Faster and Better 3D Splatting via Group Training

Supplementary Material

A. Proof of Property 1: Opacity-based Effective Gaussians Densification

Under the assumptions of mutual independence between Gaussian attributes and intra-primitive parameter independence, the partial derivatives $[\frac{\partial \Delta x}{\partial x_{\mathrm{m}}}, \frac{\partial \Delta y}{\partial y_{\mathrm{m}}}]$ for any arbitrary Gaussian admit the following derivation:

$$\frac{\partial \alpha}{\partial G} = O + \frac{\partial O}{\partial G} G = O + 0 \cdot G = O, \tag{1}$$

$$\begin{aligned} \frac{\partial L}{\partial x_{\mathrm{m}}} &= \sum_{\text{pixel}} \frac{\partial L}{\partial G_m^{\mathrm{2D}}} \frac{\partial G_m^{\mathrm{2D}}}{\partial \Delta x} \frac{\partial \Delta x}{\partial x_{\mathrm{m}}} \\ &= \sum_{\text{pixel}} o_m \frac{\partial L}{\partial \alpha_m} \frac{\partial G_m^{\mathrm{2D}}}{\partial \Delta x} \frac{\partial \Delta x}{\partial x_{\mathrm{m}}} \\ &= o_m \sum_{\text{pixel}} \frac{\partial L}{\partial \hat{C}} \frac{\partial \hat{C}}{\partial \alpha_m} \frac{\partial G_m^{\mathrm{2D}}}{\partial \Delta x} \frac{\partial \Delta x}{\partial x_{\mathrm{m}}}, \end{aligned} \tag{2}$$

$$\frac{\partial L}{\partial y_{\mathrm{m}}} = o_m \sum_{\text{pixel}} \frac{\partial L}{\partial \hat{C}} \frac{\partial \hat{C}}{\partial \alpha_m} \frac{\partial G_m^{\mathrm{2D}}}{\partial \Delta y} \frac{\partial \Delta y}{\partial y_{\mathrm{m}}}, \tag{3}$$

where $[\frac{\partial \Delta x}{\partial x_{\mathrm{m}}}, \frac{\partial \Delta y}{\partial y_{\mathrm{m}}}]$ remain constant parameters determined by the resolution $[W, H]$; $[\frac{\partial G_m^{\mathrm{2D}}}{\partial \Delta x}, \frac{\partial G_m^{\mathrm{2D}}}{\partial \Delta y}]$ derive from the scale, the rotation and the world coordinates of Gaussian primitives (independent of their opacity); and $\frac{\partial L}{\partial \hat{C}}$ represents the loss gradient with respect to the current pixel value.

Given that $\hat{C}$ is formulated as the composite rendering of N Gaussians in Eq. (4), the derivative $\frac{\partial \hat{C}}{\partial \alpha_m}$ admits computation via Eq. (5).

$$\begin{aligned} \hat{C} = & \underbrace{\sum_{i=1}^{m-1} \alpha_i c_i \prod_{j=1}^{i-1} (1-\alpha_j)}_{\text{Before Gaussian } m} + \underbrace{\alpha_m c_m \prod_{j=1}^{m-1} (1-\alpha_j)}_{\text{Gaussian } m} \\ & + \underbrace{\sum_{i=m+1}^{N} \overbrace{\prod_{j=1}^{m-1} (1-\alpha_j)}^{\text{Before G } m} \alpha_i c_i (1-\alpha_m) \overbrace{\prod_{j=m+1}^{i-1} (1-\alpha_j)}^{\text{After G } m}}_{\text{After Gaussian } m} \\ & + \underbrace{\prod_{i=1}^{N} (1-\alpha_i) c_{\mathrm{bg}}}_{\text{background}}, \end{aligned} \tag{4}$$

$$\frac{\partial \hat{C}}{\partial \alpha_m} = \overbrace{\prod_{j=1}^{m-1} (1-\alpha_j)}^{\text{Before G } m} \left[c_m - \sum_{i=m+1}^{N} \alpha_i c_i \overbrace{\prod_{j=m+1}^{i-1} (1-\alpha_j)}^{\text{After G } m} \right] - \frac{c_{\mathrm{bg}} T_N}{1-\alpha_m} \tag{5}$$

	Sampling Strategy	Tanks&Temples [6]					
		PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓
3DGS*	–	23.730	0.8491	0.176	4.6	430	15.3
Group Training	Imp. score	23.672	0.8486	**0.174**	5.8	593	15.7
	Vol.	23.718	0.8462	0.182	5.1	493	12.4
	Opac.	**23.850**	**0.8500**	0.176	**4.5**	**383**	**11.0**
	Vol.+Opac.	23.684	0.8475	0.179	4.8	438	11.9

Table 1. **Quantitative evaluation of training efficiency on the Tanks&Temples [6] reconstructed by 3DGS [5].** $*$ indicates that we retrain the model. PM stands for GPU peak memory allocation, with Size in MB and Time in minutes. Imp. score = Importance score based, Vol. = Volume-based, Opac. = Opacity-based, Vol.+Opac. = Volume & Opacity-based.

