NeRF Is a Valuable Assistant for 3D Gaussian Splatting

Supplementary Material

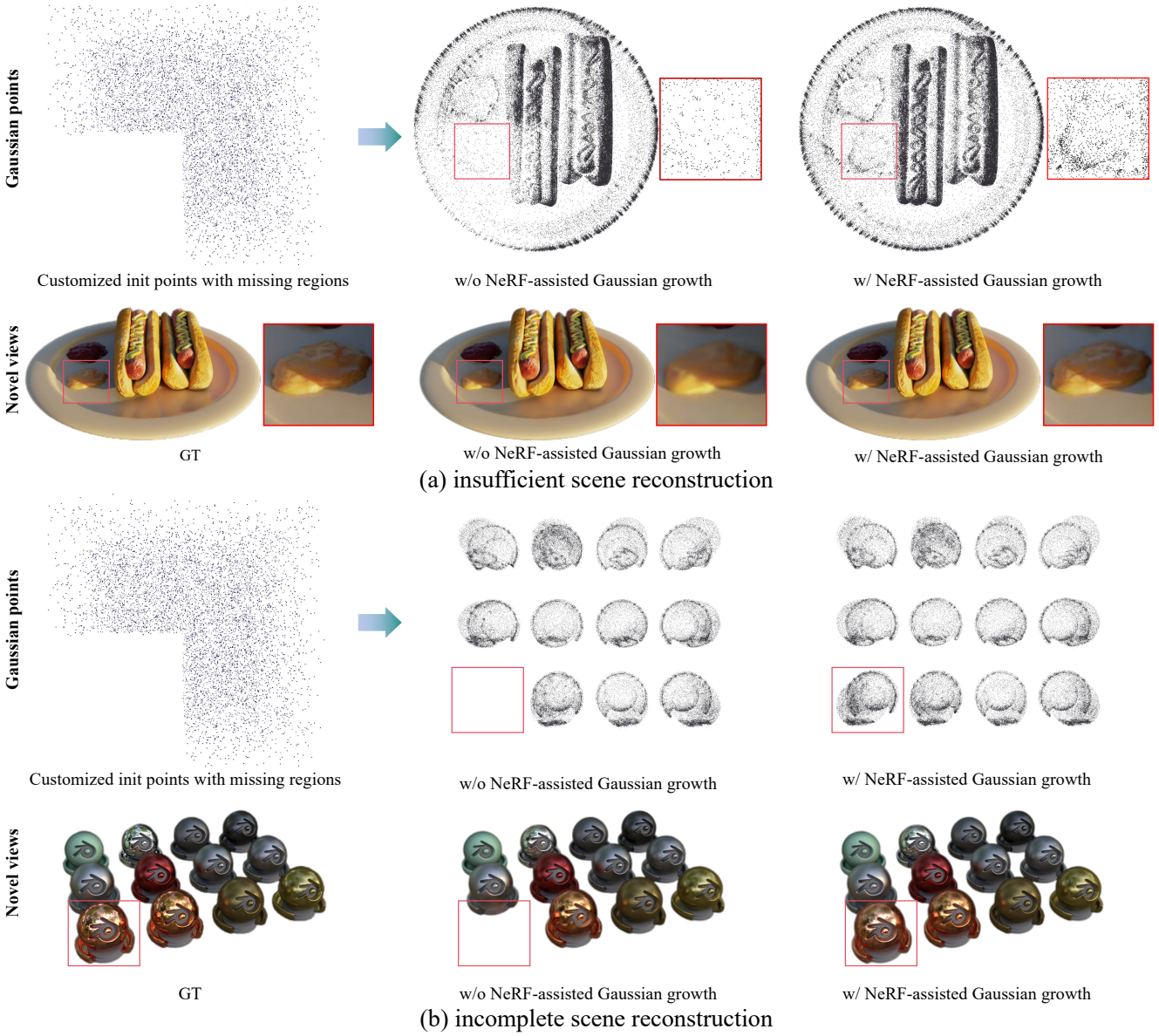

Figure 1. **Impact of NeRF-assisted Gaussian growth**. We initialize 3DGS using point clouds with missing regions to evaluate its scene perception range and sensitivity to initialization. Without NeRF-assisted Gaussian growth, 3DGS exhibits insufficient reconstruction (a) or incomplete reconstruction (b) in the missing areas. However, when employing the proposed NeRF-assisted Gaussian growth strategy in our method, these missing regions are successfully reconstructed. This demonstrates that NeRF significantly enhances the perception range of 3DGS, reducing its sensitivity to initialization and improving visual quality.

