

FACT-GS: Frequency-Aligned Complexity-Aware Texture Reparameterization for 2D Gaussian Splatting

Supplementary Material

7. Appendix

7.1. Author Contributions

Tianhao Xie and Linlian Jiang contributed equally to this work. Tianhao Xie designed the algorithm, implemented it, and conducted the experiments. Linlian Jiang led the method formulation and was primarily responsible for manuscript writing.

7.2. Datasets

We evaluate our method on five standard benchmarks for novel view synthesis, covering both synthetic and real-world scenes. The experimental scenes are listed below:

- **NeRF Synthetic** [26]: all scenes.
- **MipNeRF 360 v2** [3]: publicly available scenes including Bicycle, Counter, Garden, Room, Bonsai, Kitchen, and Stump.
- **DTU** [16]: Scan105, Scan110, Scan37, and Scan63.
- **Tanks & Temples** [18]: Train, Truck, and Church.
- **LLFF** [25]: Horns and Fortress.

7.3. Budgets for Fig. 3 and Fig. 1

- **NeRF Ship**: 10%.
- **NeRF Chair**: 10%.
- **NeRF Hotdog**: 1%
- **Tanks & Temples Truck**: 1%.
- **Tanks & Temples Train**: 1%
- **Tanks & Temples Church**: 10%
- **MipNeRF 360 Kitchen**: 10%.

7.4. Ablation on Texture Parameters

We evaluated the quality of the novel view synthesis as the number of per-Gaussian texture parameters increases from 100 to 3600 for Textured GS and our method in 5 datasets, using 10% of default 2DGS primitives (the experiments on MipNeRF-360 were conducted with 1% of default 2DGS primitives due to the GPU memory issue), as shown in Fig. 8, and Table 7. The PSNR \uparrow , SSIM \uparrow , LPIPS \downarrow were reported, and the x axis in Fig. 8 represents the per-Gaussian texture parameters count.

Across all parameter budgets and datasets, FACT-GS achieves higher PSNR and SSIM, and lower LPIPS than Textured GS [15], which demonstrates that our method improves the efficiency of texture space utilization compared with uniform-grid textures, yielding substantially greater performance gains than simply increasing the texture resolution across datasets.

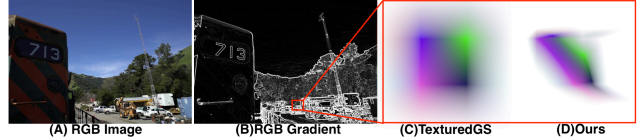


Figure 7. RGB Gradient and per-Gaussian Texture Visualization.

Textured GS	Ours
$5 \times 5 \times 4 = 100$	$4 \times 4 \times 6 = 96$
$10 \times 10 \times 4 = 400$	$8 \times 8 \times 6 = 384$
$15 \times 15 \times 4 = 900$	$12 \times 12 \times 6 = 864$
$20 \times 20 \times 4 = 1600$	$16 \times 16 \times 6 = 1536$
$30 \times 30 \times 4 = 3600$	$24 \times 24 \times 6 = 3456$

Table 4. Exact texture parameters count of Textured GS and our method for the ablation results in Fig. 8.

Parameters Count. To ensure a fair comparison with Textured GS, we matched the parameter number of both methods as closely as possible; nevertheless, our method consistently uses fewer parameters. The texture parameters of Textured GS are computed as $\tau_{\text{tex}} \times \tau_{\text{tex}} \times 4$, while those of our method are $\tau_{\text{FACT}} \times \tau_{\text{FACT}} \times 6$. We report the exact number of texture parameters for both methods in Table 4.

7.5. Model Size

Model size is matched for TexturedGS and our method in the main paper with identical Gaussian counts and total texture parameter budgets; an explicit size-matched comparison with 2DGS is further reported in Tab. 5 and Fig. 9.

7.6. Interpreting RGB Gradient.

We provide additional qualitative evidence for texture capacity utilization in Fig. 7. RGB gradient magnitude provides a practical proxy for local spatial frequency and effective texel allocation under a fixed texel budget, as higher gradients correspond to more rapidly varying color signals. Fig. 7(C-D) visualizes per-Gaussian local texture footprints, where our method concentrates color variation into a more compact and structured support, producing sharper transitions and clearer edges than TexturedGS. Accordingly, Fig. 5 shows a right-shift in the RGB gradient distribution under parameter-matched settings, indicating increased effective utilization of texture capacity for high-frequency detail reconstruction.