	Sampling Strategy	Deep Blending [4]					
		PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓
3DGS*	–	29.503	0.9038	**0.244**	7.8	677	25.2
Group Training	Imp. score	29.589	0.9051	0.246	8.5	765	23.2
	Vol.	29.448	0.9036	0.251	7.5	623	19.8
	Opac.	**29.768**	**0.9067**	0.245	**6.8**	**489**	**17.2**
	Vol.+Opac.	29.619	0.9048	0.247	7.0	533	19.0

Table 2. **Quantitative evaluation of training efficiency on the Deep Blending [4] reconstructed by 3DGS [5].** Group Training with Opacity-based Prioritised Sampling demonstrates the fastest reconstruction speed and superior performance compared to other sampling strategies.

Subsequently, the mathematical expectation of this derivative is formally established through Eq. (6)

$$\begin{aligned} \mathbb{E}\left[\frac{\partial \hat{C}}{\partial \alpha_m}\right] &= \overbrace{(1-\alpha_0)^{m-1}}^{\text{Before G } m} \left\{ c_0 - c_0\alpha_0 \sum_{i=m+1}^{N} \mathbb{E}\left[\overbrace{\prod_{j=m+1}^{i-1} (1-\alpha_j)}^{\text{After G } m} \right] \right\} - \frac{c_{\mathrm{bg}} T_{\text{sta.}}}{1-\alpha_0} \\ &= (1-\alpha_0)^{m-1} \left[c_0 - c_0\alpha_0 \sum_{i=m+1}^{N} (1-\alpha_0)^{i-m-1} \right] - \frac{c_{\mathrm{bg}} T_{\text{saturation}}}{1-\alpha_0} \\ &= \frac{(c_0 - c_{\mathrm{bg}}) T_{\text{saturation}}}{1-\alpha_0} \\ &= \frac{(c_0 - c_{\mathrm{bg}}) T_{\text{saturation}}}{1 - \mathbb{E}[o_i]\, \mathbb{E}[G_i]}, \end{aligned} \tag{6}$$

B. Efficiency And Effectiveness For Various Sampling Strategies

We propose various sampling strategies for Group Training, incorporating Prioritized Sampling based on distinct sampling metrics. The sampling probability for each Gaussian primitive G_i is defined as follows:

$$p_i = \frac{\theta_i}{\sum_{i=1}^{N} \theta_i}, \tag{7}$$

	Mip-NeRF360 [1]			Tanks&Temples [6]			Deep Blending [4]			Blender [8]		
	PSNR ↑	Time ↓	Accel. ↑	PSNR ↑	Time ↓	Accel. ↑	PSNR ↑	Time ↓	Accel. ↑	PSNR ↑	Time ↓	Accel. ↑
3D-GS [5]	27.45	26.7	–	23.70	15.0	–	29.59	23.9	–	33.77	6.1	–
+Group Training	27.56	19.6	27%	23.85	11.0	27%	29.75	16.9	29%	33.81	**4.8**	21%
Mini-Splatting [3]	27.27	20.7	–	23.26	12.6	–	29.95	17.8	–	31.60	10.0	–
+Group Training	27.25	17.9	13%	23.10	9.9	21%	29.85	14.7	17%	31.98	**8.4**	16%
LightGaussian [2]	27.06	27.5	–	23.09	16.1	–	27.28	25.9	–	32.95	6.1	–
+Group Training	27.34	20.5	25%	23.55	11.9	26%	28.50	19.0	27%	33.18	**4.6**	24%

Table 3. **Quantitative comparisons on different baselines and datasets.** Group Training with 3DGS achieves faster reconstruction speed than Mini-Splatting across all datasets. Furthermore, Group Training demonstrates **consistent acceleration effects** on both 3DGS acceleration model (13%∼21% speedup on Mini-Splatting [3]) and compression model (24%∼27% speedup on LightGaussian [2]).Accel. = Acceleration Ratio in training time compared to the baseline.

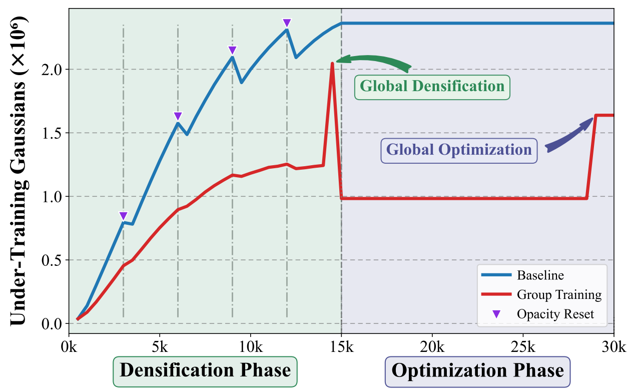


Figure 1. **Comparsion of Under-Training Gaussian Primitives.** Our Group-Training methodology selectively trains a subset of Gaussian primitives, demonstrating enhanced computational efficiency while mitigating loss of potentially critical points during opacity reset operations.

