. SLAM in the field: An evaluation of monocular mapping and localization on challenging dynamic agricultural environment. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1761–1771, 2021. URL https://arxiv.org/abs/2011.01122.

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*, pp. 8248–8258, 2022.

Jiajun Tang, Fan Fei, Zhihao Li, Xiao Tang, Shiyong Liu, Youyu Chen, Binxiao Huang, Zhenyu Chen, Xiaofei Wu, and Boxin Shi. SpecTRe-GS: Modeling highly specular surfaces with reflected nearby objects by tracing rays in 3D gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16133–16142, 2025.

Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer Petersson. NeuRAD: Neural rendering for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14895–14904, 2024.

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*, pp. 12922–12931, 2022.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-ω, 2026. URL https://arxiv.org/abs/2605.15195.

Yufei Wang, Zhihao Li, Lanqing Guo, Wenhan Yang, Alex C. Kot, and Bihan Wen. ContextGS: Compact 3D gaussian splatting with anchor level context model. In *Advances in Neural Information Processing Systems*, volume 37, 2024a. URL https://arxiv.org/abs/2405.20721.

Zhou Wang and Alan C. Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE Signal Processing Magazine*, 26(1):98–117, 2009. doi: 10.1109/MSP.2008.930649.

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. doi: 10.1109/TIP.2003.819861.

Zihan Wang, Shuzhe Wang, Matias Turkulainen, Junyuan Fang, and Juho Kannala. Gaussian splatting in mirrors: Reflection-aware rendering via virtual camera optimization. In *British Machine Vision Conference*, 2024b.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 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.

Yongchang Wu, Zipeng Qi, Zhenwei Shi, and Zhengxia Zou. BlockGaussian: Efficient large-scale scene novel view synthesis via adaptive block-based gaussian splatting, 2025. URL https://arxiv.org/abs/2504.09048.

Kunnong Zeng, Chensheng Peng, Yichen Xie, Masayoshi Tomizuka, and Cem Yuksel. RT-GS: Gaussian splatting with reflection and transmittance primitives. *arXiv preprint arXiv:2604.00509*, 2026.

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*, 2018.

Youjia Zhang, Anpei Chen, Yumin Wan, Zikai Song, Junqing Yu, Yawei Luo, and Wei Yang. Ref-GS: Directional factorization for 2D gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26483–26492, 2025. URL https://arxiv.org/abs/2412.00905.

Jianhao Zheng, Zihan Zhu, Valentin Bieri, Marc Pollefeys, Songyou Peng, and Iro Armeni. WildGS-SLAM: Monocular gaussian splatting SLAM in dynamic environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Hongxing Zhou, Juan Chen, and Zhiqing Li. Dynamic SLAM with 3-d gaussian splatting supporting monocular sensing. *IEEE Sensors Journal*, 25(12):21900–21910, 2025a. doi: 10.1109/JSEN.2025.3560481.

Yongyang Zhou, Fang-Lue Zhang, Zichen Wang, and Lei Zhang. RTR-GS: 3D gaussian splatting for inverse rendering with radiance transfer and reflection. In *Proceedings of the ACM International Conference on Multimedia*, 2025b.

SPECGUARD: A CAMERA-MANIFOLD GAUSSIAN ATLAS FOR RADIANCE FIELD RECONSTRUCTION SUPPLEMENTARY MATERIAL

Anonymous authors
Paper under double-blind review

1 COMPLETE PER-SCENE RENDERING RESULTS

Tables S1 through S4 expand the main comparison into all 17 scenes and all eight methods. Each scene reports PSNR, SSIM, and LPIPS from the completed full-session evaluation. The first four methods estimate their own poses. The reflection-aware methods and SpecGuard use the shared VGGT-Omega camera estimates. Global 3DGS is an internal ablation control, evaluated separately from the seven external baselines. MirrorGS denotes the unofficial Mirror-3DGS implementation at `https://github.com/TingtingLiao/MirrorGS`, based on revision `c599b0e5ed64` with the shared-pose full-session evaluation adapters. The main paper's overall mean weights all 17 scenes equally.

Table S1: **Full-session reconstruction results on SG-Robot.** Methods are grouped by pose source. SpecGuard uses the compressed atlas.

Method	Metric	Facade	Campus	Mean
Self-estimated poses				
MonoGS (Matsuki et al., 2024)	PSNR (dB) ↑	17.44	17.55	17.50
	SSIM ↑	0.431	0.499	0.465
	LPIPS ↓	0.821	0.795	0.808
S3PO-GS (Cheng et al., 2025)	PSNR (dB) ↑	**20.63**	**18.91**	**19.77**
	SSIM ↑	**0.553**	**0.538**	**0.546**
	LPIPS ↓	0.536	**0.505**	**0.521**
Splat-SLAM (Sandström et al., 2024)	PSNR (dB) ↑	18.57	9.64	14.11
	SSIM ↑	0.491	0.225	0.358
	LPIPS ↓	**0.508**	0.920	0.714
WildGS-SLAM (Zheng et al., 2025)	PSNR (dB) ↑	18.50	12.92	15.71
	SSIM ↑	0.548	0.451	0.500
	LPIPS ↓	0.538	0.842	0.690
Shared VGGT-Omega poses				
MirrorGS (Meng et al., 2024)	PSNR (dB) ↑	12.92	13.68	13.30
	SSIM ↑	0.367	0.422	0.395
	LPIPS ↓	0.727	0.695	0.711
RTR-GS (Zhou et al., 2025)	PSNR (dB) ↑	16.62	16.25	16.44
	SSIM ↑	0.444	0.519	0.482
	LPIPS ↓	0.794	0.737	0.766
RT-Splatting (Shi et al., 2026)	PSNR (dB) ↑	20.43	21.16	20.79
	SSIM ↑	0.537	0.610	0.574
	LPIPS ↓	0.523	0.502	0.512
SpecGuard	PSNR (dB) ↑	**22.90**	**21.21**	**22.06**
	SSIM ↑	**0.695**	**0.634**	**0.664**
	LPIPS ↓	**0.297**	**0.493**	**0.395**

2 PROTOCOL AND INFERENCE CONTRACT

SpecGuard is evaluated as a given-pose, full-session Gaussian mapping backend. Its input is an ordered RGB trajectory, camera intrinsics, and fixed camera-to-world poses produced by the declared geometric frontend. All posed session frames may participate in map construction, and the final map is rasterized at the session cameras. Evaluation therefore measures reconstruction of the

Table S2: **Full-session reconstruction results on DL3DV.** Methods are grouped by pose source. SpecGuard uses the compressed atlas.

Method	Metric	Art	Botanical	Business	Market 1	Market 2	Mean
Self-estimated poses							
MonoGS (Matsuki et al., 2024)	PSNR (dB) ↑	23.65	14.72	16.03	17.92	21.25	18.71
	SSIM ↑	0.756	0.228	0.428	0.430	0.670	0.502
	LPIPS ↓	0.328	0.704	0.667	0.745	0.420	0.573
S3PO-GS (Cheng et al., 2025)	PSNR (dB) ↑	24.35	16.21	**21.11**	**20.93**	**25.04**	**21.53**
	SSIM ↑	0.803	0.320	**0.687**	0.580	0.822	0.642
	LPIPS ↓	0.210	0.564	0.316	0.496	0.219	0.361
Splat-SLAM (Sandström et al., 2024)	PSNR (dB) ↑	**24.70**	**17.78**	17.56	20.53	21.62	20.44
	SSIM ↑	**0.825**	**0.446**	0.669	**0.696**	0.773	**0.682**
	LPIPS ↓	**0.154**	**0.406**	**0.268**	0.280	0.229	**0.267**
WildGS-SLAM (Zheng et al., 2025)	PSNR (dB) ↑	21.86	10.59	17.84	20.65	24.72	19.13
	SSIM ↑	0.739	0.147	0.645	0.671	**0.861**	0.613
	LPIPS ↓	0.240	0.641	0.330	**0.279**	**0.137**	0.326
Shared VGGT-Omega poses							
MirrorGS (Meng et al., 2024)	PSNR (dB) ↑	16.66	13.03	13.38	14.68	15.77	14.70
	SSIM ↑	0.676	0.264	0.486	0.487	0.620	0.506
	LPIPS ↓	0.341	0.606	0.494	0.497	0.392	0.466
RTR-GS (Zhou et al., 2025)	PSNR (dB) ↑	18.81	14.31	14.68	18.60	21.80	17.64
	SSIM ↑	0.652	0.324	0.465	0.561	0.763	0.553
	LPIPS ↓	0.312	0.495	0.430	0.471	0.197	0.381
RT-Splatting (Shi et al., 2026)	PSNR (dB) ↑	26.86	19.37	23.35	23.38	27.58	24.11
	SSIM ↑	0.874	0.548	0.796	0.748	0.893	0.772
	LPIPS ↓	**0.089**	0.279	0.162	0.204	**0.069**	0.161
SpecGuard	PSNR (dB) ↑	**29.05**	**21.69**	**25.97**	**26.52**	**28.27**	**26.30**
	SSIM ↑	**0.909**	**0.721**	**0.864**	**0.868**	**0.911**	**0.855**
	LPIPS ↓	0.091	**0.180**	**0.107**	**0.110**	0.077	**0.113**

Table S3: **Full-session reconstruction results on GSIM.** Methods are grouped by pose source. SpecGuard uses the compressed atlas.