1. Analysis of Gaussian Adaptive Control from NeRF Branch

The continuous spatial representation of NeRF enables queries at any spatial location, allowing it to perceive the entire 3D scene. In contrast, individual Gaussian sphere in 3DGS has a limited perceptual range, making 3DGS sensitive to initialization and less effective in adaptive control. As shown in Figure 1, we deliberately design a Gaussian initialization with missing regions in certain spatial areas.

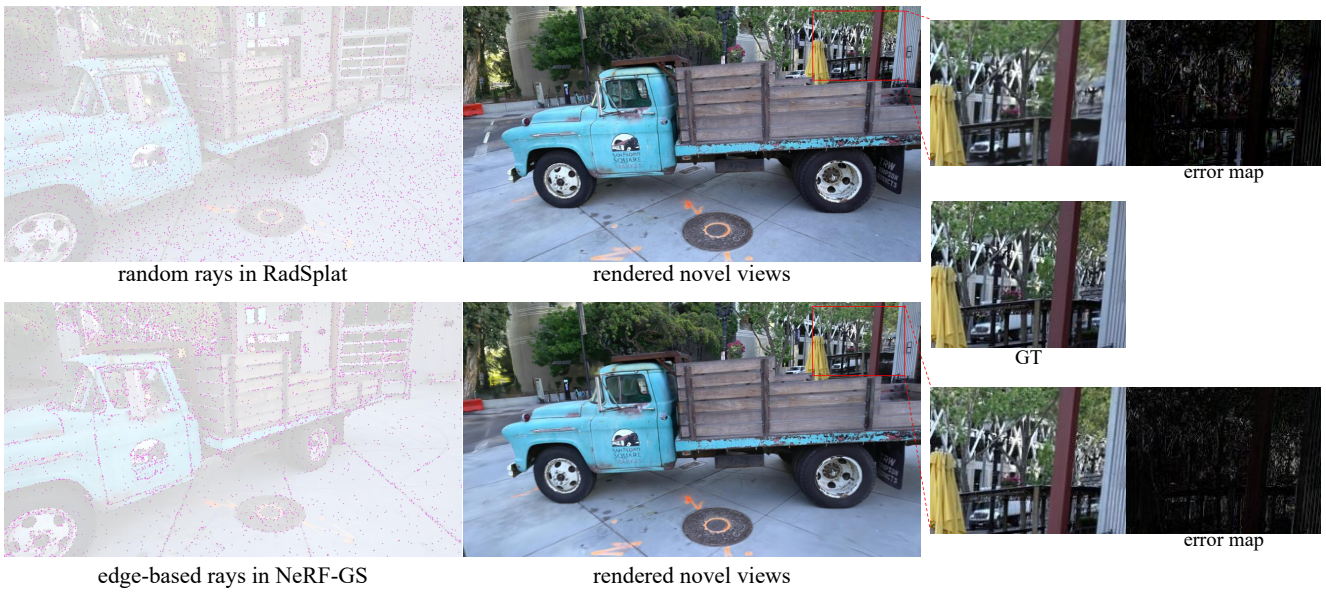

Figure 2. **Comparison of initialization with RadSplat**. NeRF-GS focuses more on the contours of the scene during ray sampling, alleviating the burden of position optimization in the GS branch while achieving superior visual results in regions with complex textures.