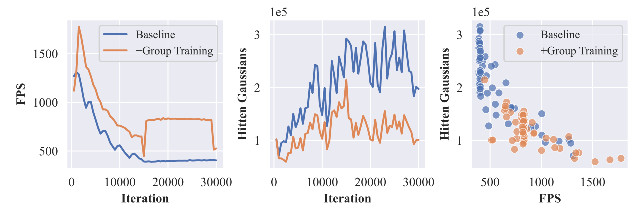


Figure 2. **Comparsion of Forward Rendering Efficiency.** We measured the number of hit Gaussians and forward rendering FPS throughout the training process. 3DGS with Group Training consistently demonstrated higher FPS and fewer hit Gaussians compared to the baseline method during training.

where θ_i represents the sampling metrics (opacity [2], volume [7] or importance score [1] [3, 9, 11]) of Gaussian primitive G_i, and N is the total number of Gaussian primitives. We also evaluated the metric which both Opacity and Volume are considered simultaneously, referred to as the Volume & Opacity-based method, as applied in [2]. The sampling metric θ_i for Volume & Opacity-based Prioritized Sampling is computed as follows:

$$\theta_i = \alpha_i \cdot V_i, \tag{8}$$

where α_i represents the opacity and V_i represents the volume of Gaussian primitive G_i.

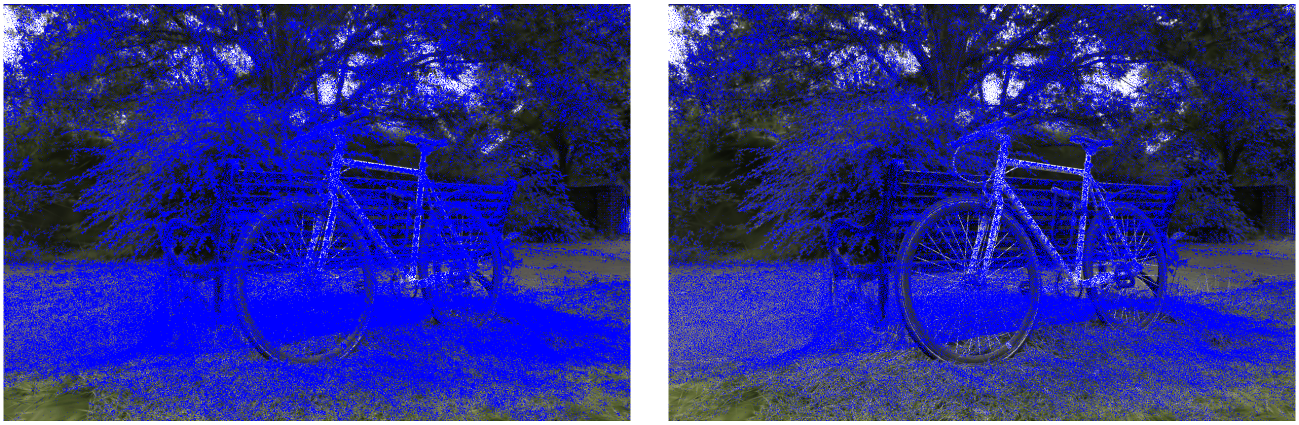
We conducted experiments using 3D Gaussian Splatting (3DGS) on two datasets: Tanks&Temples [6] and Deep Blending [4], both captured with camera-based systems. The comprehensive comparative results are presented in Tab. 1 and Tab. 2. Our results demonstrate that Group Training with Opacity-based Prioritized Sampling (OPS) consistently achieves significant improvements in both reconstruction speed and the quality of 3DGS models. Additionally, the reconstructed models exhibit greater compactness, as evidenced by a marked reduction in redundant Gaussian primitives.

However, volume and importance scores are not the most effective sampling metrics, as they fail to differentiate Gaussians that contribute to densification. This deficiency leads to abrupt vacancies in the Gaussian space under high sampling rates, causing the over-reconstruction and under-reconstruction [5]. Consequently, this exacerbates Gaussian densification, introducing redundancy between newly densified Gaussians and those already cached. The detailed analysis is provided in Sec. 3.2.