Method	Metric	Discussion	Living	Lounge	Market	Office	Washroom	Mean
Self-estimated poses								
MonoGS (Matsuki et al., 2024)	PSNR (dB) ↑	14.43	24.04	16.70	18.17	24.80	24.61	20.46
	SSIM ↑	0.607	0.776	0.630	0.496	0.819	0.725	0.675
	LPIPS ↓	0.694	0.303	0.630	0.718	0.502	0.399	0.541
S3PO-GS (Cheng et al., 2025)	PSNR (dB) ↑	**17.72**	32.14	22.78	23.28	32.08	28.58	26.10
	SSIM ↑	0.666	0.940	0.813	0.667	0.932	0.823	0.807
	LPIPS ↓	0.535	0.062	0.257	0.382	0.143	0.206	0.264
Splat-SLAM (Sandström et al., 2024)	PSNR (dB) ↑	17.45	**36.49**	**25.18**	25.52	35.99	**35.98**	**29.44**
	SSIM ↑	0.687	**0.974**	**0.889**	0.860	0.965	**0.948**	**0.887**
	LPIPS ↓	0.525	**0.019**	**0.140**	0.123	0.041	**0.044**	**0.149**
WildGS-SLAM (Zheng et al., 2025)	PSNR (dB) ↑	15.38	35.28	23.27	**26.98**	**39.65**	27.22	27.96
	SSIM ↑	**0.689**	**0.974**	0.875	**0.862**	**0.983**	0.819	0.867
	LPIPS ↓	**0.506**	0.024	0.143	**0.109**	**0.028**	0.303	0.186
Shared VGGT-Omega poses								
MirrorGS (Meng et al., 2024)	PSNR (dB) ↑	16.92	15.83	16.10	16.97	20.90	16.92	17.27
	SSIM ↑	0.728	0.801	0.721	0.622	0.836	0.758	0.744
	LPIPS ↓	0.340	0.263	0.366	0.378	0.269	0.295	0.318
RTR-GS (Zhou et al., 2025)	PSNR (dB) ↑	16.32	13.17	20.08	21.33	15.17	19.38	17.58
	SSIM ↑	0.703	0.693	0.764	0.658	0.671	0.724	0.702
	LPIPS ↓	0.509	0.515	0.333	0.360	0.441	0.297	0.409
RT-Splatting (Shi et al., 2026)	PSNR (dB) ↑	**28.83**	35.53	**27.24**	26.55	35.77	36.20	31.69
	SSIM ↑	**0.887**	0.967	**0.884**	0.788	0.973	0.952	0.908
	LPIPS ↓	**0.107**	**0.017**	**0.120**	**0.141**	0.021	**0.025**	**0.072**
SpecGuard	PSNR (dB) ↑	27.67	**36.50**	26.56	**27.38**	**38.66**	**36.78**	**32.26**
	SSIM ↑	0.878	**0.971**	0.875	**0.805**	**0.983**	**0.958**	**0.912**
	LPIPS ↓	0.129	0.019	0.150	0.147	**0.017**	0.029	0.082

Table S4: **Full-session reconstruction results on Wild-SLAM.** Methods are grouped by pose source. SpecGuard uses the compressed atlas.

Method	Metric	Crowd	Static 1	ANYmal 1	Static 2	Mean
Self-estimated poses						
MonoGS (Matsuki et al., 2024)	PSNR (dB) ↑	15.98	28.33	23.64	27.59	23.89
	SSIM ↑	0.651	0.882	0.833	0.882	0.812
	LPIPS ↓	0.600	0.203	0.340	0.271	0.353
S3PO-GS (Cheng et al., 2025)	PSNR (dB) ↑	**16.44**	31.31	**26.03**	31.44	26.31
	SSIM ↑	0.758	0.946	**0.918**	0.947	0.892
	LPIPS ↓	0.343	0.113	0.198	0.154	0.202
Splat-SLAM (Sandström et al., 2024)	PSNR (dB) ↑	16.40	**32.14**	25.07	33.39	**26.75**
	SSIM ↑	0.774	**0.960**	0.905	0.964	0.901
	LPIPS ↓	0.311	**0.070**	**0.166**	0.092	**0.160**
WildGS-SLAM (Zheng et al., 2025)	PSNR (dB) ↑	16.05	30.47	24.07	**34.71**	26.33
	SSIM ↑	**0.839**	0.944	0.893	**0.974**	**0.912**
	LPIPS ↓	**0.237**	0.106	0.239	**0.061**	0.161
Shared VGGT-Omega poses						
MirrorGS (Meng et al., 2024)	PSNR (dB) ↑	15.35	17.84	18.14	18.01	17.33
	SSIM ↑	0.733	0.756	0.775	0.783	0.762
	LPIPS ↓	0.323	0.307	0.322	0.280	0.308
RTR-GS (Zhou et al., 2025)	PSNR (dB) ↑	15.10	21.87	21.60	19.27	19.46
	SSIM ↑	0.699	0.839	0.872	0.841	0.813
	LPIPS ↓	0.511	0.242	0.316	0.246	0.329
RT-Splatting (Shi et al., 2026)	PSNR (dB) ↑	17.87	34.02	27.48	34.42	28.45
	SSIM ↑	0.842	0.962	0.932	0.963	0.925
	LPIPS ↓	0.211	0.045	0.107	**0.058**	0.105
SpecGuard	PSNR (dB) ↑	**24.14**	**37.19**	**30.43**	**37.71**	**32.37**
	SSIM ↑	**0.898**	**0.976**	**0.950**	**0.977**	**0.950**
	LPIPS ↓	**0.152**	**0.044**	**0.084**	**0.058**	**0.084**

mapping observations with externally supplied poses. Held-out trajectory interpolation is evaluated separately in the additional experiments.

Reference formulation and evaluated implementation. The main paper presents compact reference equations. The results use importance-visibility seed scoring, sampled pose anchors with squared angular dissimilarity, centered calibration at temperature 0.7, and percentile-clipped family distances. The definitions below specify these evaluated choices.

The delivered map has one Gaussian scaffold and a set of trajectory-local Gaussian submaps. Ordered windows allocate construction observations, whereas query-time access uses pose anchors and receives $(\mathbf{T}, \mathbf{K})$ rather than a frame identifier. An ordered-window renderer is used only as a diagnostic upper bound and is excluded from the reported method.

Table S5: Method components and their roles during construction and querying.

Object	Status	Construction rule	Query-time role
Poses $\mathbf{T}_t$	fixed input	Declared posed RGB frontend	Camera input
Gaussian scaffold $\mathcal{G}$	optimized	Full-session reconstruction objective	Global fallback
Observation windows $\{W_k\}$	deterministic	Uniform ordered trajectory windows	Not queried by frame index
Frustum-incidence scores	computed	Scaffold coverage from the window cameras	Local seed point cloud
Local submaps $\mathcal{G}_k$	optimized	Standard 3DGS at the final population budget	Candidate representations
Viewpoint proximity s_k	computed	Sampled pose anchors and support scales	Candidate retrieval
Calibration parameters (β_k, κ_k)	calibrated	Observed-frame photometric calibration	Composition weights
Submap-family bases	factorized	Non-DC SH coefficients and sparse residuals	Map serialization

After construction, the query state consists only of Gaussian attributes, local coordinate transforms, pose-anchor metadata, calibration parameters, and optional appearance factorization data. Source images are not part of the serialized map.

3 Gaussian State and Scaffold-Seeded Submap Construction

The trainable state of one Gaussian is

$$g_i = (\boldsymbol{\mu}_i, \mathbf{q}_i, \boldsymbol{\ell}_i, \zeta_i, \mathbf{h}_i), \qquad \bar{\mathbf{q}}_i = \frac{\mathbf{q}_i}{\|\mathbf{q}_i\|_2}, \qquad \boldsymbol{\Sigma}_i = R(\bar{\mathbf{q}}_i)\,\mathrm{diag}(e^{2\boldsymbol{\ell}_i})R(\bar{\mathbf{q}}_i)^\top, \qquad \alpha_i = \sigma(\zeta_i).$$

Here $\boldsymbol{\mu}_i$ is the center, $\mathbf{q}_i$ is an orientation parameter normalized to the unit quaternion $\bar{\mathbf{q}}_i$, $\boldsymbol{\ell}_i$ is a three-dimensional log-scale vector, and ζ_i is the opacity logit. Covariance and opacity are therefore derived quantities rather than independent parameters. We separate the direction-independent DC coefficients from the non-DC SH coefficients,

$$\mathbf{h}_i = (\mathbf{h}_i^{(0)}, \mathbf{h}_i^{(>0)}), \qquad \mathbf{h}_i^{(0)} \in \mathbb{R}^3, \qquad \mathbf{h}_i^{(>0)} \in \mathbb{R}^{3((d_{\mathrm{sh}}+1)^2-1)},$$

where $d_{\mathrm{sh}} = 3$ in the default configuration, giving 45 non-DC coefficients. Here $\mathbf{h}_i$ corresponds to the full SH vector $\mathbf{c}_i$ in the main-paper setup, while its factorization uses only $\mathbf{h}_i^{(>0)}$. For viewing direction $\mathbf{v}$, the evaluated color is

$$\mathbf{c}_i(\mathbf{v}) = \mathrm{SH}_{d_{\mathrm{sh}}}\left(\mathbf{v}, \mathbf{h}_i^{(0)}, \mathbf{h}_i^{(>0)}\right),$$

that is, the sum of the degree-d_{sh} spherical-harmonic components. We denote rasterization by $(C, O) = \mathcal{R}(\mathcal{G}, T)$, where C is the rendered RGB image and O is rasterized opacity at camera pose T. Opacity describes projected coverage, not ground-truth visibility. The direction $\mathbf{v}_i$ used below is the normalized viewing direction evaluated by the rasterizer. The distinction between primitive opacity and projected pixel alpha is maintained throughout rasterization.

