TileGS: Adaptive Gaussian Densification Through Tile-Guided Perceptual Analysis

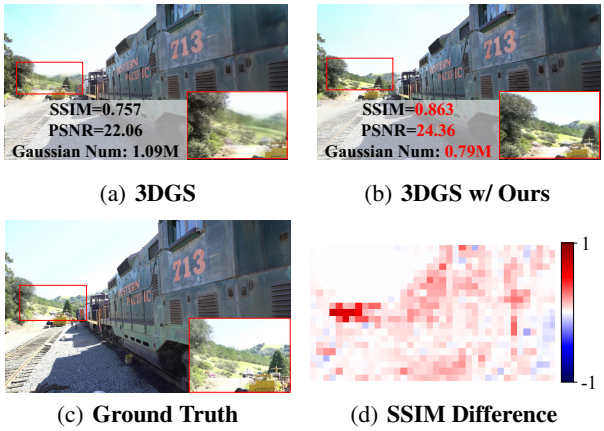
Yiwen Wang, Ran Yi*, Lizhuang Ma

Shanghai Jiao Tong University
{wangyiwen999, ranyi, lzma}@sjtu.edu.cn

Abstract

3D Gaussian Splatting (3DGS) has become a powerful technique for real-time novel view synthesis, using explicit, end-to-end optimized 3D Gaussians to represent scenes. However, its training objective is primarily based on pixel-wise photometric loss, and its densification strategy fails to account for structural consistency and localized perceptual priorities. As a result, 3DGS struggles to capture fine textures and boundary details in underconstrained areas, leading to inefficient use of representational capacity and degraded rendering quality in critical regions. To overcome this limitation, we introduce **TileGS**, a tile-wise, perceptually guided framework designed to refine scene representation based on local rendering quality. Our method features a tile-guided densification approach that performs per-tile perceptual analysis between rendered and ground-truth tiles to identify areas and Gaussians requiring refinement. Additionally, we incorporate a tile-level structural loss to enforce localized consistency during training. TileGS is designed to be a plug-and-play framework, seamlessly integrating into existing 3DGS pipelines with minimal adjustments. Experiments across multiple datasets demonstrate that TileGS improves rendering quality while maintaining an efficient representation, showcasing its versatility and effectiveness in diverse rendering scenarios.

Code — https://github.com/waleve/TileGS



(a) **3DGS** (b) **3DGS w/ Ours**

(c) **Ground Truth** (d) **SSIM Difference**

Figure 1: As shown in the zoomed-in insets, our method preserves fine textures and structural details that are severely blurred in the vanilla 3DGS output. The SSIM and PSNR metrics also demonstrate notable improvements. Furthermore, the per-tile SSIM (Ours - 3DGS) difference map highlights that most structural gains are concentrated in regions with rich textures and boundaries, confirming the effectiveness of our tile-guided perceptual refinement strategy.

Introduction

Novel view synthesis (NVS) is a core problem in neural rendering, which aims to generate photorealistic images from novel viewpoints given a set of input views. Early approaches such as Neural Radiance Fields (NeRF) (Mildenhall et al. 2020) and its variants (Müller et al. 2022; Barron et al. 2022, 2023; Yu et al. 2021) represent scenes using continuous volumetric radiance fields and optimize them via multi-view photometric supervision. While these methods achieve impressive visual fidelity, their reliance on dense ray marching and implicit representations leads to slow inference and limited real-time applicability.

Recently, 3D Gaussian Splatting (3DGS) (Kerbl et al. 2023) has emerged as a compelling solution for real-time novel view synthesis. It represents a scene as a collection of explicit anisotropic Gaussians and directly splats them into the image plane (Zwicker et al. 2001), replacing costly ray marching with a highly efficient rasterization-based pipeline. To progressively improve reconstruction quality during training, 3DGS further introduces an adaptive density control mechanism that inserts new Gaussians based on accumulated position gradients.

