DUAL-SPLATTING FOR EFFICIENT UHD 3D GAUSSIAN SPLATTING SUPER-RESOLUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

3D Gaussian Splatting (3DGS) enables novel view synthesis with high visual quality, but its rendering quality remains limited by the resolution of captured observations. To support close inspection, recent 3DGS super-resolution methods introduce high-frequency priors from pretrained single image super-resolution (SISR) models. However, fitting generated RGB details can require dense Gaussian primitives and high resolution rasterization, limiting ultra high definition (UHD) scalability on consumer hardware. To address this issue, we propose Dual-Splatting, a framework that integrates the SISR latent generative process into a dual branch representation. The RGB branch reconstructs scene structure and appearance from low resolution observations, while the latent branch models high-frequency details through SISR latent priors. The two branches share geometry to align scene reconstruction and detail synthesis. A latent refinement head transforms features rendered at the input resolution into latents for the frozen SISR decoder, enabling detail synthesis without Gaussian rasterization at the target resolution. To further improve consistency across views, we introduce multiview regularization in latent space that aggregates reprojected predictions from neighboring views. Experiments across multiple datasets and resolutions show compact scene representations and improved consistency over independent SISR, with practical 4K/6K super-resolution on a 24 GB GPU.



Figure 1: Dual-Splatting for UHD 3DGS super-resolution: (a) renderings, (b) resolution scalability, and (c) MUSIQ and memory comparisons on an RTX 4090.

1 INTRODUCTION

3D Gaussian Splatting (3DGS) enables efficient novel view synthesis with high visual quality (Kerbl et al., 2023). However, its rendering quality is strongly coupled to the scale of the input observations. For example, reconstructions from distant drone imagery can preserve scene structure but lack fine

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.

textures under close inspection. Supporting both global browsing and local inspection therefore requires synthesizing details beyond the captured observations, as illustrated in Figure 1.

A natural solution is to apply single image super-resolution (SISR) to rendered views. However, enhancing each view independently may introduce inconsistent details during scene navigation. Recent 3DGS super-resolution (3DGSSR) methods incorporate SISR image priors into a shared Gaussian representation (Feng et al., 2024b). Nevertheless, directly fitting generated RGB images can require smaller and denser primitives to represent fine textures, which increases representation complexity and rasterization cost. This becomes particularly problematic for ultra high definition (UHD) reconstruction under limited GPU memory.

A more efficient solution should separate high-frequency detail synthesis from scene reconstruction, reducing the need to represent both through increasingly dense RGB primitives. Notably, SISR models operating in latent space synthesize fine textures through compact representations before decoding them into RGB images. This observation suggests integrating the latent generative process of SISR into 3D reconstruction, allowing generated details to be modeled compactly while preserving scene structure from captured observations.

Based on this observation, we propose Dual-Splatting, a 3DGS super-resolution framework with an RGB branch and a latent branch. The RGB branch reconstructs scene structure and appearance from low resolution (LR) observations, while the latent branch models high-frequency details through latent priors from pretrained SISR models. The branches share Gaussian geometry to align scene reconstruction and detail synthesis. A latent refinement head transforms rendered features into latents for the frozen SISR decoder, enabling high resolution (HR) synthesis with Gaussian rasterization at the LR input resolution. To further improve multiview consistency, we introduce regularization that aggregates reprojected latent predictions from neighboring viewpoints to constrain inconsistent generated details. Experiments show reduced Gaussian complexity and improved multiview consistency over independent SISR, with practical 4K/6K synthesis on a 24 GB GPU.

Our contributions are summarized as follows:

- We propose an efficient 3DGSSR framework that integrates SISR latent generation into 3D reconstruction, supporting UHD synthesis under limited GPU memory.
- We propose a dual branch representation that separates RGB scene reconstruction from latent detail synthesis while aligning both branches through shared Gaussian geometry.
- We introduce multiview regularization that aggregates neighboring latent predictions to constrain inconsistent generated details across views.

2 Related Work

3D Gaussian Splatting. 3DGS (Kerbl et al., 2023) enables efficient novel view synthesis through explicit Gaussians and differentiable rasterization, motivating applications such as GigaSLAM (Deng et al., 2025), TrackerSplat (Yin et al., 2025), and EVSplitting (Feng et al., 2024a). However, rendering quality remains sensitive to viewing scale. To improve scale compatibility, Mip-Splatting (Yu et al., 2024b) filters Gaussians to reduce aliasing, while Analytic-Splatting (Liang et al., 2024) approximates integration over pixel windows. Upscaling guided by gradients (Niedermayr et al., 2025) also supports higher resolution rendering. Although these methods reduce aliasing artifacts, they do not explicitly model missing details, leading to blurry textures under close inspection.

3DGS super-resolution. To synthesize high-frequency appearance in 3D space, 3DGSSR methods (Xie et al., 2025) introduce latent diffusion priors (Rombach et al., 2022) through SISR models such as StableSR (Wang et al., 2024a). Since independently enhanced views may contain inconsistent details, SRGS (Feng et al., 2024b) constrains downsampled patches, while SplatSure (Asthana et al., 2025) reweights supervision using reconstruction errors. SuperGaussian (Shen et al., 2024) uses video priors to improve consistency, and S2Gaussian (Wan et al., 2025) and IE-SRGS (Feng et al., 2026) introduce geometric regularization. GaussianSR (Yu et al., 2024a), Arbi-3DGSR (Zeng et al., 2026), and 3DSR (Chen et al., 2025) also incorporate diffusion priors during 3D optimization. However, their Gaussian appearance remains optimized for RGB reconstruction, where fitting generated details can require denser primitives and increase representation complexity.