If the projected Gaussians contributing to pixel $\mathbf{u}$ are ordered by depth as $i = 1, \ldots, m(\mathbf{u})$, the compositor is

$$\begin{aligned} \gamma_i(\mathbf{u}) &= \mathcal{T}_i(\mathbf{u})a_i(\mathbf{u}), & \mathcal{T}_i(\mathbf{u}) &= \prod_{j<i}\big(1 - a_j(\mathbf{u})\big), \\ \mathbf{C}_{\mathcal{G}}(\mathbf{u}) &= \sum_{i=1}^{m(\mathbf{u})} \gamma_i(\mathbf{u})\mathbf{c}_i(\mathbf{v}_i) + \mathcal{T}_{m(\mathbf{u})+1}(\mathbf{u})\mathbf{c}_{\mathrm{bg}}, \\ O_{\mathcal{G}}(\mathbf{u}) &= 1 - \mathcal{T}_{m(\mathbf{u})+1}(\mathbf{u}). \end{aligned} \tag{1}$$

Here $a_i(\mathbf{u})$ is the rasterized pixel alpha and $\mathbf{c}_{\mathrm{bg}}$ is the background color, whereas $\alpha_i = \sigma(\zeta_i)$ is the primitive opacity. The two quantities are not interchangeable.

The Gaussian scaffold is trained once in the normalized posed-camera coordinate frame. For scaffold mean $\boldsymbol{\mu}_i$ and camera-to-world pose $\mathbf{T}_t = [\mathbf{R}_t \mid \mathbf{o}_t]$, define its camera-space position, pixel projection, and frustum-incidence indicator as

$$\begin{aligned} \mathbf{x}_{it} &= \mathbf{R}_t^\top(\boldsymbol{\mu}_i - \mathbf{o}_t) = (x_{it}, y_{it}, z_{it})^\top, & \mathbf{p}_{it} &= \mathrm{proj}(\mathbf{K}_t\mathbf{x}_{it}), \\ v_{it}^{\mathrm{fr}} &= \mathbf{1}[z_{it} > z_{\min}]\,\mathbf{1}\left[\mathbf{p}_{it} \in \mathcal{U}_t^{\delta}\right], & V_{ik} &= \sum_{t \in W_k} v_{it}^{\mathrm{fr}}. \end{aligned} \tag{2}$$

Here $\mathrm{proj}([x, y, z]^\top) = (x/z, y/z)^\top$ is perspective division, $\mathcal{U}_t^{\delta}$ is the image domain expanded by boundary margin δ, and $z_{\min} = 0.02$ is the positive-depth threshold. The quantity V_{ik} is a view-frustum incidence count: it measures repeated camera coverage without invoking depth-sorted occlusion.

Using the frustum-incidence count V_{ik}, the main-paper reference score is

$$v_i^{(k)} = \alpha_i \frac{V_{ik}}{|W_k|}. \tag{3}$$

Here α_i is the scaffold primitive's opacity and $|W_k|$ is the number of observations in the window. Thus $V_{ik}/|W_k|$ is the fraction of window frusta containing its center. The evaluated implementation instead ranks eligible primitives using importance-weighted incidence:

$$\begin{aligned} I_i &= \alpha_i \sqrt{\bar{\ell}_i} \sqrt{e_i^{\mathrm{DC}} + 0.25 e_i^{\mathrm{SH}} + 10^{-6}}, \\ q_i^{(k)} &= I_i \left(0.5 + 0.5 \frac{V_{ik}}{\max(1, \max_{j \in E_k} V_{jk})} \right). \end{aligned} \tag{4}$$

Here $\bar{\ell}_i$ is the mean activated scale, e_i^{DC} is the DC coefficient norm, and e_i^{SH} is the root-mean-square non-DC coefficient magnitude. The first two quantities are floored at 10^{-8}. The eligible set E_k contains finite centers with opacity at least 0.005, within the coordinate-wise 1st–99th percentile bounds of its finite, opacity-filtered centers, and visible in at least two window frusta with a 5% image margin. When more than 200,000 candidates qualify, we retain the highest scores. Otherwise all eligible candidates are retained. The seed comparison holds these filters and capacity fixed and changes only the ranking score. The selected centers and DC colors define the initial point set $\mathcal{X}_k^{(0)}$, from which the local Gaussian parameters are initialized. This transfer gives all local submaps a spatial initialization derived from the shared scaffold. Their Gaussian states are subsequently optimized on the corresponding observation windows.

Let the fixed global-to-local similarity be $\mathbf{A}_k = (\varsigma_k, \mathbf{Q}_k, \mathbf{b}_k)$, where $\varsigma_k > 0$, $\mathbf{Q}_k \in SO(3)$, and $\mathbf{b}_k \in \mathbb{R}^3$. Its action on a camera-to-world pose is

$$\Phi_{\mathbf{A}_k}(\mathbf{T}_t) = \begin{bmatrix} \mathbf{Q}_k \mathbf{R}_t & \varsigma_k \mathbf{Q}_k \mathbf{o}_t + \mathbf{b}_k \\ \mathbf{0}^\top & 1 \end{bmatrix}, \qquad t \in W_k.$$

The induced action on a Gaussian mean and covariance is

$$\boldsymbol{\mu}_{ki} = \varsigma_k \mathbf{Q}_k \boldsymbol{\mu}_i + \mathbf{b}_k, \qquad \boldsymbol{\Sigma}_{ki} = \varsigma_k^2 \mathbf{Q}_k \boldsymbol{\Sigma}_i \mathbf{Q}_k^\top.$$

This preserves a valid camera rotation while consistently rescaling the spatial support. The transform is fixed during local optimization and query calibration.

4 Capacity-Controlled Local Gaussian Optimization

Each local submap uses all frames in its observation window and the same reconstruction objective as the Gaussian scaffold:

$$\begin{aligned} \mathcal{L}_{\mathrm{rec}} =& \lambda_{\mathrm{rgb}} \mathcal{E}_{\mathrm{rgb}} + \lambda_{\mathrm{ssim}} \mathcal{E}_{\mathrm{ssim}} + \lambda_{\mathrm{opacity}} \mathcal{E}_{\mathrm{opacity}} + \lambda_{\mathrm{scale}} \mathcal{E}_{\mathrm{scale}}, \\ \mathcal{E}_{\mathrm{rgb}} =& \frac{1}{3HW} \sum_{\mathbf{u},c} \left| C_c(\mathbf{u}) - I_c(\mathbf{u}) \right|, \qquad \mathcal{E}_{\mathrm{ssim}} = 1 - \mathrm{SSIM}(\mathbf{C}, \mathbf{I}), \\ \mathcal{E}_{\mathrm{opacity}} =& \frac{1}{N} \sum_{i=1}^{N} \sigma(\zeta_i), \qquad \mathcal{E}_{\mathrm{scale}} = \frac{1}{3N} \sum_{i=1}^{N} \sum_{d=1}^{3} e^{\ell_{id}}. \end{aligned} \tag{5}$$

Here H and W are image height and width, c indexes the RGB channels, N is the number of primitives in the representation being optimized, $\lambda_{\mathrm{rgb}} = 1 - \lambda_{\mathrm{ssim}}$, and the opacity and scale terms are evaluated on the current Gaussian state. In the default setting, $\lambda_{\mathrm{ssim}} = 0.2$, $\lambda_{\mathrm{opacity}} = 0.01$, and $\lambda_{\mathrm{scale}} = 0.1$. The local population is constrained by the declared final capacity during optimization. In the default configuration, the window span is $L = 20$, the maximum local population is 200,000, the local optimization budget is 7,000 steps, and the SH degree is three. We use a standard population-based 3DGS optimizer for the Gaussian state. The rasterization and optimization settings are held fixed across scenes.

The population bound is enforced throughout local optimization, so the delivered local-submap capacity is explicit.