Building on this framework, recent works have extended 3DGS to a variety of downstream tasks, including VR rendering (Jiang et al. 2024), simulation (Feng et al. 2024; Xie et al. 2023), scene editing (Chen et al. 2023; Luo et al. 2024), 3d content generation (Wang et al. 2025; Yang et al. 2025) and dynamic scene reconstruction (Wu et al. 2024; Li et al. 2024; Yu et al. 2023; Wen et al. 2025), highlighting the versatility and generalizability of Gaussian-based representations.

However, the densification strategy in 3DGS's adaptive density control is global, heuristic-driven, and has its limi-

*Corresponding author.

tations: it relies on fixed thresholds and aggregate gradient magnitudes, without considering how well different regions are visually reconstructed. As a result, redundant Gaussians may accumulate in already well-modeled areas, increasing memory consumption and computational burden without contributing to visual improvements. Meanwhile, regions containing complex textures or fine-grained structures remain underrepresented due to insufficient gradient cues. These under-constrained regions, though critical to perceptual quality, are easily overlooked by purely gradient-based criteria, leading to visible artifacts and local degradation in rendering quality. For instance, in scenes like *train* scene (shown in Fig. 1) from the Tank&Temple (Barron et al. 2022) dataset, regions such as gravel tracks and background hill contain intricate textures and sharp structural variations that are perceptually important yet challenging to model. Because the accumulated position gradients in these areas remain relatively small, they are unlikely to surpass the global threshold for densification in 3DGS. This results in a sparse and inadequate representation of these regions, manifesting as texture blurring and structural artifacts in the rendered view. At the same time, excessive Gaussians are unnecessarily placed in redundant regions, inflating storage and memory usage. This imbalance highlights the need for a more localized and perceptually aware densification strategy.

Several recent works (Zhang et al. 2024b; Ye et al. 2024; Zhang et al. 2024a; Rota Bulò, Porzi, and Kontschieder 2024; Fang and Wang 2024) have attempted to address this issue by modifying the gradient accumulation strategy or dynamically adjusting the insertion frequency. Most of these approaches fall into one of two categories: either indirectly amplifying the gradient signal through reweighting or auxiliary supervision, or lowering the thresholds for Gaussian insertion to make densification more aggressive. While such methods do improve coverage, they remain fundamentally global and optimization-driven, lacking an explicit understanding of perceptual quality or spatial context.

An alternative perspective arises from the tile-based rendering architecture of 3DGS, originally introduced to accelerate splatting, which inherently divides the image into local regions. While designed for rendering performance, this tile-based structure also presents a unique opportunity: it enables localized perceptual analysis of rendered content at the tile level. Each tile provides a spatially confined view, capturing both structural and textural details of the scene. More importantly, it provides a natural interface to monitor the behavior and contribution of Gaussians in specific regions, which global gradient statistics cannot reveal directly.

This perspective suggests a new direction for density control in 3DGS. Rather than relying purely on global signals, we explore whether local perceptual feedback can serve as a more effective guide for representation refinement. Building on this idea, we propose **TileGS**, a new tile-wise perceptual guided framework that integrates local quality feedback into the density control process and achieves better reconstruction quality. First, we design a **tile-guided densification mechanism** to identify and densify under-represented regions by evaluating the structural similarity between each rendered tile and its ground-truth counterpart. Specifically, we conduct tile-level perceptual analysis by computing the Structural Similarity Index Measure (SSIM) for each tile, and treat those tiles with low scores as under-represented. The Gaussians contributing to these tiles are marked as candidates for densification, while those with minimal participation across tiles are pruned. This allows the model to allocate representational capacity more effectively, focusing on regions that are perceptually deficient. In addition to this tile-guided densification process, we further propose a **tile-level structural loss** that directly supervises training. By encouraging local structural consistency between rendered tiles and their references, this loss provides perceptual gradients that are more aligned with human visual sensitivity than conventional pixel-wise objectives.