Efficient 3D representations. Efficient 3DGSSR also depends on compact scene representations. Scaffold-GS (Lu et al., 2024) groups Gaussians into anchors with shared features, while Octree-GS (Ren et al., 2024) organizes anchors hierarchically. ContextGS (Wang et al., 2024b) and CompGS (Liu et al., 2024) improve storage efficiency, and Taming 3DGS (Mallick et al., 2024) reduces reconstruction cost. Another approach is to reconstruct scenes in compact latent spaces, as in LRF (Zhou et al., 2025) and L3DG (Roessle et al., 2024) using learned image encodings (Esser et al., 2021). Feature3DGS (Zhou et al., 2024) and LangSplat (Qin et al., 2024) attach learned features to Gaussians, while PartField (Liu et al., 2025) learns features for 3D part understanding. However, latent priors for super-resolution may contain inconsistent details across views. We therefore combine latent modeling with RGB supervision of shared geometry, allowing captured observations to constrain scene structure during detail synthesis.

3 METHOD

Our framework decouples 3D super-resolution into RGB scene reconstruction and latent detail synthesis. Figure 2 shows how the two branches share a Gaussian representation.

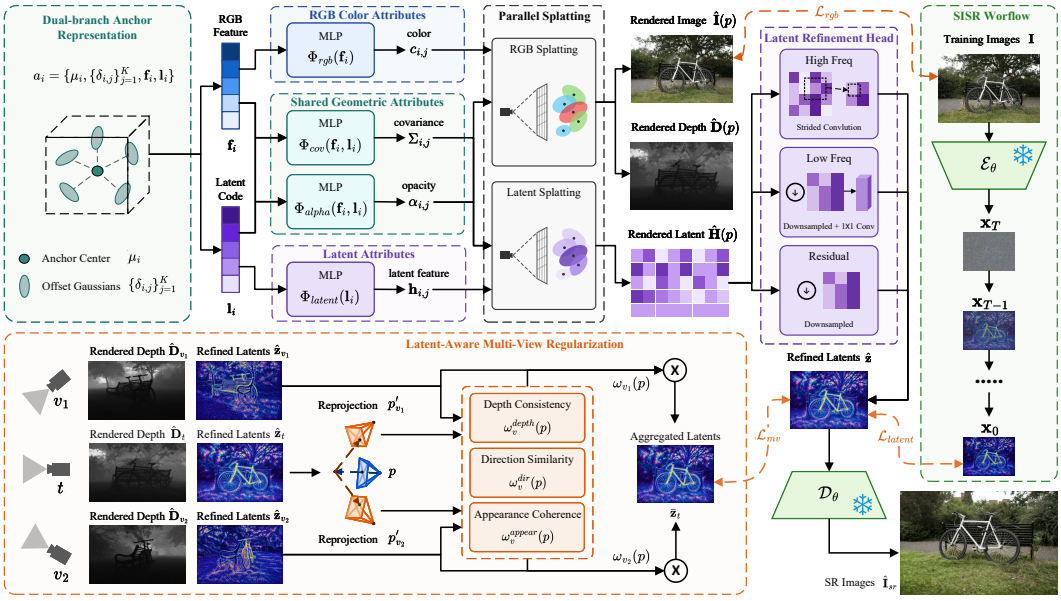

Figure 2: Overview of Dual-Splatting, with an RGB branch for scene reconstruction and a latent branch for high-frequency detail synthesis. The branches share Gaussian geometry.

Following 3DGS (Kerbl et al., 2023), we represent the scene with Gaussian primitives defined by means μ_i, covariances Σ_i, and opacities α_i. Pixels are rendered through alpha blending in depth order (Zwicker et al., 2002). To improve representation efficiency, we extend Scaffold-GS anchors (Lu et al., 2024) with latent codes.

3.1 RGB AND LATENT DUAL-SPLATTING

Given LR observations $\mathbf{I}_{lr} \in \mathbb{R}^{3\times H\times W}$, we represent the scene with anchors

$$\mathcal{A} = \{a_i\}_{i=1}^{N}, \qquad a_i = \{\mu_i, \{\delta_{i,j}\}_{j=1}^{K}, \mathbf{f}_i, \mathbf{l}_i\}, \tag{1}$$

where μ_i is the anchor center, $\delta_{i,j}$ denotes offset j, $\mathbf{f}_i \in \mathbb{R}^{32}$ is the RGB feature, and $\mathbf{l}_i \in \mathbb{R}^{M}$ is the latent code. We use separate RGB features and latent codes to accommodate the different supervision from captured images and generated SISR priors. The RGB features support reconstruction of observed appearance, while the latent codes model generated details. Both features contribute to shared geometry, allowing captured observations to constrain the underlying primitives as the latent branch fits SISR targets.

Branch attributes and shared geometry. Given viewing direction d_i from the camera to anchor i, the MLPs Φ_{rgb} and Φ_{latent} decode the anchor features into K colors $c_{i,j} \in [0,1]^3$ and latent features $\mathbf{h}_{i,j} \in \mathbb{R}^C$, respectively:

$$\{c_{i,j}\}_{j=1}^{K} = \mathrm{sigmoid}(\Phi_{rgb}(\mathbf{f}_i, d_i)), \quad \{\mathbf{h}_{i,j}\}_{j=1}^{K} = \Phi_{latent}(\mathbf{l}_i, d_i), \tag{2}$$

To keep synthesized details aligned with scene structure, both branches share Gaussian positions, covariances, and opacities:

$$\mu_{i,j} = \mu_i + \delta_{i,j}, \quad \Sigma_{i,j} = \Phi_{cov}(\mathbf{f}_i, \mathbf{l}_i), \quad \alpha_{i,j} = \Phi_{alpha}(\mathbf{f}_i, \mathbf{l}_i), \tag{3}$$

where Φ_{cov} and Φ_{alpha} are MLPs that predict covariance and opacity from both features. This connects RGB reconstruction and latent synthesis through the shared Gaussian geometry.