5 Pose-Supported Soft Composition

Each local submap k stores a pose set $\mathcal{P}_k$ sampled from its observation window. All window poses are retained up to 32 anchors. Longer windows use rounded, uniformly spaced frame indices, not

pose clusters. With the default windows, the anchor sets contain 19 or 20 poses. For query pose $\mathbf{T}_q$, the evaluated viewpoint proximity is

$$s_k(\mathbf{T}_q) = \max_{\mathbf{T}_i \in \mathcal{P}_k} \exp\left[-\frac{\|\mathbf{t}(\mathbf{T}_q) - \mathbf{t}_i\|_2^2}{2\sigma_{t,k}^2} - \frac{(1 - \mathrm{clip}(\mathbf{v}(\mathbf{T}_q)^\top \mathbf{v}_i, -1, 1))^2}{2\sigma_{v,k}^2} \right]. \tag{6}$$

Here $(\mathbf{t}_i, \mathbf{v}_i)$ denotes the position and unit viewing direction of anchor pose $\mathbf{T}_i$. The scales $\sigma_{t,k}$ and $\sigma_{v,k}$ are estimated from each anchor set, as detailed below. Unlike the main-paper reference kernel, this implementation squares the angular dissimilarity. Changing its power at fixed bandwidth changes the response shape and is not an equivalent parameterization. The three highest-scoring local submaps form the active set $\mathcal{A}(\mathbf{T}_q)$.

Let m_t be the median nonzero distance between successive anchor positions and e_t the diagonal length of their axis-aligned bounding box. We set $\sigma_{t,k} = \max(3m_t, 0.18e_t, 10^{-3})$. For directions, m_v is the median nonzero successive dissimilarity $1 - \mathbf{v}_i^\top \mathbf{v}_{i+1}$, and e_v is the 90th percentile of $1 - \mathbf{v}_i^\top \mathbf{m}_v$, with $\mathbf{m}_v$ the unnormalized mean of unit anchor directions. We use $\sigma_{v,k} = \max(3m_v, 0.45e_v, 10^{-3})$. Dot products are clipped to $[-1, 1]$. Steps no greater than 10^{-8} are excluded, with an empty median set to zero. A single-anchor window uses spatial and directional scales of 0.05 and 0.12, respectively.

For each pixel, let $\widetilde{O}_b = \max(O_b, \epsilon)$ and $\widetilde{s}_k = \max(s_k, \epsilon)$, where ϵ is the machine epsilon of the rendered-color dtype. The implemented branch logits and weights are

$$\begin{aligned} \eta_0(\mathbf{u}) &= \log \widetilde{O}_0(\mathbf{u}), \\ \eta_k(\mathbf{u}) &= \log \widetilde{O}_k(\mathbf{u}) + \log \widetilde{s}_k + \beta_k + \kappa_k(\widetilde{s}_k - \tfrac{1}{2}), \\ w_b(\mathbf{u}) &= \mathrm{softmax}_b(\boldsymbol{\eta}(\mathbf{u})/\tau), \qquad \tau = 0.7. \end{aligned} \tag{7}$$

These weights differ from the main-paper reference rule in their floors, centering, and temperature. A bias shift can absorb centering for fixed parameters but changes the initialization and the quadratic calibration penalty, so it does not establish equivalence of the optimization procedures. The pre-clamp mixture is $\widehat{\mathbf{C}} = \sum_b w_b \mathbf{C}_b$, and the rendered image is $\widehat{\mathbf{C}}_{\mathrm{out}} = \mathrm{clip}(\widehat{\mathbf{C}}, 0, 1)$. The derivatives of the pre-clamp mixture are

$$\begin{aligned} \frac{\partial \widehat{\mathbf{C}}}{\partial \beta_k} &= \frac{w_k}{\tau}(\mathbf{C}_k - \widehat{\mathbf{C}}), \\ \frac{\partial \widehat{\mathbf{C}}}{\partial \kappa_k} &= (\widetilde{s}_k - \tfrac{1}{2})\frac{w_k}{\tau}(\mathbf{C}_k - \widehat{\mathbf{C}}). \end{aligned}$$

The output clamp masks gradients in saturated RGB channels. Query calibration minimizes the observed-RGB loss with a quadratic parameter penalty:

$$\mathcal{L}_q = \rho(\widehat{\mathbf{C}}_{\mathrm{out}}, \mathbf{I}) + \lambda_q \frac{1}{2K} \sum_{k=1}^{K} \left(\beta_k^2 + \kappa_k^2\right). \tag{8}$$

The first term is the mean Charbonnier RGB penalty

$$\rho(\widehat{\mathbf{C}}_{\mathrm{out}}, \mathbf{I}) = \frac{1}{3HW} \sum_{\mathbf{u},c} \sqrt{(\widehat{C}_{\mathrm{out},\mathbf{u},c} - I_{\mathbf{u},c})^2 + \epsilon_q}, \qquad \epsilon_q = 10^{-6}, \tag{9}$$

with $\mathbf{u}$ ranging over the $H \times W$ pixels and c over the three RGB channels. The second term in Eq. 8 is the corresponding ℓ_2 penalty on the two scalars per local submap. During query calibration, the Gaussian tensors are frozen and only the calibration parameters receive gradients. These parameters are then fixed for rendering.

At query time, the renderer uses the global fallback and the retrieved local submaps under the composition rule above. It does not use observation-window indices, source RGB, residual images, or metric-selected branches.

6 QUERY-AWARE APPEARANCE FACTORIZATION

For local submap k, let $\boldsymbol{\theta}_k = (\beta_k, \kappa_k)^\top$, let $\bar{\mathbf{t}}_k$ be the mean anchor position, and let $\bar{\mathbf{v}}_k$ be the normalized mean viewing direction. We form the pairwise distances

$$\widetilde{d}_\theta(k,l) = \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_l\|_2, \qquad \widetilde{d}_t(k,l) = \|\bar{\mathbf{t}}_k - \bar{\mathbf{t}}_l\|_2,$$
$$\widetilde{d}_v(k,l) = 1 - \bar{\mathbf{v}}_k^\top \bar{\mathbf{v}}_l.$$

Each distance matrix is divided by its 95th percentile, floored at 10^{-8}, and clipped to $[0,1]$, yielding d_θ, d_t, and d_v. Local submaps are then grouped by

$$d(k,l) = 0.50 d_\theta(k,l) + 0.30 d_t(k,l) + 0.20 d_v(k,l).$$

The percentile includes all finite matrix entries, including the diagonal and both symmetric halves. Direction dot products are clipped to $[-1,1]$, and mean-direction normalization uses a norm floor of 10^{-8}. This is the evaluated normalization of the main paper's raw-distance reference formulation, not an algebraically equivalent rescaling. The target family count is $F = \lceil \sqrt{K} \rceil$, bounded by the number of local submaps, following a fixed storage policy. For a family f, the degree-three non-DC SH tensor has dimension $3((3+1)^2 - 1) = 45$. A rank-r PCA model is

$$\widehat{\mathbf{h}}_{ki}^{>0} = \bar{\mathbf{h}}_f + \mathbf{B}_f \widehat{\boldsymbol{\xi}}_{ki} + \widehat{\boldsymbol{\varepsilon}}_{ki}, \qquad r = 16. \tag{10}$$

The basis is fitted from the family coefficient covariance and is orthonormal, with $\mathbf{B}_f \in \mathbb{R}^{45 \times r}$ and $\boldsymbol{\xi}_{ki} \in \mathbb{R}^r$. Before quantization, projection gives $\boldsymbol{\xi}_{ki} = \mathbf{B}_f^\top (\mathbf{h}_{ki}^{>0} - \bar{\mathbf{h}}_f)$. Hats denote decoded quantities. The per-Gaussian coordinates are quantized per coordinate to signed 8-bit integers:

$$\mathbf{z}_{ki} = \mathrm{clip}_{[-127,127]} \,\mathrm{round}(\boldsymbol{\xi}_{ki} \oslash \boldsymbol{\delta}_k), \qquad \widehat{\boldsymbol{\xi}}_{ki} = \boldsymbol{\delta}_k \odot \mathbf{z}_{ki}.$$

Here $\boldsymbol{\delta}_k \in \mathbb{R}^r$ contains the per-coordinate quantization scales, and $\oslash$ and $\odot$ denote elementwise division and multiplication, respectively. The residual is computed after coordinate quantization. Its highest-energy rows are retained and quantized to signed 8-bit integers, with per-coordinate scales. Before rasterization, the coordinates are dequantized and the sparse residual is restored to the native non-DC SH tensor. Geometry, opacity, DC coefficients, local coordinate transforms, and pose anchors are copied unchanged. The decoded appearance uses the same rasterizer and composition rule.

7 DATA BOUNDARY AND REPRODUCIBILITY

Table S6: Information available at each stage of the full-session protocol.

Stage	Construction access	Metric-time access
Pose frontend	Ordered RGB session	Fixed poses and intrinsics
Gaussian scaffold and local submaps	Posed RGB frames and fixed camera parameters	Frozen Gaussian tensors
Query calibration	Observed mapping frames and frozen Gaussian renders	Fixed calibration parameters
Appearance factorization	Non-DC SH coefficients and pose metadata	Serialized means, bases, coordinates, residuals, and quantization scales
Evaluation	No parameter update	Target RGB read-only for PSNR, SSIM, and LPIPS

Evaluation first renders from the serialized map and target camera, and then accesses the target RGB only to compute PSNR, SSIM, and LPIPS. The pose source, observation-window schedule, capacity, optimization budget, composition temperature, and factorization settings are recorded for every scene.