Number of Primitives. We report the average number of primitives in the default 100% 2DGS setting in Table 6.

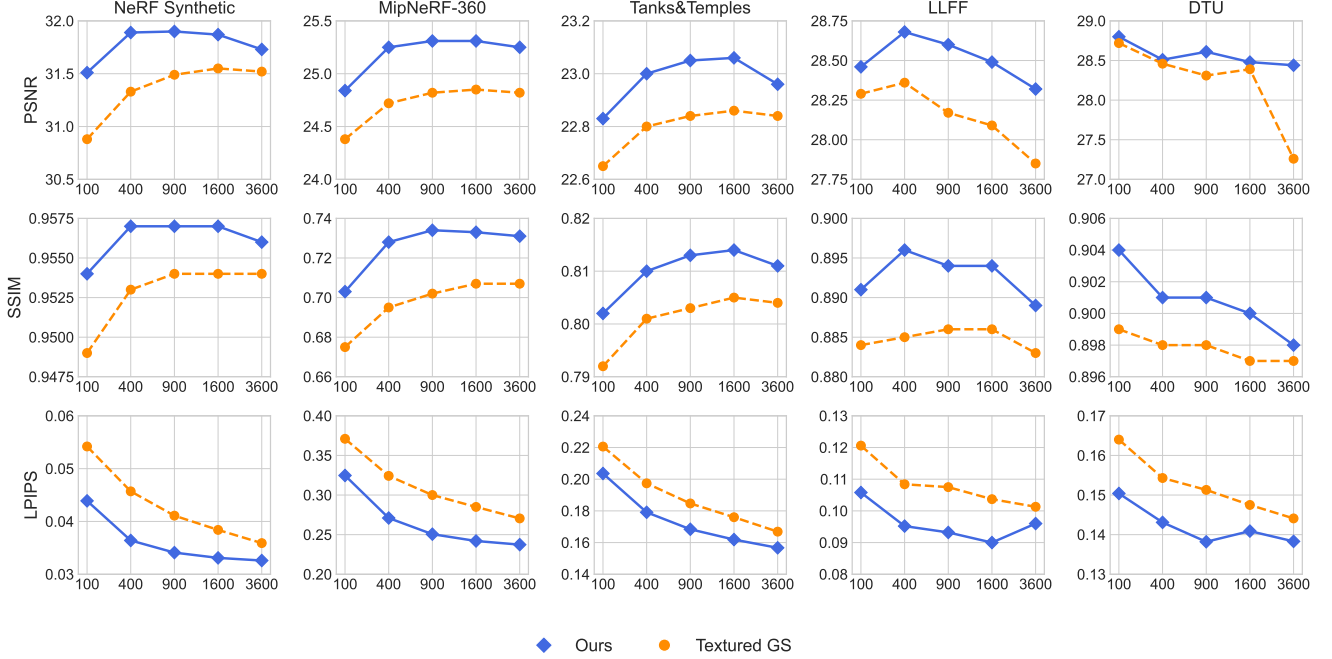


Figure 8. Ablation of the number of per-Gaussian texture parameters (the x axis represents the per-Gaussian texture parameters count). The per-Gaussian texture parameters increase from 100 to 3600, using 10% of default 2DGS primitives (the experiments on MipNeRF-360 were conducted with 1% of default 2DGS primitives due to the GPU memory issue). Across all datasets and parameter settings, FACT-GS achieves a consistent lead in all metrics (PSNR \uparrow , SSIM \uparrow , LPIPS \downarrow), demonstrating its superior texture-space efficiency compared with uniform-grid textures.

Method	Average Size (MB)	Tanks & Temples	LLFF	DTU
		PSNR (\uparrow) / SSIM (\uparrow) / LPIPS (\downarrow)		
2DGS	100.94	22.79 / 0.805 / 0.194	27.68 / 0.881 / 0.113	27.85 / 0.905 / 0.133
Ours	99.23	23.11 / 0.816 / 0.177	28.46 / 0.891 / 0.106	28.90 / 0.910 / 0.128

Table 5. Size-matched comparison with 2DGS.



Figure 9. Quantitative size-matched comparison with 2DGS.