Parallel rasterization. The two branches are rasterized in parallel with shared blending weights. We sort the NK offset Gaussians by depth and denote their colors and features by c_n and $\mathbf{h}_n$:

$$\begin{bmatrix} \hat{\mathbf{I}}(p) \\ \hat{\mathbf{H}}(p) \end{bmatrix} = \sum_{n=1}^{NK} \begin{bmatrix} c_n \\ \mathbf{h}_n \end{bmatrix} \sigma_n(p) \prod_{m<n} (1 - \sigma_m(p)). \tag{4}$$

Here $\hat{\mathbf{I}}$ and $\hat{\mathbf{H}}$ denote the rendered RGB image and latent feature map, and $\sigma_n(p)$ is the effective opacity of projected Gaussian n at pixel p. To avoid HR Gaussian rasterization, both outputs are computed at the LR input resolution. Concatenating their $3 + C$ channels allows a single rasterization, after which the latent features are refined and decoded into HR images.

Latent refinement. The rendered features $\hat{\mathbf{H}} \in \mathbb{R}^{C \times H \times W}$ must be transformed into latents compatible with the pretrained SISR decoder. We therefore introduce a refinement head Φ_{refine} with three stems. The high-frequency stem extracts local details $\mathbf{z}_{hf}$ through strided convolutions, while the low-frequency stem models global structure $\mathbf{z}_{lf}$ using a 1×1 convolution on pooled features. The residual stem downsamples $\hat{\mathbf{H}}$ for the residual connection $\mathbf{z}_{res}$. Their outputs are fused as

$$\hat{\mathbf{z}} = \Phi_{refine}(\hat{\mathbf{H}}) = \tfrac{1}{2}\left[\tanh(\tau(\mathbf{z}_{hf} + \mathbf{z}_{lf}) + \mathbf{z}_{res}) + 1\right], \tag{5}$$

where τ is a learnable temperature that scales the combined frequency contributions. The latents have four channels and are normalized to $[0,1]$ during optimization. With downsampling factor s from the rendered feature grid to the decoder's latent grid, we obtain $\hat{\mathbf{z}} \in [0,1]^{4 \times H/s \times W/s}$. We then denormalize these latents and decode them with the frozen SISR decoder as $\hat{\mathbf{I}}_{sr} = \mathcal{D}_\theta(\hat{\mathbf{z}}) \in \mathbb{R}^{3 \times rH \times rW}$, where r is the supported super-resolution factor.

3.2 Multiview Regularization in Latent Space

Since SISR priors are generated independently for each view, their latent details may disagree across views. We introduce multiview regularization that uses the shared geometry to aggregate neighboring predictions. Given target view t, we select neighbors $V(t)$ by camera position and viewing direction, then reproject pixel p using the rendered depth:

$$p' = \pi_v\Big(\mathbf{T}_{v \leftarrow t} \pi_t^{-1}(p, \hat{\mathbf{D}}_t(p))\Big), \tag{6}$$

where π_v projects into view v and $\mathbf{T}_{v \leftarrow t}$ transforms points from view t to v. We use depth maps and camera intrinsics at the latent resolution and sample valid neighboring predictions at p', denoted by $\hat{\mathbf{z}}_{t \leftarrow v}(p)$.

To reduce the influence of inconsistent correspondences, we assign each neighbor a weight $\omega_v(p) = \omega_v^{depth}(p)\omega_v^{dir}(p)\omega_v^{appear}(p)$. These terms measure agreement in depth, viewing direction, and latent appearance, as defined in Supplementary Section A.1. Aggregating the neighboring predictions gives the latent target

$$\bar{\mathbf{z}}_t(p) = \frac{\sum_{v \in V(t)} \omega_v(p) \hat{\mathbf{z}}_{t \leftarrow v}(p)}{\sum_{v \in V(t)} \omega_v(p) + \epsilon}. \tag{7}$$

The multiview regularization is defined as the Charbonnier loss between the predicted and aggregated latents over valid pixels $\mathcal{S}$:

$$\mathcal{L}_{mv} = \frac{1}{|\mathcal{S}|} \sum_{p \in \mathcal{S}} \sqrt{\|\hat{\mathbf{z}}_t(p) - \bar{\mathbf{z}}_t(p)\|_2^2 + \epsilon^2}. \tag{8}$$

3.3 Training Objective

The training objective combines RGB reconstruction, latent reconstruction, and multiview regularization:

$$\mathcal{L} = \lambda_{rgb}\mathcal{L}_{rgb} + \lambda_{latent}\mathcal{L}_{latent} + \lambda_{mv}\mathcal{L}_{mv}. \tag{9}$$

The coefficients control the relative contribution of each loss. RGB reconstruction combines L1 and SSIM losses against captured images, while latent reconstruction additionally uses a perceptual loss against normalized denoised SISR targets $\mathbf{z}_t^\star$. These losses, detailed in Supplementary Section A.2, jointly optimize both branches and the refinement head.