C. Temporal Evolution of Under-Training Gaussian Primitives

We visually compare the quantitative differences in under-training Gaussian primitives between Group-Training and 3DGS during scene reconstruction in Fig. 1. 3DGS with Group-Training reduces the training overhead by avoiding full optimization of all Gaussian primitives. Furthermore, during each opacity reset operation, the proposed

[1]Based on code: https://github.com/fatPeter/mini-splatting.git



(a) 3DGS (PSNR: 25.21dB Time: 34.1min) (b) 3DGS + Group Training (PSNR: 25.22dB Time: 21.8min)

Figure 3. **The visual comparison of Gaussian primitive distributions in the imaging plane.** We visualize the Gaussian projection information on the imaging plane during images rendering. **Left:** Gaussian distribution on the imaging plane for the "Bicycle" scene [1]. **Right:** 3DGS with Group Training achieves comparable rendering quality using fewer Gaussian primitives.

method retains a higher proportion of geometrically significant primitives compared to baseline. These retained elements, despite their low-opacity values, preserve critical structural information that contributes to scene geometry fidelity.

D. Comparison of Scene Representation Efficiency

Comparative analysis of per-iteration FPS and hit Gaussian counts was conducted during 3DGS reconstruction of the 'train' scene under baseline conditions and Group Training. Experimental results show the baseline method required 12.5 minutes to reach a PSNR of 21.985 dB, whereas Group Training with OPS acceleration attained a PSNR of 22.156 dB in just 9.3 minutes. These measurements confirm that Group Training consistently delivers accelerated rendering frame rates alongside a substantial reduction in hitten Gaussians count during training. Consequently, Group Training demonstrates higher scene representation efficiency, utilizing significantly fewer Gaussian primitives without compromising reconstruction quality.

E. Distribution of Gaussian Primitives in Imaging Plane Space

Fig. 3 illustrates the projection of rendering Gaussian primitives onto the imaging plane. Our Group Training approach significantly reduces the number of primitives required per image compared to the baseline, without compromising rendering quality, and further improves reconstruction speed.

F. Methodological Applicability

We perform comparative validation across two distinct 3DGS architectures: an acceleration-optimized model [3] and a compression-focused LightGaussian [2].

Empirical results demonstrate Group-Training's consistent efficacy across dataset scales, particularly evidenced by reduced temporal overhead in the Blender [8], as shown in Tab. 3. Crucially, our method synergistically integrates with existing acceleration techniques like Mini-Splatting [3], achieving compounded acceleration gains while providing sustained acceleration for compressed models with concurrent fidelity enhancement.

G. Detailed Experimental Results for All Scenes

We present the reconstruction results for all scenes using Group Training with Random Sampling (RS) and Opacity-based Prioritized Sampling (OPS), evaluated on 3D Gaussian Splatting (3DGS) [5] and Mip-Splatting [10]. The detailed results are provided in Tabs. 4 to 11.

The experimental results demonstrate that Group Training consistently delivers significant improvements in both reconstruction speed and quality across all tests, with the acceleration effect being particularly pronounced on complex datasets. Notably, Group Training with OPS achieves the fastest reconstruction times while maintaining optimal or near-optimal reconstruction quality.

We compare the effects of enabling RS and OPS during the Gaussian densification phase. The results indicate that Group Training with RS generates a significantly larger number of Gaussian primitives across all scenarios. For example, when reconstructing the "Bicycle" scene using Mip-Splatting, the high density of Gaussian primitives required the use of an NVIDIA A100 GPU for Group Training with RS. In contrast, Group Training with OPS produces sparser Gaussian primitives while delivering comparable or even superior reconstruction quality. Additionally, the reduced number of Gaussian primitives significantly alleviates the burden on peak memory usage.

References

[1] Jonathan T. Barron. Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields. In *CVPR*, 2022. 2, 3, 5, 7

[2] Zhiwen Fan, Kevin Wang, Kairun Wen, Zehao Zhu, Dejia Xu, and Zhangyang Wang. Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps, 2023. 2, 3

[3] Guangchi Fang and Bing Wang. Mini-splatting: Representing scenes with a constrained number of gaussians, 2024. 2, 3

[4] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep Blending for Free-Viewpoint Image-Based Rendering. In *SIGGRAPH*, 2018. 1, 2, 5, 7

[5] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 1, 2, 3, 5, 6

[6] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and Temples: Benchmarking Large-Scale Scene Reconstruction. In *SIGGRAPH*, 2017. 1, 2, 6, 8

[7] Yang Liu, Chuanchen Luo, Lue Fan, Naiyan Wang, Junran Peng, and Zhaoxiang Zhang. Citygaussian: Real-time high-quality large-scale scene rendering with gaussians. In *European Conference on Computer Vision*, pages 265–282. Springer, 2025. 2

[8] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 2, 3, 6, 8

[9] Michael Niemeyer, Fabian Manhardt, Marie-Julie Rakotosaona, Michael Oechsle, Daniel Duckworth, Rama Gosula, Keisuke Tateno, John Bates, Dominik Kaeser, and Federico Tombari. Radsplat: Radiance field-informed gaussian splatting for robust real-time rendering with 900+ fps. *arXiv.org*, 2024. 2