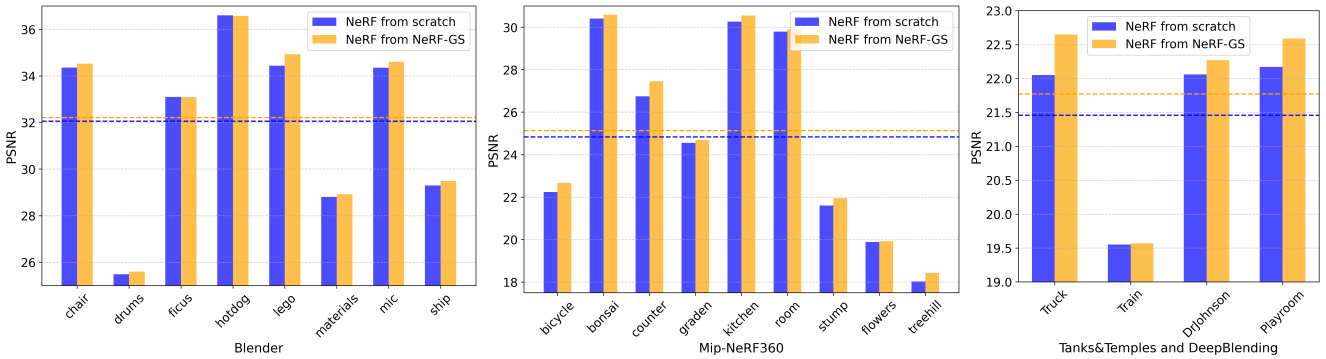

Figure 3. **Impact of joint optimization on the NeRF branch**. The dashed line indicates the mean PSNR. Given equivalent training iterations, the NeRF obtained through NeRF-GS outperforms training this NeRF independently. This demonstrates that dual-branch training not only benefits the GS branch but also enhances the performance of the NeRF branch.

After iterative optimization, it can be observed that GS allocates a limited number of Gaussians to these regions without assistance from the NeRF branch, and in extreme cases, it fails to perceive the missing areas entirely, resulting in poor or incomplete scene reconstruction. Conversely, our NeRF-assisted adaptive control strategy successfully senses these regions, significantly enhancing the global perceptual capability of the GS branch and reducing its sensitivity to initialization.

2. Analysis of Edge-based Initialization

NeRF-GS utilizes pre-trained NeRF to obtain candidate Gaussian positions. To enhance initialization efficiency, we incorporate an edge detection step that pre-identifies critical rays and increases their sampling probability during initialization. This design is predicated on the observation that Gaussian spatial distribution should ideally align with the contours of the actual 3D scene, with more Gaussians in textured areas and fewer in blank areas. In the baseline RadSplat, rays are sampled uniformly at random without discrimination, which we consider inefficient. To illustrate this, we conduct a visualization experiment in Figure 2, showing that our approach yields a Gaussian distribution that clusters around areas rich in texture, with fewer Gaussians in low-texture or empty regions. The rendering results demonstrate that our edge-based initialization method effectively captures complex scene textures, outperforming uniform sampling in accurately representing the scene.

Table 1. **Additional comparisons**. We evaluate the performance of the NeRF branch in NeRF-GS and compare it to Instant-NGP [2], which also utilizes a hash-based structure.

Method	DeepBlending			Tanks&Temples			Mip-NeRF360		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Instant-NGP	23.62	0.797	0.423	21.72	0.723	0.330	26.43	0.725	0.339
$\text{Branch}_{\text{NeRF}}$	22.43	0.784	0.441	21.11	0.718	0.338	25.12	0.722	0.343
$\text{Branch}_{\text{GS}}$	30.70	0.910	0.245	24.44	0.860	0.172	28.32	0.824	0.217

Table 2. **Additional ablation studies**. Numbers are PSNR metric. $\text{D}_{\text{3dgs}}^{\text{random}}$ and $\text{D}_{\text{3dgs}}^{\text{edge}}$ denote direct optimizing 3DGS after initialization using the random initialization and the proposed edge-based initialization, respectively.

	Tanks&Temples			DeepBlending		
	Truck	Train	Avg	Drjohnson	Playroom	Avg
w/o $\mathcal{L}_{\text{gs}}^{\text{vol}}$	26.10	22.48	24.29	30.02	31.07	30.55
w/o $\mathcal{L}_{\text{nerf}}$	25.44	21.15	23.30	28.79	29.46	29.13
$\text{D}_{\text{3dgs}}^{\text{random}}$	25.46	21.83	23.65	29.07	29.92	29.50
$\text{D}_{\text{3dgs}}^{\text{edge}}$	25.87	22.11	23.99	29.40	30.38	29.89
Full	26.27	22.61	24.44	30.17	31.23	30.70

3. Analysis of NeRF Branch Performance

Mutual Promotion between NeRF and GS Branches. While the primary aim of this work is to leverage NeRF characteristics to address 3DGS limitations, we have found that the GS branch also positively impacts the NeRF branch during joint training. As depicted in Figure 3, the NeRF branch trained jointly with the GS branch outperforms an independently optimized NeRF under the same number of iterations. This improvement arises from feature sharing and joint loss constraints between NeRF and GS branches, which enhance NeRF optimization as well. The simultaneous performance gains of both branches further confirm the complementary relationship between NeRF and 3DGS, offering insights for exploring integration with other forms of 3D representation.