In summary, this paper makes the following contributions:

- **Tile-wise Perceptual-Guided Densification**: We propose TileGS, a novel framework that integrates perceptual signals at the tile level to guide Gaussian densification, enabling the model to refine under-represented regions and prune redundant Gaussians based on local SSIM analysis.
- **Tile-level Structural Loss for Local Consistency**: To further enhance the structural integrity of rendered scenes, we introduce a tile-level structural loss that encourages localized geometric and textural consistency, aligning optimization with perceptual quality.
- **Plug-and-Play Capability**: TileGS is designed as a plug-and-play component that can be seamlessly integrated into existing 3D Gaussian Splatting pipelines, enhancing quality without architectural overhaul.

Related Works

Neural Rendering and Novel View Synthesis

Novel view synthesis has become a central topic in neural rendering, with early breakthroughs such as Neural Radiance Fields (NeRF) (Mildenhall et al. 2020) demonstrating the capability to generate photorealistic views from sparse multi-view images. Follow-up works like Mip-NeRF (Barron et al. 2022), Instant-NGP (Müller et al. 2022), and TensoRF (Chen et al. 2022a) have improved training speed, view generalization, and representation compactness. Beyond static view synthesis, NeRF-based models have also shown strong potential in a range of downstream applications. These include relighting (Srinivasan et al. 2021; Verbin et al. 2021; Boss et al. 2021), scene editing (Haque et al. 2023; Yuan et al. 2022), human avatar modeling (Zheng et al. 2023), dynamic scene reconstruction (Luiten et al. 2024), and even robotics and navigation tasks (Adamkiewicz et al. 2022; Kwon, Park, and Oh 2023; Zhu et al. 2025). Such versatility highlights the general-purpose nature of neural scene representations and motivates further exploration of efficient, structure-aware approaches for scene modeling and refinement.

Point-based Representations and Densification in 3D Gaussian Splatting

Point-based scene representations have emerged as a promising direction in neural rendering, offering a balance

between expressiveness and computational efficiency. Methods such as Neural Point-Based Graphics (Aliev et al. 2020), Point-NeRF (Xu et al. 2022), INPC (Hahlbohm et al. 2025) have been proposed to capture geometry and appearance using discrete primitives, often augmented with learned attributes for view-dependent effects.

Among these methods, 3D Gaussian Splatting (3DGS) stands out for its ability to achieve high-fidelity rendering by modeling the scene with anisotropic Gaussian primitives that are directly splatted into image space. Its explicit and rasterization-friendly nature allows it to bypass ray marching entirely, enabling real-time rendering while preserving the benefits of gradient-based optimization.

A core component of 3DGS is its adaptive density control mechanism, which dynamically inserts new Gaussians during training based on the accumulated magnitude of position gradients. This process aims to densify regions with insufficient detail or geometric support. While effective in improving scene representation, this strategy is inherently global and heuristic, relying on fixed thresholds and indirect signals of reconstruction quality.

Several recent efforts (Zhang et al. 2024a,b; Ye et al. 2024; Lyu et al. 2024; Fang and Wang 2024) have attempted to improve the densification process in 3D Gaussian Splatting by modifying the gradient accumulation rules or introducing alternative criteria for triggering Gaussian insertion. FreGS (Zhang et al. 2024a) proposes a frequency-domain regularization that encourages high-frequency signal retention during optimization. AbsGS (Ye et al. 2024) identifies inconsistencies in gradient accumulation across pixels and resolves them by analyzing inter-pixel conflicts. PixelGS (Zhang et al. 2024b) takes into account the actual pixel area covered by each Gaussian and uses this information to weight gradient contributions more effectively.

While these methods differ in formulation, they ultimately share the same objective: to artificially increase the magnitude or effectiveness of position gradients, thereby making it easier for Gaussians to exceed the densification threshold. In essence, they retain the original accumulation-based framework but aim to make it more responsive in practice.