4 Experiments

4.1 Experimental Settings

Datasets and resolutions. We evaluate SR quality and scalability from 1K to 6K. T&T/DB combines Tanks & Temples (Knapitsch et al., 2017) and Deep Blending (Hedman et al., 2018). The other benchmark groups are Mip-NeRF 360 (Barron et al., 2022) and Mill 19 (Turki et al., 2022)/UrbanScene3D (Lin et al., 2022). Supplementary Section A.3 lists the scenes. For T&T/DB, 1K inputs and InvSR's $4\times$ scale limit evaluation to 4K.

Baselines and implementation. We use the baselines' officially recommended settings, including bicubic upsampling as 3DGS-bicubic, direct HR rendering as 3DGS-hr, and independent SISR as 3DGS-InvSR. While 3DGS and Mip-Splatting use only LR observations, 3DGSSR methods also use InvSR priors (Yue et al., 2025) at $4\times$ scale. All experiments run on a 24 GB RTX 4090. Our method uses Scaffold-GS anchors, denoised InvSR latent targets, and a frozen decoder, with $M = 64$, $C = 8$, $\lambda_{rgb} = \lambda_{latent} = 1$, and $\lambda_{mv} = 0.1$.

Metrics. We measure scene fidelity with PSNR in dB and SSIM at the captured scale. Without HR references, we evaluate SR quality using NIQE, MUSIQ, MANIQA, and TRES. Multiview consistency is measured by cross view reprojection error, denoted as CVRE, using L1, PSNR, and SSIM between reprojected renderings. Efficiency is measured by Gaussian count in millions and peak training memory in GB. Lower NIQE, CVRE L1, memory, and count indicate better performance. Other metrics favor higher values. OOM denotes training memory failures.

4.2 Main Results

Tables 1 to 3 summarize efficiency, SR quality, LR fidelity, and multiview consistency. For our method, we select each metric's best value independently across evaluated configurations. Figure 1 uses the same selection for quality and memory.

Representation efficiency and UHD scalability. Our method achieves the lowest reported Gaussian count across all evaluated datasets and resolutions. At 2K on Mip-NeRF 360, it uses 0.85M primitives, compared with 4.26M for SRGS and 1.24M for SplatSure, reducing the count by 80.0% and 31.5%, respectively. At 4K and 6K, all four evaluated 3DGSSR baselines encounter OOM on the 24 GB GPU, while our method continues to synthesize SR views. This demonstrates the feasibility of UHD synthesis with a compact Gaussian representation. Fewer primitives do not always imply lower memory use. At 4K on Mip-NeRF 360, our method requires 12.65 GB, compared with 10.20 GB for 3DGS trained on LR observations.

High-frequency recovery. In addition to representation efficiency, our method achieves the highest reported MUSIQ and MANIQA at 1K and 2K across the three benchmark groups in Table 1. At 2K on T&T/DB, it reaches 74.29 MUSIQ and 0.478 MANIQA, compared with 69.31/0.302 for SRGS and 60.41/0.313 for SplatSure. The qualitative comparisons in Figure 3 further show the improvement in local details. In the upper UHD rows, our method produces clearer truck markings and object contours than bicubic upsampling and direct HR rendering. The lower HR rows show clearer chair slats and foliage. GaussianSR still obtains lower NIQE at 1K on T&T/DB and Mip-NeRF 360. These comparisons support improved perceptual appearance, although the accuracy of synthesized details cannot be established without HR references.

Table 1: Quantitative comparison of 3DGSSR methods at 1K and 2K. PSNR and SSIM measure fidelity at the captured scale. Best values are shown in bold.

		LR fidelity		SR quality			Efficiency	
Dataset	Method	PSNR	SSIM	NIQE	MUSIQ	MANIQA	GB	M pts
1K target resolution								
T&T/DB	3DGS-bicubic	**24.62**	**0.821**	7.35	30.32	0.226	**4.28**	1.06
	3DGS-hr			4.18	51.67	0.249		
	Mip-Splatting	24.41	0.817	4.82	45.68	0.231	6.74	1.76
	GaussianSR	21.35	0.707	**3.17**	54.51	0.270	22.94	0.76
	SRGS	21.18	0.670	3.60	63.08	0.294	6.80	2.63
	SplatSure	22.65	0.762	4.18	57.74	0.285	6.17	0.65
	Ours	23.77	0.796	3.74	**70.48**	**0.513**	5.74	**0.58**
Mip-NeRF 360	3DGS-bicubic	26.52	0.818	7.54	31.51	0.247	5.75	1.40
	3DGS-hr			3.26	55.26	0.29		
	Mip-Splatting	**26.65**	**0.822**	5.42	42.64	0.219	9.29	2.08
	GaussianSR	19.97	0.633	**2.87**	53.21	0.288	19.10	1.08
	SRGS	20.95	0.609	3.01	63.95	0.306	11.10	3.45
	SplatSure	21.13	0.674	3.33	63.40	0.313	5.36	0.99
	Arbi-3DGSR	24.47	0.716	4.11	51.49	0.228	22.91	1.28
	Ours	24.71	0.746	3.03	**71.75**	**0.529**	**5.27**	**0.65**
Mill 19 / UrbanScene 3D	3DGS-bicubic	**21.36**	**0.720**	7.52	32.03	0.228	16.01	3.95
	3DGS-hr			3.92	58.17	0.323		
	Mip-Splatting	20.73	0.708	4.72	46.06	0.215	10.98	3.76
	SRGS	17.22	0.477	2.80	60.47	0.253	14.99	6.53
	Ours	20.29	0.706	**2.78**	**68.39**	**0.479**	**5.47**	**0.75**
OOM: T&T/DB: Arbi-3DGSR. Mill 19/UrbanScene3D: GaussianSR/SplatSure/Arbi-3DGSR.								
2K target resolution								
T&T/DB	3DGS-bicubic	**26.02**	**0.857**	7.64	41.37	0.261	**6.52**	1.67
	3DGS-hr			4.11	57.81	0.263		
	Mip-Splatting	25.74	0.855	5.50	53.42	0.250	9.11	2.46
	SRGS	22.53	0.738	4.11	69.31	0.302	8.69	3.28
	SplatSure	24.23	0.812	4.12	60.41	0.313	17.33	0.98
	Ours	25.14	0.844	**3.75**	**74.29**	**0.478**	8.20	**0.78**
Mip-NeRF 360	3DGS-bicubic	27.81	0.837	7.79	43.28	0.254	8.11	1.88
	3DGS-hr			3.83	54.20	0.264		
	Mip-Splatting	**27.82**	**0.841**	6.15	50.82	0.224	12.17	2.81
	SRGS	21.75	0.662	3.91	68.86	0.292	13.36	4.26
	SplatSure	22.81	0.730	3.70	66.10	0.287	17.39	1.24
	Ours	25.72	0.783	**3.26**	**73.28**	**0.432**	**7.95**	**0.85**
Mill 19 / UrbanScene 3D	3DGS-bicubic	**23.75**	**0.806**	7.58	45.17	0.235	15.46	4.60
	3DGS-hr			4.15	58.48	0.291		
	Mip-Splatting	22.83	0.791	5.26	54.21	0.210	14.93	4.65
	Ours	22.18	**0.806**	**2.78**	**71.83**	**0.432**	**6.68**	**0.95**
OOM: all groups: GaussianSR/Arbi-3DGSR. Mill 19/UrbanScene3D: SRGS/SplatSure.								