7.7. Backpropagation of FACT Texture

Forward. Let $\mathbf{T}_i \in \mathbb{R}^{\tau \times \tau \times 4}$ denote the texture map of the i -th Gaussian, and $\mathbf{D}_i \in \mathbb{R}^{\tau \times \tau \times 2}$ its learned deformation field. Given a camera ray from pixel \mathbf{p} that intersects the i -th Gaussian at local coordinates (u, v) , the sampled displacement $(\Delta u, \Delta v)$ is computed by bilinear interpolation over \mathbf{D}_i :

$$\Delta u = \sum_{j \in N_{(u,v)}} b_j(u, v) D_j^u, \quad \Delta v = \sum_{j \in N_{(u,v)}} b_j(u, v) D_j^v, \quad (10)$$

where $N_{(u,v)}$ denotes the four neighboring texels surrounding (u, v) , and $b_j(u, v)$ are their corresponding bilinear interpolation weights. Let $\Phi(u, v) = (u, v) + (\Delta u, \Delta v)$ be the frequency-aligned sampling coordinate. The sampled color c from texture \mathbf{T}_i is then obtained as

$$c = \sum_{k \in N_{\Phi(u,v)}} b_k(\Phi(u, v)) T_k, \quad (11)$$

Number of primitives	
NeRF Synthetics [26]	91,396
MipNeRF 360 v2 [3]	2,763,229
DTU [16]	316,989
Tanks & Temple [18]	1,584,660
LLFF [25]	490,511

Table 6. The average number of primitives under the default 100% optimization setting.

where $N_{\Phi(u,v)}$ are the four neighboring texels around $\Phi(u, v)$ and $b_k(\Phi(u, v))$ are the corresponding interpolation weights.

Backward. The gradient of the sampled color c with respect to D_j^u at texel j is given by

$$\frac{\partial c}{\partial D_j^u} = \left(\sum_{k \in N_{\Phi(u,v)}} T_k \frac{\partial b_k(\Phi(u, v))}{\partial(u + \Delta u)} \right) \frac{\partial \Delta u}{\partial D_j^u}. \quad (12)$$

Substituting Eq. (10) into Eq. (12) yields

$$\frac{\partial c}{\partial D_j^u} = \left(\sum_{k \in N_{\Phi(u,v)}} T_k \frac{\partial b_k(\Phi(u, v))}{\partial(u + \Delta u)} \right) b_j(u, v), \quad (13)$$

which defines the differentiable gradient flow from the rendered color c to the deformation field \mathbf{D}_i .

Methods	NeRF Synthetic [26]			MipNeRF 360v2 [3]			DTU [16]			Tanks & Temples [18]			LLFF [25]		
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
Textured GS (100)	30.88	0.949	0.0542	24.38	0.675	0.3710	28.72	0.899	0.1640	22.65	0.792	0.2206	28.29	0.884	0.1206
Ours (96)	31.51	0.954	0.0439	24.84	0.703	0.3247	28.80	0.904	0.1504	22.83	0.802	0.2036	28.46	0.891	0.1058
Textured GS (400)	31.33	0.953	0.0457	24.72	0.695	0.3242	28.46	0.898	0.1543	22.80	0.801	0.1974	28.36	0.885	0.1084
Ours (384)	31.89	0.957	0.0364	25.25	0.728	0.2710	28.51	0.901	0.1431	23.00	0.810	0.1791	28.68	0.896	0.0952
Textured GS (900)	31.49	0.954	0.0411	24.82	0.702	0.3000	28.31	0.898	0.1513	22.84	0.803	0.1847	28.17	0.886	0.1075
Ours (864)	31.90	0.957	0.0341	25.31	0.734	0.2506	28.61	0.901	0.1382	23.05	0.813	0.1684	28.60	0.894	0.0932
Textured GS (1600)	31.55	0.954	0.0383	24.85	0.707	0.2850	28.39	0.897	0.1475	22.86	0.805	0.1760	28.09	0.886	0.1037
Ours (1536)	31.87	0.957	0.0331	25.31	0.733	0.2420	28.48	0.900	0.1409	23.06	0.814	0.1619	28.49	0.894	0.0900
Textured GS (3600)	31.52	0.954	0.0359	24.82	0.707	0.2705	27.26	0.897	0.1441	22.84	0.804	0.1669	27.85	0.883	0.1013
Ours (3456)	31.73	0.956	0.0326	25.25	0.731	0.2374	28.44	0.898	0.1383	22.96	0.811	0.1567	28.32	0.889	0.0960

Table 7. Ablation on the number of per-Gaussian texture parameters. In the Methods column, the numbers in parentheses indicate the exact texture parameter counts. We report PSNR \uparrow , SSIM \uparrow , and LPIPS \downarrow , which are the metrics used to plot the curves in Fig. 8. Best results are highlighted in **bold**.