Implementation settings. The default setting uses $\tau = 0.7$, three active local submaps, a local capacity of 200,000 Gaussians, degree-three SH coefficients, and rank-16 query-aware appearance factorization with a four-percent sparse residual for the non-DC SH coefficients. Rasterized opacity and support thresholds are fixed construction constants and do not encode reflection or transmission semantics.

Hardware and software. The experiments use the repository's gsplat-based rasterizer and recorded environment. Each formal system manifest stores the source commit, input provenance, checkpoint hashes, frame count, and integrity flags. Qualitative renders are generated by the same serialized Gaussian map and are not used to compute the numerical tables.

8 Additional Experiments

We expand the main-paper ablations into scene-level comparisons, followed by storage sensitivity, runtime, and diagnostic experiments. Each diagnostic specifies its evaluated subset separately from the full-benchmark results. Bold values denote the best results within each comparison group. Where shown, underlining denotes second-best results. Ties at displayed precision share the same marking. Comparisons in reconstruction tables are grouped by pose source. Differences between displayed scores use their displayed precision. Directly reported metric changes, confidence intervals, and mean per-scene reductions are computed before rounding.

8.1 Data Acquisition

The self-collected SG-Robot data comprise two monocular video sequences, named Facade and Campus after their dominant visual content. They were recorded using a ground robot equipped with a Pinyou PY-C0217T030-V1.0 camera and an onboard NVIDIA Orin computer. The sequences capture glass building facades during robot motion and complement the public datasets with robot-acquired video.

8.2 Resource-Matched Global Controls

We compare four single-map controls that vary the Gaussian cap and optimizer update count while retaining one global map (Table S7). The representation comparison uses the strongest control for each scene.

Paired block-bootstrap intervals characterize within-sequence variation on a six-scene subset (Table S8 and Figure S1). Matched views illustrate local-detail differences in Wild Crowd, SG-Robot Campus, and DL3DV Art Galleries (Figures S2 and S3).

Table S7: Single-map controls and local submaps. PSNR (dB), higher is better. Best values per scene are bold. Caps and updates are allocated budgets, not measured runtime.

Scene	Default	Cap	Steps	Cap + steps	**Ours**
Art Gallery	26.10	27.19	26.98	27.57	**29.11**
Botanical Garden	20.34	21.15	20.95	21.70	**21.73**
Business Center	23.69	24.74	24.76	25.28	**25.99**
Supermarket 1	24.64	25.70	25.38	25.63	**26.54**
Supermarket 2	27.08	28.38	28.73	**29.23**	28.32
GSIM Discussion	27.35	27.83	28.10	**28.55**	27.66
Living Room	36.06	36.66	36.77	**37.36**	36.54
Lounge	26.29	26.73	26.56	**26.99**	26.57
Market	26.59	27.48	27.29	**27.80**	27.40
GSIM Office	38.31	38.85	39.00	**39.57**	38.66
Washroom	36.50	37.84	37.57	**38.21**	36.80
SG-Robot Campus	14.43	14.31	19.91	19.57	**21.22**
SG-Robot Facade	20.83	11.44	18.82	18.52	**22.92**
Wild Crowd	17.59	12.08	18.06	15.90	**24.15**
Static 1	33.23	34.23	32.55	18.09	**37.44**
ANYmal 1	27.08	27.01	27.31	21.59	**30.44**
Wild Static 2	34.13	14.87	34.05	20.28	**38.02**

Table S8: **Single-map controls under different Gaussian and optimization budgets.** PSNR intervals use paired 20-frame block bootstrap.

Method	SG-Robot Campus	GSIM Discussion	GSIM Office
Global (30k steps)	14.43	27.35	38.31
Global (matched Gaussian cap)	14.31	27.83	38.85
Global (matched steps)	19.91	28.10	39.00
Global (matched cap and steps)	19.57	**28.55**	**39.57**
SpecGuard	**21.22**	27.66	38.66
Δ vs. best Global	+1.31	-0.89	-0.91
95% interval	[+1.08, +1.54]	[-1.25, -0.48]	[-1.15, -0.65]

Method	SG-Robot Facade	Wild Crowd	Wild Static 2
Global (30k steps)	20.83	17.59	34.13
Global (matched Gaussian cap)	11.44	12.08	14.87
Global (matched steps)	18.82	18.06	34.05
Global (matched cap and steps)	18.52	15.90	20.28
SpecGuard	**22.92**	**24.15**	**38.02**
Δ vs. best Global	+2.09	+6.09	+3.89
95% interval	[+1.88, +2.31]	[+5.65, +6.53]	[+3.43, +4.35]

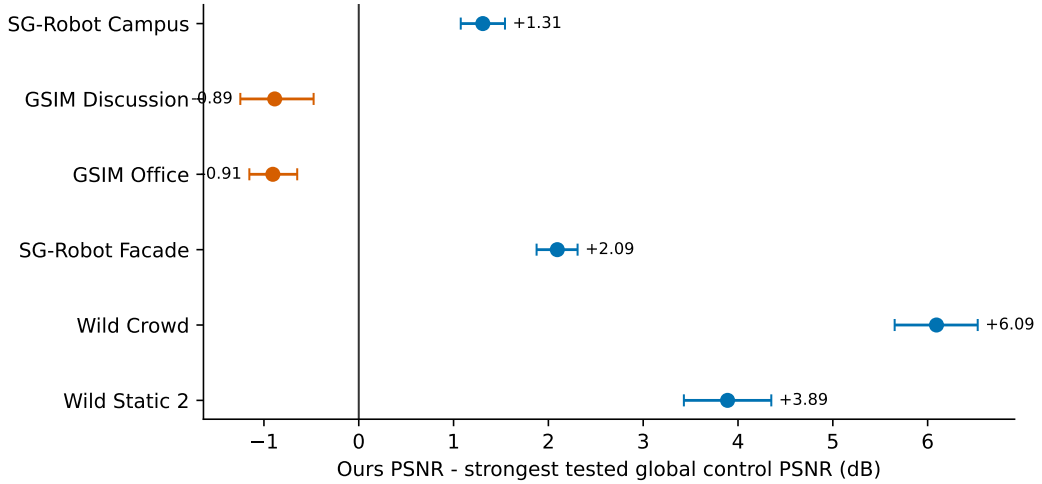


Figure S1: **Local submaps versus the strongest tested global Gaussian map.** PSNR differences vary across scenes. Error bars are paired 20-frame block-bootstrap 95% intervals.

8.3 Scaffold-Seeded Construction Details

We compare initialization at equal final Gaussian counts (Table S9). Seed reduction measures the decrease in materialized input points. Tables S11 and S12 provide a three-scene quality and measured-resource diagnostic. The resource table reports the local seed point clouds, peak local-training memory, and mean construction time from the corresponding construction runs.

8.4 Composition Details

Table S13 evaluates composition variants on the same frozen global Gaussian map and local submaps. The target-RGB oracle is a diagnostic upper bound and is not part of the method. Pose blend replaces each local submap's rasterized opacity by its image-wide mean while retaining the same candidate submaps, viewpoint proximity, calibration parameters, and global fallback. Pose



Figure S2: **Global and local reconstruction on Wild Crowd.** The scene includes moving people and a rigid background. Absolute-error maps use a fixed $[0, 0.25]$ scale.

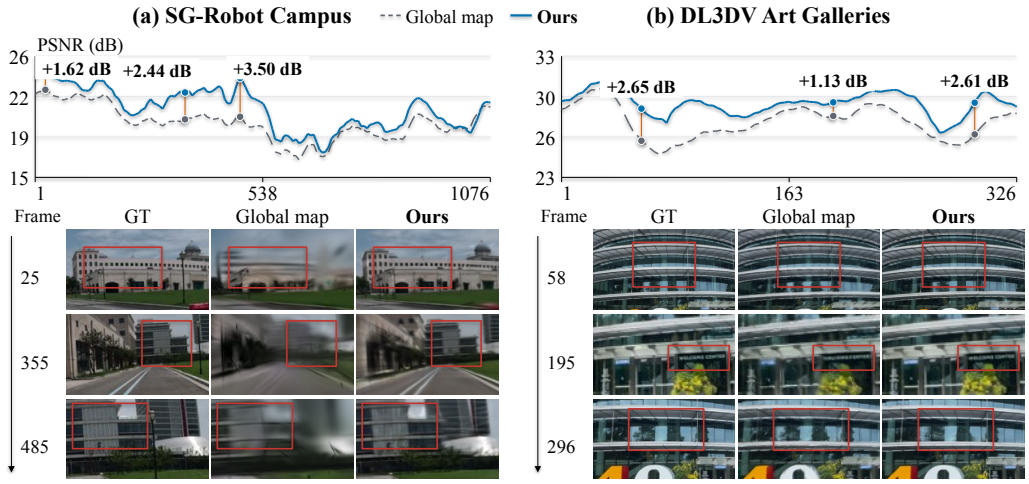


Figure S3: **Local-submap comparisons on SG-Robot Campus and DL3DV Art Galleries.** Observed-view PSNR curves and three matched crops per scene compare Ours with the strongest tested global Gaussian map. Curves show 31-frame moving averages.

Table S9: Initialization at equal final Gaussian counts within each scene. The PSNR difference is scaffold minus window. Seed reduction concerns the input point clouds.