Global fidelity. Our method improves fidelity at the captured scale over the evaluated 3DGSSR baselines. At 2K on T&T/DB, it achieves 25.14 dB PSNR and 0.844 SSIM. The corresponding values are 22.53 dB and 0.738 for SRGS, and 24.23 dB and 0.812 for SplatSure. However, 3DGS trained only on LR observations achieves higher fidelity at 26.02 dB and 0.857. The improvement over the 3DGSSR baselines is consistent with our dual branch design, where RGB observations constrain scene appearance while latent features model generated details.

Table 2: Scalability to 4K and 6K. Dashes denote unreported LR fidelity at 6K.

Dataset	Method	LR fidelity PSNR	LR fidelity SSIM	SR quality NIQE	SR quality MUSIQ	SR quality MANIQA	Efficiency GB	Efficiency M pts
4K target resolution								
T&T/DB	3DGS-bicubic	**26.45**	**0.866**	8.86	53.17	0.282	9.04	2.02
	3DGS-hr			6.18	58.48	0.278		
	Mip-Splatting	26.05	0.863	7.30	57.03	0.267	**9.01**	2.88
	Ours	25.55	0.850	**4.61**	**71.01**	**0.404**	10.44	**0.88**
Mip-NeRF 360	3DGS-bicubic	27.30	0.805	8.13	60.56	0.256	**10.20**	2.55
	3DGS-hr			5.24	62.23	0.254		
	Mip-Splatting	**27.34**	**0.814**	6.75	64.22	0.232	17.87	3.96
	Ours	25.28	0.764	**3.82**	**73.19**	**0.332**	12.65	**1.11**
Mill 19 / UrbanScene 3D	3DGS-bicubic	**25.17**	**0.836**	8.39	56.57	0.225	14.45	5.42
	3DGS-hr			6.15	58.62	0.238		
	Mip-Splatting	23.66	0.812	6.68	58.24	0.212	17.77	5.54
	Ours	23.00	0.827	**3.68**	**68.82**	**0.374**	**11.39**	**1.12**
6K target resolution								
Mip-NeRF 360	3DGS-bicubic	–	–	8.40	64.48	0.259	**10.86**	2.68
	3DGS-hr	–	–	5.25	66.36	0.248		
	Mip-Splatting	–	–	7.07	66.90	0.236	19.78	4.30
	Ours	–	–	**4.22**	**71.63**	**0.300**	15.64	**1.16**
Mill 19 / UrbanScene 3D	3DGS-bicubic	–	–	8.31	61.33	0.230	20.74	5.87
	3DGS-hr	–	–	6.29	60.77	0.264		
	Mip-Splatting	–	–	6.38	61.13	0.225	22.07	6.56
	Ours	–	–	**4.03**	**66.54**	**0.334**	**15.53**	**1.23**

OOM in every setting above: GaussianSR, SRGS, SplatSure, Arbi-3DGSR

Multiview consistency. Table 3 compares multiview consistency on T&T/DB. Our method improves CVRE over independent InvSR enhancement at 1K, 2K, and 4K. At 4K, reprojection L1 decreases from 0.0546 to 0.0427 and PSNR increases from 21.01 to 23.25 dB. SplatSure achieves better CVRE at 1K and 2K but encounters OOM at 4K, while 3DGS-InvSR obtains higher MUSIQ at 2K and 4K. Thus, independent SISR provides strong perceptual quality, while our method improves consistency between reprojected views and supports UHD synthesis.

Table 3: Quantitative comparison of multiview consistency on T&T/DB, measured by CVRE.