[10] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *CVPR*, 2024. 3, 7, 8

[11] Zhaoliang Zhang, Tianchen Song, Yongjae Lee, Li Yang, Cheng Peng, Rama Chellappa, and Deliang Fan. Lp-3dgs: Learning to prune 3d gaussian splatting. *arXiv preprint arXiv:2405.18784*, 2024. 2

	Grouping Iterations	Dr. Johnson						Playroom					
		PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓
3D Gaussian Splatting [5]	–	28.766	0.899	0.244	–	–	–	30.044	0.906	0.241	–	–	–
3D Gaussian Splatting∗	–	29.190	0.901	0.2442	9.0	782	26.8	29.981	0.907	0.2431	6.4	549	21.0
Group Training w/ RS	0∼15K	28.383	0.894	0.2517	8.8	733	24.2	29.987	0.908	0.2429	6.5	554	19.5
	0∼30K	28.701	0.902	0.2513	8.8	734	20.3	30.133	0.912	0.2448	6.5	552	16.6
Group Training w/ OPS	0∼15K	29.287	0.903	0.2430	8.1	592	21.7	30.138	0.909	0.2439	5.6	382	16.8
	0∼30K	29.309	0.904	0.2451	8.1	594	18.6	30.183	0.910	0.2448	5.6	380	15.1

Table 4. **Comprehensive quantitative evaluation results on the DeepBlending [4] reconstructed by 3DGS [5].** RS denotes Random Sampling, and OPS denotes Opacity-based Prioritized Sampling.

		Grouping Iterations	Mip-NeRF360 [1]								
			bicycle	flowers	garden	stump	treehill	bonsai	counter	kitchen	room
PSNR	3DGS [5]	–	25.246	21.520	27.410	26.550	22.490	31.980	28.700	31.317	30.632
	3DGS∗	–	25.205	21.484	27.397	26.620	22.514	32.202	28.980	31.222	31.377
	Group Training w/ RS	0∼15K	25.228	21.748	27.552	26.854	22.441	32.430	29.121	31.579	31.634
		0∼30K	25.217	21.806	27.463	27.095	22.671	31.975	28.850	31.319	31.438
	Group Training w/ OPS	0∼15K	25.260	21.751	27.434	26.809	22.402	32.312	29.031	31.539	31.699
		0∼30K	25.219	21.741	27.418	26.830	22.522	32.205	28.973	31.425	31.744
SSIM	3DGS [5]	–	0.771	0.605	0.868	0.775	0.638	0.938	0.905	0.922	0.914
	3DGS∗	–	0.765	0.605	0.866	0.773	0.634	0.942	0.908	0.927	0.919
	Group Training w/ RS	0∼15K	0.769	0.619	0.872	0.787	0.638	0.946	0.913	0.930	0.923
		0∼30K	0.769	0.616	0.871	0.795	0.642	0.942	0.909	0.927	0.920
	Group Training w/ OPS	0∼15K	0.770	0.617	0.869	0.785	0.635	0.945	0.911	0.929	0.922
		0∼30K	0.768	0.616	0.868	0.786	0.637	0.944	0.909	0.927	0.921
LPIPS	3DGS [5]	–	0.205	0.336	0.103	0.210	0.317	0.205	0.204	0.129	0.220
	3DGS∗	–	0.2103	0.3355	0.1069	0.2149	0.3240	0.2036	0.2001	0.1261	0.2184
	Group Training w/ RS	0∼15K	0.2094	0.3246	0.0985	0.1989	0.3182	0.1936	0.1909	0.1216	0.2110
		0∼30K	0.2225	0.3324	0.1061	0.2032	0.3281	0.2008	0.2008	0.1280	0.2198
	Group Training w/ OPS	0∼15K	0.2074	0.3262	0.1033	0.2044	0.3203	0.1968	0.1954	0.1245	0.2148
		0∼30K	0.2125	0.3307	0.1051	0.2075	0.3226	0.1992	0.1987	0.1272	0.2161
PM	3DGS∗	–	11.6	7.7	11.3	9.0	7.5	7.7	6.7	8.4	8.5
	Group Training w/ RS	0∼15K	12.1	8.8	12.0	10.9	9.0	8.4	7.1	8.5	8.8
		0∼30K	12.0	8.8	12.0	10.9	9.0	8.4	7.2	8.5	8.7
	Group Training w/ OPS	0∼15K	11.1	8.1	10.6	9.4	8.2	7.6	6.4	7.6	8.1
		0∼30K	11.1	8.0	10.6	9.3	8.4	7.6	6.4	7.6	8.1
Size	3DGS∗	–	1450	858	1391	1163	896	296	284	428	366
	Group Training w/ RS	0∼15K	1516	1019	1474	1457	1106	408	348	439	397
		0∼30K	1491	1016	1472	1452	1105	407	349	439	390
	Group Training w/ OPS	0∼15K	1192	795	1083	1078	899	267	225	277	287
		0∼30K	1195	790	1084	1060	921	267	225	275	291
Time	3DGS∗	–	34.1	24.0	35.9	27.2	24.0	20.5	22.8	28.3	23.7
	Group Training w/ RS	0∼15K	31.2	24.2	33.2	28.7	25.0	22.1	24.3	27.6	23.3
		0∼30K	24.8	20.1	27.3	23.6	20.3	20.6	21.7	24.2	20.6
	Group Training w/ OPS	0∼15K	26.9	21.1	27.4	23.5	22.0	18.5	20.4	22.4	20.6
		0∼30K	21.8	17.9	23.2	19.4	18.4	17.4	18.7	20.4	18.8