Scene	Window PSNR ↑	Scaffold PSNR ↑	ΔPSNR	Seed reduction (%)
Art Gallery	27.52	29.11	+1.59	0.00
Botanical Garden	20.95	21.73	+0.78	0.00
Business Center	24.29	25.99	+1.70	0.00
Supermarket 1	25.28	26.54	+1.26	2.04
Supermarket 2	27.39	28.32	+0.93	0.00
GSIM Discussion	27.56	27.66	+0.10	9.49
Living Room	36.24	36.54	+0.30	0.00
Lounge	26.35	26.57	+0.22	26.34
Market	26.77	27.40	+0.63	7.21
GSIM Office	38.35	38.66	+0.31	0.00
Washroom	36.52	36.80	+0.28	0.00
SG-Robot Campus	21.99	21.22	-0.77	98.46
SG-Robot Facade	21.61	22.92	+1.31	17.44
Wild Crowd	23.29	24.15	+0.86	0.00
Static 1	34.06	37.44	+3.38	0.00
ANYmal 1	28.42	30.44	+2.02	0.00
Wild Static 2	34.61	38.02	+3.41	0.00

blend and full composition share frozen calibration parameters. Hard assignment selects one local submap without calibrated blending. The no-calibration variant sets both calibration parameters to zero, while slope removal retains the learned bias and sets the slope to zero without refitting. Within each scene, adjacent-render differences from all sampled boundaries are pooled before taking the

Table S10: Scaffold-seeded construction at equal delivered Gaussian counts. Positive resource reductions indicate lower scaffold-seeded cost, while PSNR change is measured against per-window dense initialization.

Scene	PSNR (dB)			Resource reduction (%) ↑		
	Window	Scaffold	Δ	Seeds	GPU	Time
SG-Robot Campus	21.99	21.22	-0.77	+98.5	+81.7	+67.4
GSIM Discussion	27.56	27.66	+0.10	+9.5	+2.5	-1.1
GSIM Office	38.35	38.66	+0.31	+0.0	+0.5	+2.5

Table S11: **Local-submap initialization at equal final Gaussian counts.** Counts are 3.77M for SG-Robot Campus, 2.60M for GSIM Discussion, and 3.00M for GSIM Office.

Method	SG-Robot Campus			GSIM Discussion			GSIM Office		
	PSNR (dB) ↑	SSIM ↑	LPIPS ↓	PSNR (dB) ↑	SSIM ↑	LPIPS ↓	PSNR (dB) ↑	SSIM ↑	LPIPS ↓
Window seed point cloud	**21.99**	**0.652**	**0.342**	27.56	0.874	0.133	38.35	0.982	0.018
Scaffold seed point cloud	21.22	0.634	0.492	**27.66**	**0.878**	**0.129**	**38.66**	**0.983**	**0.017**

Table S12: Local-construction resource measurements. Seeds are materialized local seed points. Memory is peak local-training GPU allocation and time is the mean per submap. These construction-run measurements are separate from the equal-delivered-size quality comparison.

Scene	Initialization	Seeds (M)	GPU (GiB)	Time (s)
SG-Robot Campus	Window seed point cloud	10.80	1.524	328.5
	Scaffold seed point cloud	0.17	0.279	107.1
GSIM Discussion	Window seed point cloud	2.60	1.424	228.3
	Scaffold seed point cloud	2.35	1.389	230.8
GSIM Office	Window seed point cloud	3.00	1.496	389.7
	Scaffold seed point cloud	3.00	1.489	379.8

95th percentile. Avg. averages those scene statistics. Each window boundary is sampled at nine interpolated cameras, yielding eight adjacent-render pairs. The qualitative example uses four consecutive samples from a denser 41-camera interpolation of one boundary. Interpolated views have no target RGB, so these statistics measure temporal change between adjacent renders. Pose blend accounts for most of the reduction (Table S14).

Table S13: **Composition ablation with frozen maps.** The slope-removal variant is not refitted. The target-RGB oracle is excluded from ranking.

Composition	PSNR (dB) ↑	SSIM ↑	LPIPS ↓
Global Gaussian map	27.07	0.834	0.182
Hard assignment	**29.70**	**0.883**	**0.119**
Nearest + rasterized opacity	29.65	0.882	0.120
Top-3 composition, no calibration	28.38	0.860	0.149
Composition w/o slope (no refit)	29.05	0.867	0.137
Ours (uncompressed)	29.38	0.875	0.127
Target-RGB oracle (diagnostic)	33.23	0.935	0.067

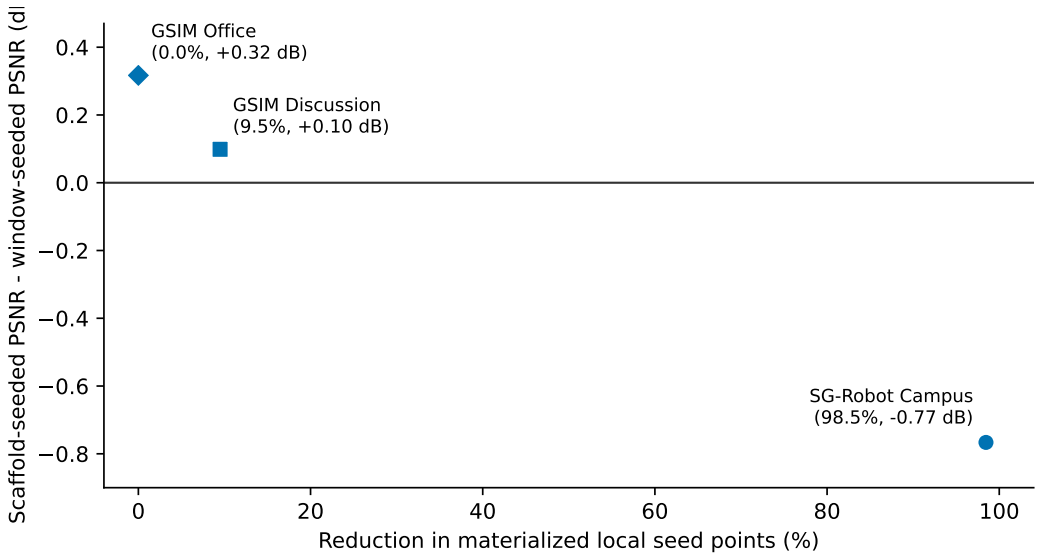


Figure S4: **Quality–resource trade-off in scaffold-seeded initialization.** Seed reduction and PSNR change are measured relative to window seeding at equal final Gaussian counts.



Figure S5: **Window-seeded versus scaffold-seeded initialization.** Matched facade crops on SG-Robot Campus compare reconstruction and absolute error at the same final count of 3.77M Gaussians.

Table S14: Frozen-map composition. Each scene reports the 95th percentile of adjacent-render LPIPS/L1 over interpolated boundary views. These are not target-image fidelity errors.

Scene	Hard assignment		Pose blend		**Ours**	
	LPIPS ↓	L1 ↓	LPIPS ↓	L1 ↓	LPIPS ↓	L1 ↓
Art Gallery	0.084	0.044	**0.042**	**0.040**	**0.042**	**0.040**
Botanical Garden	0.245	0.106	**0.127**	**0.076**	**0.127**	0.078
Business Center	0.134	0.058	**0.045**	**0.035**	**0.045**	**0.035**
Supermarket 1	0.191	0.078	0.075	**0.064**	**0.074**	**0.064**
Supermarket 2	0.084	0.040	0.040	**0.030**	**0.039**	**0.030**
GSIM Discussion	0.637	0.253	**0.611**	**0.208**	0.612	0.209
Living Room	0.071	**0.039**	**0.070**	**0.039**	**0.070**	**0.039**
Lounge	0.281	0.121	**0.269**	**0.111**	**0.269**	0.112
Market	0.144	0.069	**0.104**	**0.065**	**0.104**	**0.065**
GSIM Office	0.081	**0.024**	**0.077**	**0.024**	**0.077**	**0.024**
Washroom	0.064	**0.029**	**0.059**	**0.029**	**0.059**	**0.029**
SG-Robot Campus	0.332	0.072	**0.028**	**0.018**	0.029	**0.018**
SG-Robot Facade	0.333	0.100	**0.199**	**0.079**	0.200	**0.079**
Wild Crowd	0.198	0.049	**0.012**	**0.009**	**0.012**	**0.009**
Static 1	0.021	0.012	**0.012**	**0.009**	**0.012**	**0.009**
ANYmal 1	0.078	0.018	**0.008**	**0.006**	**0.008**	**0.006**
Wild Static 2	0.027	0.011	**0.010**	**0.007**	**0.010**	**0.007**
Avg.	0.177	0.066	0.105	0.050	0.105	0.050

Table S15: Observed-view fidelity with frozen maps and composition parameters. Unlike the adjacent-render statistics, these metrics compare renders to observed RGB. Best displayed values per scene and metric are bold.