Perceptual and Structural Guidance

Perceptual and structural priors have proven to be highly effective in image restoration, synthesis, and neural rendering. Metrics such as Structural Similarity Index Measure (SSIM) (Wang et al. 2004) and Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al. 2018) are widely used not only for evaluation, but also as training objectives to improve visual fidelity. These metrics capture human-perceivable attributes like texture coherence and structural consistency that are often missed by pixel-wise losses.

In neural view synthesis, especially within the NeRF family of models, perceptual supervision has played a significant role in enhancing realism. Several methods (Verbin et al. 2021; Chen et al. 2022b; Xie et al. 2023) have demonstrated that incorporating perceptual or structural cues—either through loss functions or model designs—can effectively guide the network to preserve textures, details, and view-dependent structures across complex scenes.

Inspired by these insights, we extend the use of perceptual and structure-aware guidance to the domain of 3D Gaussian Splatting. While prior 3DGS methods primarily rely on pixel-wise photometric losses and global statistics for densification, we explore how localized perceptual signals can provide finer supervision. In particular, we leverage tile-level SSIM to inform the refinement and pruning of Gaussians, and introduce a tile-based structural loss to supervise training. This design allows our model to prioritize regions that are perceptually significant and structurally complex, leading to better preservation of edges, textures, and spatial coherence.

Method

In this section, we present **TileGS**, a tile-wise perceptually guided density control framework designed to enhance the quality of 3D Gaussian Splatting. As shown in Fig. 2, our method leverages the existing tile-based rendering pipeline of 3DGS to evaluate localized reconstruction quality and infer the effectiveness of individual Gaussians.

Rethinking 3DGS Densification

In 3D Gaussian Splatting, new Gaussians are periodically inserted based on the accumulated position gradient of existing ones. For each Gaussian g_j, the system records the sum of position gradient magnitudes over a fixed training interval:

$$G_j = \sum_{t=t_0}^{t_0+\Delta t} \left\| \nabla^{(t)}_{\boldsymbol{\mu}_j} \mathcal{L}_{\text{recon}} \right\|_2, \tag{1}$$

where $\nabla^{(t)}_{\boldsymbol{\mu}_j} \mathcal{L}_{\text{recon}}$ is the gradient of the reconstruction loss with respect to the position of g_j at iteration t; t_0 represents the beginning iteration of the current accumulation interval, and Δt denotes the number of iterations over which gradients are aggregated (e.g., every 100 steps).

A new Gaussian is inserted if this accumulation exceeds a fixed threshold:

$$\text{Densify } g_j \text{ if } \quad G_j > \tau_{\text{grad}}. \tag{2}$$

While effective at increasing representational capacity, this strategy is purely gradient-driven and does not consider perceptual importance of local regions or localized rendering quality. This densification strategy has some limitations: 1. Since in some high-frequency, complex textured areas, the RGB values can be relatively similar, (e.g., hill in Fig. 1), these areas will have low position gradients, which hardly triggers the densification condition, resulting in sparse Gaussians in these regions, leading to under-representation of complex textures and fine-grained structures. 2. The strategy can also result in redundant Gaussians in already well-modeled areas.

Tile-Guided Refinement

To overcome the limitations of 3DGS's gradient-based density control, which ignores spatial structure and localized perceptual quality, we propose a tile-wise perceptual framework for adaptive Gaussian refinement. Our method assesses