Compare with Structurally Similar NeRF Method. We further compare the NeRF branch to the GS branch and the Instant-NGP [2] based on the same hash structure. It should be noted that this article focuses more on the improvement of the GS branch performance by NeRF, where we observe a significant performance improvement in the GS branch.

4. Additional Ablation Studies

We further conduct ablation studies on additional loss terms, including the introduced volume regularization [1] and the overall loss term of the NeRF branch, $\mathcal{L}_{\text{nerf}}$. Additionally, we evaluate the performance of directly optimizing 3DGS after initialization using the random initialization ($\text{D}_{\text{3dgs}}^{\text{random}}$) and the proposed edge-based initialization ($\text{D}_{\text{3dgs}}^{\text{edge}}$). The results, presented in Table 2, indicate a significant performance drop when $\mathcal{L}_{\text{nerf}}$ is removed, demonstrating that jointly optimizing the NeRF branch benefits the GS branch. Similarly, direct optimization of GS after initialization leads to performance degradation, validating the effectiveness of our proposed joint optimization strategy. Moreover, we observe that $\text{D}_{\text{3dgs}}^{\text{random}}$ underperforms compared to $\text{D}_{\text{3dgs}}^{\text{edge}}$, further confirming the superiority of our initialization strategy.

5. Per-scene Breakdown Results of NeRF-GS

We provide a detailed quantitative assessment of NeRF-GS across various scenes in Tables 3, 4 and 5, including metrics such as PSNR, SSIM, and LPIPS.

Table 3. **Per-scene results of Blender dataset of our method**.

	Full views								
	chair	drums	ficus	hotdog	lego	materials	mic	ship	Avg
PSNR	35.36	26.34	35.15	37.81	36.45	30.873	36.78	30.9	33.71
SSIM	0.985	0.948	0.9852	0.984	0.983	0.962	0.988	0.887	0.970
LPIPS	0.012	0.047	0.013	0.019	0.014	0.036	0.0075	0.111	0.032
	12 views								
	chair	drums	ficus	hotdog	lego	materials	mic	ship	Avg
PSNR	28.32	22.67	26.48	29.58	26.18	24.26	29.02	24.21	26.34
SSIM	0.950	0.8991	0.9371	0.942	0.912	0.888	0.966	0.799	0.912
LPIPS	0.040	0.082	0.035	0.063	0.081	0.106	0.027	0.203	0.080
	8 views								
	chair	drums	ficus	hotdog	lego	materials	mic	ship	Avg
PSNR	25.95	20.58	23.12	27.27	25.01	20.83	25.72	22.93	23.92
SSIM	0.917	0.871	0.892	0.937	0.885	0.834	0.941	0.773	0.881
LPIPS	0.061	0.114	0.101	0.099	0.101	0.184	0.112	0.225	0.124

Table 4. **Per-scene results of Tanks&Temples and DeepBlending datasets of our method**.

	Tanks&Temples			DeepBlending		
	Truck	Train	Avg	Dr Johnson	Playroom	Avg
PSNR	26.27	22.61	24.44	30.17	31.23	30.70
SSIM	0.887	0.833	0.860	0.91	0.914	0.912
LPIPS	0.127	0.195	0.161	0.235	0.238	0.237

Table 5. **Per-scene results of Mip-NeRF360 dataset of our method**.

	bicycle	bonsai	counter	graden	kitchen	room	stump	flowers	treehill	Avg
PSNR	25.52	33.97	30.5	27.84	32.56	32.78	27.08	21.71	22.99	28.32
SSIM	0.695	0.957	0.93	0.868	0.939	0.941	0.785	0.613	0.626	0.817
LPIPS	0.327	0.145	0.144	0.102	0.102	0.155	0.206	0.314	0.395	0.210

References

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

[2] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, 2022. 3