Scale	Method	SR quality without references MUSIQ↑	SR quality without references MANIQA↑	Reprojection CVRE L1↓	Reprojection CVRE PSNR↑	Reprojection CVRE SSIM↑
1K	3DGS-InvSR	69.79	0.375	0.0584	20.64	0.641
	GaussianSR	54.51	0.270	0.0567	21.57	0.664
	SRGS	63.08	0.294	0.0544	21.32	0.692
	SplatSure	57.74	0.285	**0.0461**	**23.10**	**0.744**
	Ours	**70.48**	**0.513**	0.0510	21.68	0.667
2K	3DGS-InvSR	**75.04**	0.393	0.0560	20.80	0.682
	SRGS	69.31	0.302	0.0548	21.06	0.725
	SplatSure	60.41	0.313	**0.0452**	**23.41**	**0.749**
	Ours	**74.29**	**0.478**	0.0469	22.32	0.702
4K	3DGS-InvSR	**75.00**	0.383	0.0546	21.01	0.723
	Ours	**71.01**	**0.404**	**0.0427**	**23.25**	**0.761**

OOM: Arbi-3DGSR at all scales. GaussianSR at $\geq$2K. SRGS/SplatSure at 4K.

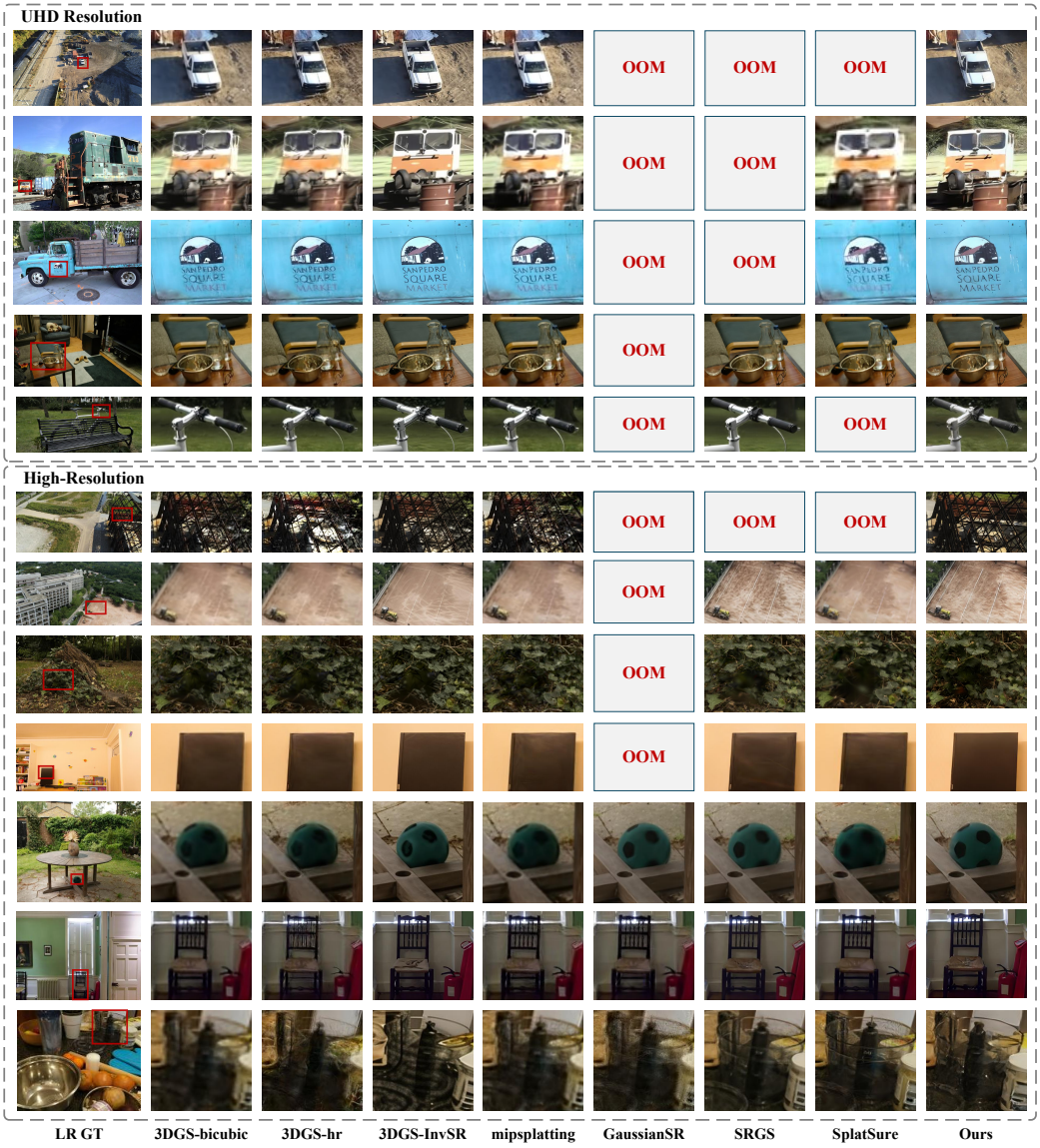

Figure 3: Qualitative comparison of Dual-Splatting and the evaluated baselines across scenes and resolutions. The upper and lower groups show UHD and HR renderings, respectively.

4.3 Ablation Study

RGB branch. Table 4 compares reconstruction with and without RGB supervision. Removing it retains similar MUSIQ but increases CVRE L1 from 0.0469 to 0.0681 and lowers reprojection PSNR from 22.32 to 19.28 dB. Supplementary Figure 4 shows more holes and floaters without RGB supervision, motivating its use to constrain scene structure. The full model uses independently selected metrics, so this comparison does not isolate the RGB branch under matched settings.

Table 4: Ablation of the RGB branch on T&T/DB. Best values are shown in bold.