Scene	Hard assignment			Pose blend			**Ours**		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Art Gallery	**29.58**	**0.919**	**0.079**	29.10	0.910	0.089	29.11	0.909	0.089
Botanical Garden	**21.82**	**0.736**	**0.165**	21.70	0.724	0.178	21.73	0.723	0.177
Business Center	25.98	**0.869**	**0.100**	25.96	0.865	0.106	**25.99**	0.865	0.106
Supermarket 1	26.48	**0.876**	**0.102**	26.40	0.869	0.110	**26.54**	0.869	0.109
Supermarket 2	27.79	0.909	0.082	28.20	0.911	0.079	**28.32**	**0.912**	**0.076**
GSIM Discussion	**29.76**	**0.920**	**0.088**	27.53	0.878	0.129	27.66	0.878	0.129
Living Room	**37.60**	**0.977**	**0.015**	36.25	0.971	0.019	36.54	0.971	0.019
Lounge	**27.38**	**0.893**	**0.113**	26.56	0.875	0.150	26.57	0.875	0.150
Market	**27.43**	**0.820**	**0.130**	27.30	0.805	0.147	27.40	0.805	0.147
GSIM Office	**39.37**	**0.986**	**0.013**	38.46	0.983	0.017	38.66	0.983	0.017
Washroom	36.56	**0.958**	0.032	36.56	**0.958**	**0.029**	**36.80**	**0.958**	**0.029**
SG-Robot Campus	21.11	**0.635**	**0.490**	**21.25**	**0.635**	0.491	21.22	0.634	0.492
SG-Robot Facade	**23.15**	**0.709**	**0.282**	22.89	0.697	0.293	22.92	0.696	0.294
Wild Crowd	24.09	**0.900**	**0.150**	**24.15**	0.898	0.151	**24.15**	0.898	0.151
Static 1	**37.78**	**0.977**	**0.041**	37.30	0.976	0.043	37.44	0.976	0.043
ANYmal 1	**30.52**	**0.951**	**0.082**	30.44	0.950	0.084	30.44	0.950	0.084
Wild Static 2	**38.40**	**0.978**	**0.054**	37.85	0.977	0.057	38.02	0.977	0.057
Avg.	**29.70**	**0.883**	**0.119**	29.29	0.875	0.128	29.38	0.875	0.127

The observed-view PSNR, SSIM, and LPIPS reveal that hard assignment can preserve sharper individual views while introducing discontinuities between them, exposing a fidelity–continuity trade-off (Table S15). Ray-wise coverage improves mean PSNR over pose blend, with scene- and metric-dependent effects.

Coordinate precision and result scope. Checkpoint inspection found finite source coordinates that overflowed when cast to FP16. We retain FP32 only for affected coordinate tensors and leave all other tensor values and dtypes unchanged. Stage B applies this policy to both seed schemes, repacks seven affected scene/configuration pairs, and repeats calibration and evaluation without retraining. For the fixed Stage A/C maps, six tensors in five scenes are repaired in the raw atlas and both compressed encodings. All eight query variants and both grouping variants are reevaluated with their existing learned states. The other twelve scenes reuse unchanged results.

The final uncompressed atlas totals 13,948,905,118 bytes across 17 scenes. Percentile-clipped compression totals 8,088,460,664 bytes, yielding a 39.7591% mean per-scene reduction. Pooling bytes before division instead gives 42.0137%, a different aggregation that is not used in the main table. The final mean PSNR is 29.3882 dB uncompressed and 29.3300 dB compressed. The main paper's Table 1 uses the latter results. The local/global, initialization, composition, rank/residual, and grouping controls use their respective matched serialization throughout each comparison.

8.5 Factorization Details

The rank/residual sweep varies only the representation of non-DC SH coefficients, with Gaussian geometry fixed (Table S16). A single shared basis and pose-only families achieve nearly identical storage and fidelity to pose + query families at the default setting (Table S19). A separate five-scene diagnostic examines the quality–storage trade-off and visible compression effects (Table S18 and Figures S6 and S7). Its averages describe that subset.

Increasing rank or retaining more residual rows costs storage and generally reduces distortion, with the operating point determined by the storage budget and scene (Table S17). At the default budget, pose + query families provide no measured fidelity advantage over the grouping controls.

Table S16: **Rank and residual sensitivity.** Storage reduction and metric differences are relative to the uncompressed atlas (PSNR in dB).

Configuration	Size (MiB) ↓	Reduction (%) ↑	ΔPSNR ↑	ΔSSIM ↑	ΔLPIPS ↓
Uncompressed SH	782.1	0.00	**0.000**	**+0.00000**	**+0.00000**
Residual sweep (rank 16)					
Res. 0%	444.4	40.87	-0.162	-0.00075	+0.00200
Res. 2%	448.9	40.32	-0.069	-0.00050	+0.00123
Res. 4% (default)	453.3	39.78	-0.058	-0.00044	+0.00105
Res. 8%	462.3	38.70	-0.045	-0.00036	+0.00082
Rank sweep (residual 4%)					
Rank 4	**398.6**	**46.41**	-0.322	-0.00298	+0.00380
Rank 8	416.8	44.20	-0.164	-0.00139	+0.00219
Rank 16 (default)	453.3	39.78	-0.058	-0.00044	+0.00105
Rank 32	526.4	30.94	-0.010	-0.00007	+0.00025

Table S17: Rank/residual sensitivity by dataset. Differences are measured against the uncompressed map. Avg. weights scenes equally.

Rank / residual	Dataset	MiB ↓	Reduction (%) ↑	ΔPSNR ↑	ΔSSIM ↑	ΔLPIPS ↓
4 / 4%	SG-Robot	384.2	47.58	-0.113	-0.00476	+0.00805
4 / 4%	DL3DV	348.3	43.86	-0.256	-0.00602	+0.00553
4 / 4%	GSIM	330.2	41.25	-0.112	-0.00082	+0.00103
4 / 4%	Wild-SLAM	571.1	56.76	-0.823	-0.00152	+0.00366
4 / 4%	Avg.	398.6	46.41	-0.322	-0.00298	+0.00380
8 / 4%	SG-Robot	400.9	45.31	-0.045	-0.00198	+0.00460
8 / 4%	DL3DV	361.3	41.77	-0.123	-0.00298	+0.00329
8 / 4%	GSIM	341.2	39.28	-0.053	-0.00032	+0.00046
8 / 4%	Wild-SLAM	607.6	54.05	-0.442	-0.00069	+0.00217
8 / 4%	Avg.	416.8	44.20	-0.164	-0.00139	+0.00219
32 / 4%	SG-Robot	501.4	31.72	-0.001	-0.00008	+0.00038
32 / 4%	DL3DV	439.1	29.24	-0.005	-0.00014	+0.00025
32 / 4%	GSIM	407.6	27.50	-0.002	-0.00002	+0.00003
32 / 4%	Wild-SLAM	826.2	37.84	-0.034	-0.00007	+0.00049
32 / 4%	Avg.	526.4	30.94	-0.010	-0.00007	+0.00025
16 / 2%	SG-Robot	430.3	41.34	-0.018	-0.00064	+0.00251
16 / 2%	DL3DV	384.0	38.11	-0.048	-0.00106	+0.00185
16 / 2%	GSIM	360.6	35.84	-0.017	-0.00011	+0.00020
16 / 2%	Wild-SLAM	671.5	49.31	-0.200	-0.00031	+0.00135
16 / 2%	Avg.	448.9	40.32	-0.069	-0.00050	+0.00123
16 / 8%	SG-Robot	442.6	39.67	-0.011	-0.00044	+0.00168
16 / 8%	DL3DV	393.6	36.57	-0.031	-0.00078	+0.00127
16 / 8%	GSIM	368.8	34.39	-0.013	-0.00009	+0.00013
16 / 8%	Wild-SLAM	698.3	47.33	-0.129	-0.00020	+0.00085
16 / 8%	Avg.	462.3	38.70	-0.045	-0.00036	+0.00082

Table S18: Factorization sensitivity with fixed Gaussian geometry. Storage reduction and ΔPSNR are measured against the uncompressed representation. A five-scene FP16 control gives 589.5 MiB, 31.08% reduction, and -0.116 dB PSNR change.

Rank, 4% residual			Residual, rank 16		
Rank	Reduction (%) ↑	ΔPSNR (dB) ↑	Residual (%)	Reduction (%) ↑	ΔPSNR (dB) ↑
4	48.16	-0.585	0	42.41	-0.332
8	45.87	-0.316	2	41.84	-0.145
16†	41.28	-0.122	4†	41.28	-0.122
32	32.11	-0.024	8	40.16	-0.094

†Delivered parameters.

Table S19: **Family-grouping and sparse-residual ablations.** Storage reduction and PSNR differences are relative to the uncompressed atlas.