Table 5. **Comprehensive quantitative evaluation results on the Mip-NeRF360 [1] reconstructed by 3DGS [5].**

	Grouping Iterations	Train						Truck					
		PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓
3D Gaussian Splatting [5]	–	21.097	0.802	0.218	–	–	–	25.187	0.879	0.148	–	–	–
3D Gaussian Splatting∗	–	21.985	0.815	0.2063	3.6	257	12.5	25.409	0.882	0.1464	5.5	610	17.5
Group Training w/ RS	0∼15K	22.064	0.818	0.2031	3.9	278	11.7	25.482	0.885	0.1375	6.3	714	17.5
	0∼30K	21.910	0.812	0.2185	3.8	273	10.1	25.495	0.884	0.1460	6.3	716	14.4
Group Training w/ OPS	0∼15K	22.159	0.818	0.2040	3.6	228	10.7	25.524	0.884	0.1411	5.5	539	14.9
	0∼30K	22.156	0.816	0.2104	3.6	227	9.3	25.549	0.884	0.1424	5.5	540	12.6

Table 6. **Comprehensive quantitative evaluation results on the Tanks&Temples [6] reconstructed by 3DGS [5].**

		Grouping Iterations	Blender [8]							
			chair	drumps	ficus	hotdog	lego	materials	mic	ship
PSNR	3DGS [5]	–	33.83	26.15	34.87	37.72	35.78	30.00	35.36	30.80
	3DGS∗	–	35.581	26.258	35.481	38.004	36.062	30.461	36.649	31.677
	Group Training w/ RS	0∼15K	35.736	26.273	35.494	38.242	36.580	30.675	36.842	31.829
		0∼30K	35.637	26.224	35.436	38.142	36.441	30.588	36.786	31.765
	Group Training w/ OPS	0∼15K	35.688	26.270	35.487	38.145	36.435	30.569	36.719	31.800
		0∼30K	35.623	26.227	35.467	38.017	36.335	30.452	36.654	31.692
SSIM	3DGS∗	–	0.988	0.955	0.987	0.985	0.983	0.960	0.993	0.906
	Group Training w/ RS	0∼15K	0.988	0.955	0.987	0.986	0.985	0.962	0.993	0.909
		0∼30K	0.988	0.956	0.987	0.987	0.985	0.963	0.993	0.910
	Group Training w/ OPS	0∼15K	0.988	0.955	0.987	0.986	0.984	0.962	0.993	0.909
		0∼30K	0.988	0.955	0.987	0.986	0.984	0.961	0.992	0.909
LPIPS	3DGS∗	–	0.0104	0.0367	0.0118	0.0201	0.0161	0.0370	0.0064	0.1060
	Group Training w/ RS	0∼15K	0.0097	0.0355	0.0117	0.0170	0.0131	0.0340	0.0061	0.0998
		0∼30K	0.0107	0.0357	0.0118	0.0184	0.0140	0.0351	0.0063	0.1037
	Group Training w/ OPS	0∼15K	0.0099	0.0359	0.0118	0.0181	0.0139	0.0356	0.0064	0.1016
		0∼30K	0.0102	0.0364	0.0119	0.0189	0.0143	0.0367	0.0065	0.1042
PM	3DGS∗	–	3.1	2.9	2.7	2.6	2.9	2.6	2.6	2.8
	Group Training w/ RS	0∼15K	2.9	2.8	2.6	2.7	3.0	2.6	2.6	2.8
		0∼30K	2.9	2.8	2.6	2.7	3.0	2.6	2.6	2.8
	Group Training w/ OPS	0∼15K	2.8	2.7	2.5	2.6	2.8	2.6	2.6	2.7
		0∼30K	2.8	2.7	2.5	2.6	2.8	2.5	2.6	2.7
Size	3DGS∗	–	116	92	63	44	82	39	46	66
	Group Training w/ RS	0∼15K	88	78	39	46	97	47	42	69
		0∼30K	87	78	39	47	97	47	43	69
	Group Training w/ OPS	0∼15K	59	57	27	31	58	32	30	49
		0∼30K	61	58	27	31	58	30	30	49
Time	3DGS∗	–	7.4	6.6	5.3	5.9	6.5	4.9	5.3	6.6
	Group Training w/ RS	0∼15K	6.0	5.7	4.5	6.0	6.4	5.2	4.8	6.7
		0∼30K	5.5	5.3	4.3	5.5	5.8	4.9	4.6	6.2
	Group Training w/ OPS	0∼15K	5.3	5.3	4.1	5.2	5.3	4.6	4.5	5.8
		0∼30K	5.1	5.0	4.0	5.0	5.0	4.5	4.2	5.5