	SR quality without references				Reprojection CVRE		
Configuration	NIQE	MUSIQ	MANIQA	TRES	L1	PSNR	SSIM
Without RGB	**3.70**	74.18	0.470	86.01	0.0681	19.28	0.621
With RGB	3.75	**74.29**	**0.478**	**86.60**	**0.0469**	**22.32**	**0.702**

Latent refinement head. We ablate the two frequency stems at 4K while retaining the residual stem, as shown in Table 5. Compared with the residual stem alone, adding both stems improves MUSIQ from 58.42 to 71.01 and MANIQA from 0.322 to 0.404. The full head achieves the lowest NIQE, while H+R gives the highest MUSIQ. LR PSNR varies from 25.49 to 25.58 dB and SSIM from 0.848 to 0.850. These observations suggest that the frequency stems improve synthesized appearance over the residual stem alone, with little change in reconstruction fidelity at the captured scale.

Table 5: Ablation of the latent refinement head at 4K on T&T/DB. H, L, and R denote the high-frequency, low-frequency, and residual stems.

	LR fidelity		SR quality without references			
Stems	PSNR↑	SSIM↑	NIQE↓	MUSIQ↑	MANIQA↑	TRES↑
H+L+R	25.55	**0.850**	**4.61**	71.01	**0.404**	**76.97**
H+R	25.49	0.850	4.64	**71.52**	0.403	76.84
L+R	**25.58**	0.850	4.96	70.61	0.395	76.57
R	25.50	0.848	6.78	58.42	0.322	62.04

Multiview regularization. In a separate sweep from the RGB branch comparison, we vary λ_{mv} to examine its effect on SR quality and multiview consistency in Table 6. Increasing λ_{mv} from 0 to 0.5 reduces reprojection L1 from 0.050 to 0.044. However, increasing it from 0.1 to 0.5 worsens NIQE and MUSIQ, despite improving MANIQA and TRES. Stronger consistency constraints therefore introduce a tradeoff between the perceptual metrics. We set $\lambda_{mv} = 0.1$ to improve consistency across views while limiting the loss in NIQE and MUSIQ.

Table 6: Ablation of the multiview regularization coefficient λ_{mv} on T&T/DB. We compare its effect on perceptual quality and reprojection consistency.

	SR quality without references				Reprojection CVRE		
λ_{mv}	NIQE	MUSIQ	MANIQA	TRES	L1	PSNR	SSIM
0	3.74	**74.38**	0.468	86.24	0.050	21.87	0.692
0.01	**3.69**	74.36	0.469	86.18	0.049	21.95	0.693
0.05	3.75	74.30	0.469	86.19	0.049	21.99	0.694
0.10	3.81	74.21	0.475	86.53	0.049	22.05	0.696
0.50	4.17	73.62	**0.486**	**86.65**	**0.044**	**22.82**	**0.721**

4.4 Limitations

Our experiments focus on InvSR and discrete viewpoints. Other SISR models and continuous camera trajectories are left for future evaluation. Without HR references, synthesized textures cannot be verified against actual scene details.

5 Conclusion

In this paper, we present Dual-Splatting for faithful scene reconstruction and plausible high-frequency synthesis under limited GPU resources. Different from 3DGS super-resolution methods that fit generated details in RGB space, our method embeds pretrained SISR latents into a shared 3D representation through an RGB and latent dual branch design. The RGB branch preserves scene structure and appearance from LR observations, while the latent branch compactly models generated details. The latent refinement head enables HR decoding from features rasterized at the LR resolution, and multiview regularization improves consistency across views. Experiments show improved LR fidelity and reduced Gaussian complexity compared with the evaluated 3DGSSR baselines, and better multiview consistency than independent SISR. Our method supports 4K and 6K synthesis on a single 24 GB GPU, providing a practical solution for UHD view synthesis. We hope this work inspires further research on efficient 3DGS super-resolution.

AI USE STATEMENT

We used generative AI tools to polish the language and check manuscript consistency and bibliographic information. Their use did not extend to method development, experimental design or execution, research data generation, or mathematical derivation. We take responsibility for the final text, scientific claims, and reported results.

ETHICS STATEMENT

Generative super-resolution produces plausible details beyond the captured observations, whose accuracy cannot be established from visual quality alone. We therefore distinguish perceptual quality from reconstruction fidelity in our evaluation and limitations. When these renderings are used for inspection or documentation, generated textures should not be interpreted as evidence of unobserved physical details.

REPRODUCIBILITY STATEMENT

We specify the representation and training objectives in Section 3, and report datasets, baselines, SISR priors, hardware, and hyperparameters in Section 4.1. Supplementary Sections A.1 and A.2 define the reprojection weights and reconstruction losses. Tables 1 to 6 evaluate reconstruction quality, efficiency, multiview consistency, and individual components. Since reported metrics are selected across configurations, reproducing these results requires mapping each value to its configuration, together with complete scene splits and preprocessing details. Comparing quality against computational cost further requires one fixed configuration, controls with the same anchor backbone, and runtime measurements for the complete pipeline.

REFERENCES

Pranav Asthana, Alex Hanson, Allen Tu, Tom Goldstein, Matthias Zwicker, and Amitabh Varshney. Splatsure: Selective super-resolution for multi-view consistent 3d gaussian splatting. *arXiv preprint arXiv:2512.02172*, 2025.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5470–5479, 2022.

Yi-Ting Chen, Ting-Hsuan Liao, Pengsheng Guo, Alexander Schwing, and Jia-Bin Huang. Bridging diffusion models and 3d representations: A 3d consistent super-resolution framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13481–13490, 2025.