(b) Appearance factorization

Variant	Map size (MiB) ↓	Reduction (%) ↑	ΔPSNR (dB) ↑
Uncompressed SH	782.1	0.00	**0.000**
Single shared basis	453.3	39.78	-0.056
Pose-only families	453.3	39.78	-0.057
Pose + query families, no residual	**444.4**	**40.87**	-0.162
Ours	453.3	39.78	-0.058

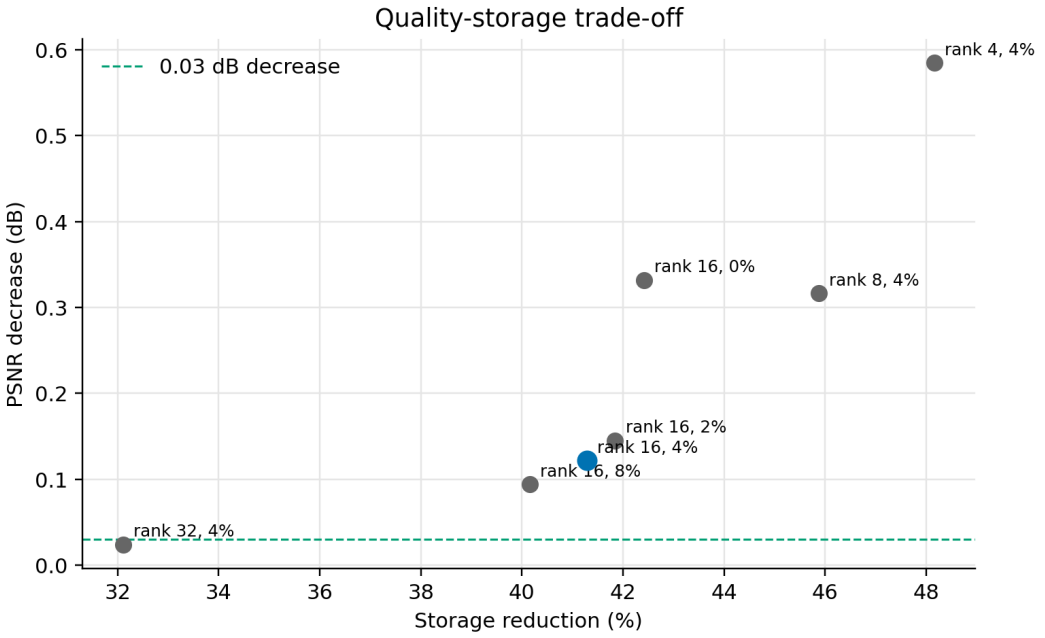


Figure S6: **Appearance-compression trade-off on five sequences.** Rank and residual settings vary storage and reconstruction fidelity. PSNR decrease is measured relative to the uncompressed atlas.

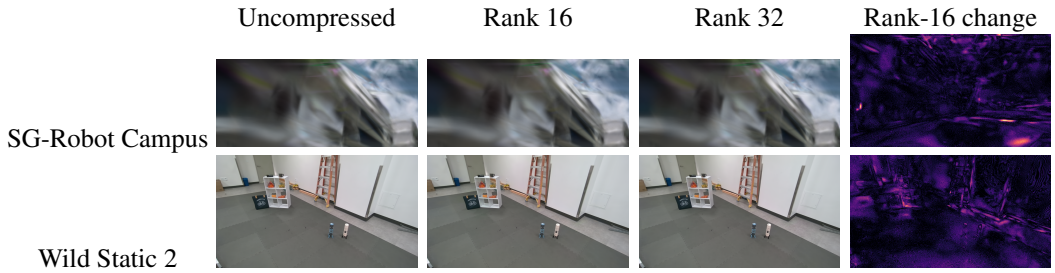


Figure S7: **Rendered appearance at different compression ranks.** Uncompressed, rank-16, and rank-32 renders use matched views. Rank-16 differences from the uncompressed render use a fixed $[0, 0.02]$ scale.

8.6 Rendering Runtime

We measure throughput with different numbers of active local submaps (Table S20), using a fresh process for each scene on an NVIDIA A100-SXM4-80GB with PyTorch 2.5.1 and CUDA 11.8. Eight uniformly spaced session cameras are rendered at their image resolutions. Each configuration has one full warmup pass and five synchronized timing passes. FPS divides the view count by the median elapsed time, followed by equal averaging across scenes. The full decoded atlas remains resident. GPU memory in the main table is peak allocated memory during the $M = 3$ rendering passes, with the peak counter reset after warmup. Loading and decoding time is measured separately, and loading peaks are excluded from the rendering-memory column.

Table S20: Rendering throughput as the number of active local submaps varies. Every row uses the same implementation and explicit GPU synchronization.

Active local submaps	FPS ↑
0 (global only)	**320.5**
1	113.8
2	96.6
3 (default)	84.0
5	66.3

8.7 Window Span

We use a fixed cross-scene span rule, $L = 20$. To characterize this choice, we distribute the same total Gaussian budget across the local submaps over $L \in \{10, 20, 40, 80\}$ on three contrasting sequences. Small L gives each local submap a narrower local viewing distribution but divides the budget among more local submaps. A large L gives each local submap more observations spanning a broader local viewing distribution.

Figure S8 shows that the preferred span depends on the sequence. SG-Robot Campus peaks at $L = 40$, GSIM Discussion favors $L = 10$, and GSIM Office is nearly flat from $L = 10$ to $L = 40$. The fixed $L = 20$ setting provides a cross-scene compromise within these scene-dependent responses.

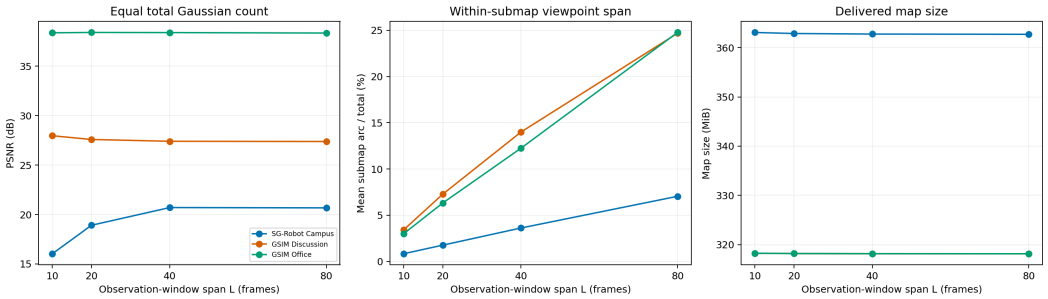


Figure S8: **Effect of observation-window span.** The preferred span varies across scenes under a fixed total Gaussian count across local submaps within each scene.

8.8 Trajectory Interpolation

The main task evaluates observed session cameras, so we separately remove training observations to test nearby trajectory interpolation. On three fixed scenes, withholding every fifth frame yields a 0.48 dB gain for SpecGuard over the global Gaussian map, while withholding one contiguous 10% segment yields 0.13 dB (Table S21). The smaller gain on a contiguous gap indicates sensitivity to observation coverage. Both splits assess nearby trajectory interpolation. Wide-baseline extrapolation remains untested.

Table S21: Rendering at withheld trajectory views. The two panels use different fixed splits. Scores are reported against the withheld RGB images, and means weight the three scenes equally.

Scene	PSNR (dB)			SSIM		LPIPS	
	Global	SpecGuard	Δ	Global	SpecGuard	Global	SpecGuard
Interleaved holdout (20%)							
SG-Robot Campus	21.23	22.45	+1.22	0.619	0.657	0.486	0.259
GSIM Discussion	24.93	24.94	+0.01	0.823	0.819	0.152	0.154
GSIM Office	37.47	37.68	+0.21	0.978	0.979	0.018	0.018
Avg.	27.88	28.36	+0.48	0.807	0.819	0.219	0.143
Contiguous holdout (10%)							
SG-Robot Campus	13.25	13.34	+0.09	0.342	0.342	0.662	0.647
GSIM Discussion	16.20	16.39	+0.19	0.780	0.787	0.335	0.331
GSIM Office	28.31	28.42	+0.11	0.914	0.915	0.090	0.089
Avg.	19.25	19.38	+0.13	0.679	0.681	0.362	0.356

References

Chong Cheng, Sicheng Yu, Zijian Wang, Yifan Zhou, and Hao Wang. Outdoor monocular SLAM with global scale-consistent 3D gaussian pointmaps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Hidenobu Matsuki, Riku Murai, Paul H. J. Kelly, and Andrew J. Davison. Gaussian splatting SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18039–18048, 2024.

Jiarui Meng, Haijie Li, Yanmin Wu, Qiankun Gao, Shuzhou Yang, Jian Zhang, and Siwei Ma. Mirror-3DGS: Incorporating mirror reflections into 3D gaussian splatting. In *IEEE International Conference on Visual Communications and Image Processing*, 2024. URL `https://arxiv.org/abs/2404.01168`.

Erik Sandström, Keisuke Tateno, Michael Oechsle, Michael Niemeyer, Luc Van Gool, Martin R. Oswald, and Federico Tombari. Splat-SLAM: Globally optimized RGB-only SLAM with 3D gaussians. *arXiv preprint arXiv:2405.16544*, 2024.

Ji Shi, Xianghua Ying, Bowei Xing, Ruohao Guo, and Wenzhen Yue. RT-Splatting: Joint reflection-transmission modeling with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Jianhao Zheng, Zihan Zhu, Valentin Bieri, Marc Pollefeys, Songyou Peng, and Iro Armeni. WildGS-SLAM: Monocular gaussian splatting SLAM in dynamic environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Yongyang Zhou, Fang-Lue Zhang, Zichen Wang, and Lei Zhang. RTR-GS: 3D gaussian splatting for inverse rendering with radiance transfer and reflection. In *Proceedings of the ACM International Conference on Multimedia*, 2025.