Table 7. **Comprehensive quantitative evaluation results on the Blender [8] reconstructed by 3DGS [5].**

	Grouping Iterations	Dr. Johnson						Playroom					
		PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓
Mip-Splatting∗	–	28.711	0.898	0.2431	10.5	981	39.3	30.005	0.907	0.2348	7.4	673	30.8
Group Training w/ RS	0∼15K	27.957	0.892	0.2526	10.1	898	34.3	29.901	0.908	0.2335	8.0	749	29.1
	0∼30K	28.500	0.902	0.2486	10.1	902	28.2	30.283	0.914	0.2354	8.0	748	24.7
Group Training w/ OPS	0∼15K	29.145	0.903	0.2393	9.4	732	31.0	30.185	0.910	0.2334	6.6	520	25.2
	0∼30K	29.271	0.904	0.2407	9.4	734	26.0	30.305	0.911	0.2360	6.6	518	22.0

Table 8. **Comprehensive quantitative evaluation results on the DeepBlending [4] reconstructed by Mip-Splatting [10].**

		Grouping Iterations	Mip-NeRF360 [1]								
			bicycle	flowers	garden	stump	treehill	bonsai	counter	kitchen	room
PSNR	Mip-Splatting∗	–	25.535	21.753	27.603	26.874	22.304	32.301	29.214	31.803	31.740
	Group Training w/ RS	0∼15K	25.664	21.960	27.726	27.124	22.371	32.299	29.291	31.719	31.756
		0∼30K	25.784	22.257	27.881	27.446	22.711	32.381	29.084	31.510	31.682
	Group Training w/ OPS	0∼15K	25.651	21.852	27.825	27.166	22.411	32.745	29.330	31.870	31.721
		0∼30K	25.634	21.844	27.858	27.145	22.441	32.653	29.255	31.864	31.856
SSIM	Mip-Splatting∗	–	0.792	0.641	0.877	0.790	0.639	0.945	0.914	0.931	0.925
	Group Training w/ RS	0∼15K	0.800	0.653	0.881	0.805	0.647	0.949	0.917	0.932	0.926
		0∼30K	0.803	0.656	0.882	0.816	0.658	0.947	0.914	0.931	0.925
	Group Training w/ OPS	0∼15K	0.796	0.648	0.879	0.803	0.645	0.948	0.915	0.932	0.926
		0∼30K	0.796	0.646	0.879	0.804	0.646	0.948	0.915	0.931	0.926
LPIPS	Mip-Splatting∗	–	0.1670	0.2727	0.0950	0.1889	0.2740	0.1881	0.1864	0.1194	0.2011
	Group Training w/ RS	0∼15K	0.1607	0.2628	0.0895	0.1736	0.2619	0.1823	0.1807	0.1163	0.1995
		0∼30K	0.1708	0.2748	0.0933	0.1751	0.2704	0.1874	0.1891	0.1213	0.2050
	Group Training w/ OPS	0∼15K	0.1655	0.2657	0.0922	0.1782	0.2683	0.1843	0.1847	0.1185	0.2026
		0∼30K	0.1673	0.2710	0.0929	0.1799	0.2723	0.1861	0.1864	0.1200	0.2029
PM	Mip-Splatting∗	–	15.6	9.5	12.3	10.9	10.1	8.4	7.4	9.2	9.7
	Group Training w/ RS	0∼15K	19.2	12.6	17.2	15.5	13.7	9.6	8.3	9.9	10.0
		0∼30K	19.2	12.7	17.1	15.6	13.7	9.6	8.3	9.9	10.0
	Group Training w/ OPS	0∼15K	15.8	10.6	13.9	12.4	11.4	8.4	7.2	8.4	9.0
		0∼30K	15.7	10.6	13.9	12.4	11.5	8.4	7.2	8.4	9.0
Size	Mip-Splatting∗	–	1957	1089	1475	1398	1232	388	364	523	517
	Group Training w/ RS	0∼15K	2494	1550	2194	2083	1763	560	491	615	558
		0∼30K	2489	1570	2187	2094	1764	560	490	610	560
	Group Training w/ OPS	0∼15K	1968	1230	1684	1564	1410	379	329	388	403
		0∼30K	1961	1230	1684	1562	1429	378	329	394	412
Time	Mip-Splatting∗	–	54.9	35.4	49.5	39.3	37.3	28.6	31.9	39.0	33.6
	Group Training w/ RS	0∼15K	43.7	39.9	57.7	46.3	42.1	32.2	34.1	40.3	32.8
		0∼30K	35.1	32.5	45.3	37.0	34.2	28.8	30.0	34.2	28.9
	Group Training w/ OPS	0∼15K	47.9	34.3	47.7	37.4	36.2	27.2	28.7	32.0	28.5
		0∼30K	37.7	28.3	38.8	30.0	29.5	24.9	26.1	28.5	25.9