Kai Deng, Yigong Zhang, Jian Yang, and Jin Xie. Gigaslam: Large-scale monocular slam with hierarchical gaussian splats. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–10, 2025.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Qi-Yuan Feng, Geng-Chen Cao, Hao-Xiang Chen, Qun-Ce Xu, Tai-Jiang Mu, Ralph Martin, and Shi-Min Hu. Evsplitting: An efficient and visually consistent splitting algorithm for 3d gaussian splatting. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024a.

Xiang Feng, Yongbo He, Linxi Chen, Yan Yang, Chengkai Wang, Yifei Chen, Yixuan Zhong, Zhenzhong Kuang, Xufei Yin, Yanming Zhu, et al. Srgs: Super-resolution 3d gaussian splatting. *arXiv preprint arXiv:2404.10318*, 2024b.

Xiang Feng, Tieshi Zhong, Shuo Chang, Weiliu Wang, Chengkai Wang, Yifei Chen, Tongyu Hu, Yuhe Wang, Zhenzhong Kuang, Xuefei Yin, et al. Ie-srgs: An internal-external knowledge fusion framework for high-fidelity 3d gaussian splatting super-resolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 3957–3965, 2026.

Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. *ACM Transactions on Graphics*, 37(6):257:1–257:15, 2018.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4), 2023.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

Zhihao Liang, Qi Zhang, Wenbo Hu, Lei Zhu, Ying Feng, and Kui Jia. Analytic-splatting: Anti-aliased 3d gaussian splatting via analytic integration. In *European conference on computer vision*, pp. 281–297. Springer, 2024.

Liqiang Lin, Yilin Liu, Yue Hu, Xingguang Yan, Ke Xie, and Hui Huang. Capturing, reconstructing, and simulating: The UrbanScene3D dataset. In *European Conference on Computer Vision*, pp. 93–109, 2022.

Minghua Liu, Mikaela Angelina Uy, Donglai Xiang, Hao Su, Sanja Fidler, Nicholas Sharp, and Jun Gao. Partfield: Learning 3d feature fields for part segmentation and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9704–9715, 2025.

Xiangrui Liu, Xinju Wu, Pingping Zhang, Shiqi Wang, Zhu Li, and Sam Kwong. Compgs: Efficient 3d scene representation via compressed gaussian splatting. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 2936–2944, 2024.

Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20654–20664, 2024.

Saswat Subhajyoti Mallick, Rahul Goel, Bernhard Kerbl, Markus Steinberger, Francisco Vicente Carrasco, and Fernando De La Torre. Taming 3dgs: High-quality radiance fields with limited resources. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

Simon Niedermayr, Christoph Neuhauser, and Rüdiger Westermann. Lightweight gradient-aware upscaling of 3d gaussian splatting images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 25862–25871, 2025.

Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20051–20060, 2024.

Kerui Ren, Lihan Jiang, Tao Lu, Mulin Yu, Linning Xu, Zhangkai Ni, and Bo Dai. Octree-gs: Towards consistent real-time rendering with lod-structured 3d gaussians. *arXiv preprint arXiv:2403.17898*, 2024.

Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulò, Peter Kontschieder, Angela Dai, and Matthias Nießner. L3dg: Latent 3d gaussian diffusion. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Yuan Shen, Duygu Ceylan, Paul Guerrero, Zexiang Xu, Niloy J Mitra, Shenlong Wang, and Anna Frühstück. Supergaussian: Repurposing video models for 3d super resolution. In *European Conference on Computer Vision*, pp. 215–233. Springer, 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.

Yecong Wan, Mingwen Shao, Yuanshuo Cheng, and Wangmeng Zuo. S2gaussian: Sparse-view super-resolution 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 711–721, 2025.

Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin CK Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *International Journal of Computer Vision*, 132(12):5929–5949, 2024a.

Yufei Wang, Zhihao Li, Lanqing Guo, Wenhan Yang, Alex C Kot, and Bihan Wen. Contextgs: Compact 3d gaussian splatting with anchor level context model. *Advances in neural information processing systems*, 37:51532–51551, 2024b.

Shiyun Xie, Zhiru Wang, Yinghao Zhu, Xu Wang, Chengwei Pan, and Xiwang Dong. Supergs: Consistent and detailed 3d super-resolution scene reconstruction via gaussian splatting. *arXiv preprint arXiv:2505.18649*, 2025.

Daheng Yin, Isaac Ding, Yili Jin, Jianxin Shi, and Jiangchuan Liu. Trackersplat: Exploiting point tracking for fast and robust dynamic 3d gaussians reconstruction. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–11, 2025.

Xiqian Yu, Hanxin Zhu, Tianyu He, and Zhibo Chen. Gaussiansr: 3d gaussian super-resolution with 2d diffusion priors. *arXiv preprint arXiv:2406.10111*, 2024a.

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19447–19456, 2024b.

Zongsheng Yue, Kang Liao, and Chen Change Loy. Arbitrary-steps image super-resolution via diffusion inversion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 23153–23163, 2025.

Huimin Zeng, Yue Bai, and Yun Fu. Arbitrary-scale 3d gaussian super-resolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 12304–12312, 2026.

Chaoyi Zhou, Xi Liu, Feng Luo, and Siyu Huang. Latent radiance fields with 3d-aware 2d representations. In *The Thirteenth International Conference on Learning Representations*, 2025.

Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21676–21685, 2024.

Matthias Zwicker, Hanspeter Pfister, Jeroen Van Baar, and Markus Gross. Ewa splatting. *IEEE Transactions on Visualization & Computer Graphics*, 8(03):223–238, 2002.