Table 9. **Comprehensive quantitative evaluation results on the Mip-NeRF360 [1] reconstructed by Mip-Splatting [10].**

	Grouping Iterations	Train						Truck					
		PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PM ↓	Size ↓	Time ↓
Mip-Splatting∗	–	21.783	0.826	0.1892	4.4	351	19.5	25.714	0.893	0.1232	6.8	767	26.5
Group Training w/ RS	0∼15K	22.004	0.829	0.1861	4.8	405	18.9	25.901	0.896	0.1146	9.2	1123	30.3
	0∼30K	22.358	0.829	0.1975	4.7	403	16.3	25.934	0.896	0.1221	9.2	1119	24.4
Group Training w/ OPS	0∼15K	21.994	0.830	0.1859	4.4	346	17.4	25.921	0.896	0.1178	7.5	874	25.9
	0∼30K	22.167	0.829	0.1915	4.4	348	14.7	25.991	0.895	0.1205	7.6	876	21.1

Table 10. **Comprehensive quantitative evaluation results on the Tanks&Temples [6] reconstructed by Mip-Splatting [10].**

		Grouping Iterations	Blender [8]							
			chair	drumps	ficus	hotdog	lego	materials	mic	ship
PSNR	Mip-Splatting∗	–	35.773	26.357	35.890	38.267	36.354	30.645	36.934	31.738
	Group Training w/ RS	0∼15K	36.078	26.367	35.930	38.474	36.883	30.839	37.152	31.902
		0∼30K	35.910	26.293	35.865	38.376	36.860	30.780	36.968	31.931
	Group Training w/ OPS	0∼15K	35.981	26.369	35.912	38.396	36.748	30.756	37.067	31.823
		0∼30K	35.958	26.363	35.914	38.326	36.739	30.704	37.049	31.824
SSIM	Mip-Splatting∗	–	0.988	0.956	0.988	0.986	0.984	0.961	0.993	0.907
	Group Training w/ RS	0∼15K	0.989	0.956	0.988	0.987	0.986	0.963	0.993	0.909
		0∼30K	0.989	0.957	0.988	0.987	0.986	0.964	0.993	0.911
	Group Training w/ OPS	0∼15K	0.989	0.956	0.988	0.987	0.985	0.963	0.993	0.909
		0∼30K	0.989	0.956	0.988	0.987	0.985	0.963	0.993	0.909
LPIPS	Mip-Splatting∗	–	0.0109	0.0366	0.0111	0.0186	0.0150	0.0359	0.0062	0.1022
	Group Training w/ RS	0∼15K	0.0101	0.0355	0.0110	0.0163	0.0126	0.0330	0.0059	0.0980
		0∼30K	0.0110	0.0358	0.0111	0.0173	0.0132	0.0341	0.0060	0.0998
	Group Training w/ OPS	0∼15K	0.0105	0.0360	0.0111	0.0169	0.0133	0.0343	0.0060	0.0997
		0∼30K	0.0105	0.0359	0.0111	0.0171	0.0134	0.0349	0.0061	0.1001
PM	Mip-Splatting∗	–	3.0	3.0	2.7	2.7	2.9	2.6	2.8	2.9
	Group Training w/ RS	0∼15K	3.3	3.3	2.9	2.9	3.3	2.8	3.0	3.4
		0∼30K	3.3	3.4	2.9	2.9	3.3	2.8	3.0	3.4
	Group Training w/ OPS	0∼15K	3.1	3.1	2.7	2.7	3.1	2.7	2.9	3.1
		0∼30K	3.1	3.1	2.7	2.7	3.1	2.7	2.9	3.1
Size	Mip-Splatting∗	–	90	98	51	51	76	40	64	83
	Group Training w/ RS	0∼15K	141	146	79	70	136	71	94	146
		0∼30K	141	147	80	71	136	71	94	146
	Group Training w/ OPS	0∼15K	109	115	57	55	98	51	72	109
		0∼30K	109	115	57	55	99	51	72	109
Time	Mip-Splatting∗	–	9.2	8.8	6.2	8.4	8.8	6.2	9.3	10.7
	Group Training w/ RS	0∼15K	9.6	9.7	6.9	8.8	9.8	7.3	9.8	12.1
		0∼30K	8.5	8.7	6.4	8.1	8.8	6.9	8.5	10.8
	Group Training w/ OPS	0∼15K	8.6	8.8	6.2	7.8	8.6	6.6	8.4	10.4
		0∼30K	7.8	8.0	6.0	7.3	7.7	6.3	7.6	9.6

Table 11. **Comprehensive quantitative evaluation results on the Blender [8] reconstructed by Mip-Splatting [10].**