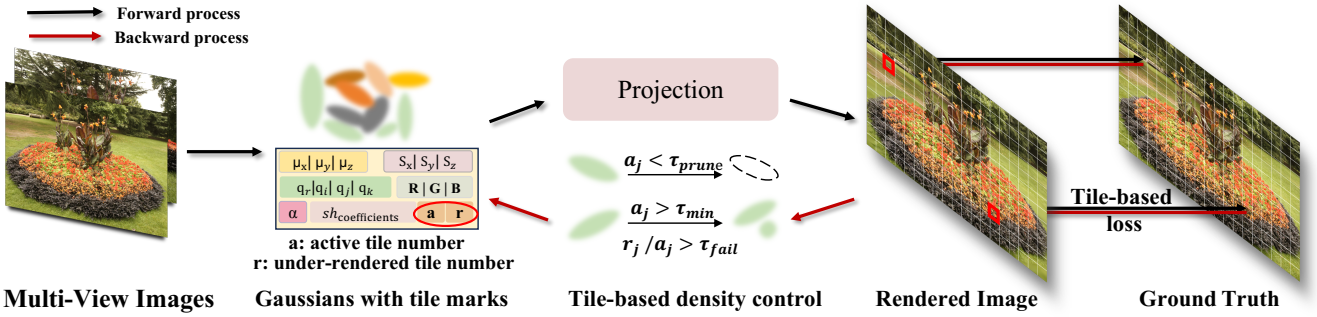


Figure 2: Overview of the TileGS pipeline. TileGS follows a similar pipeline to 3D Gaussian Splatting (3DGS), starting from multi-view images to optimize a set of 3D Gaussians for novel view synthesis. In addition to the original Gaussian parameters, we introduce two auxiliary variables: a, the number of times each Gaussian contributes to rendering, and r, the number of times it appears in low-quality tiles. These are used to guide a tile-based density control mechanism that prunes or refines Gaussians based on their perceptual effectiveness. Furthermore, we employ a tile-level perceptual loss between rendered and ground-truth tiles to enhance structural fidelity and local detail.

localized rendering quality to guide densification and pruning in a more targeted and visually aligned manner.

Local Quality Evaluation. Let the rendered image be partitioned into tiles $\mathcal{T} = \{T_1, T_2, \ldots, T_N\}$, where each T_i corresponds to a fixed-size image region. After rendering, we compute the Structural Similarity Index (SSIM) (Wang et al. 2004) between each rendered tile R_i and its ground-truth counterpart G_i:

$$s_i = \text{SSIM}(R_i, G_i), \quad \forall T_i \in \mathcal{T} \tag{3}$$

where SSIM is a perceptual metric that captures luminance, contrast, and structural consistency:

$$\text{SSIM}(R, G) = \frac{(2\mu_R\mu_G + C_1)(2\sigma_{RG} + C_2)}{(\mu_R^2 + \mu_G^2 + C_1)(\sigma_R^2 + \sigma_G^2 + C_2)}, \tag{4}$$

where μ, σ, and σ_{RG} denote local means, variances, and covariance between the rendered and ground-truth tiles.

We treat a tile as under-rendered if its SSIM score falls below a threshold τ_{SSIM}. These low-quality tiles indicate regions requiring refinement.

Gaussian Activity Tracking. To determine which Gaussians are responsible for poor rendering, we track their activity across tiles. For each Gaussian g_j, we maintain two counters:

- a_j (active tile number): the number of tiles where g_j is active;
- r_j (under-rendered tile number): the number of under-rendered tiles ($s_i < \tau_{\text{SSIM}}$) where g_j is active.

Whenever g_j contributes to a tile T_i, we increment:

$$a_j \mathrel{+}= 1, \quad r_j \mathrel{+}= \mathbb{I}[s_i < \tau_{\text{SSIM}}], \tag{5}$$

where $\mathbb{I}[\cdot]$ is the indicator function, s_i is the SSIM score of rendered tile T_i.

Tile-Guided Densification. To determine whether a Gaussian g_j should be densified or removed, we evaluate its failure ratio, denoted as f_j, measuring the proportion of under-rendered tiles it contributes to over a fixed training interval:

$$f_j = \frac{r_j}{a_j + \epsilon}, \tag{6}$$

where ϵ is a small constant to avoid division by zero.

Then we design a new tile-guided densification condition: A Gaussian is selected for densification if it frequently contributes to under-rendered tiles:

$$f_j > \tau_{\text{fail}}, \quad a_j > \tau_{\text{min}} \tag{7}$$

ensuring that the decision is statistically meaningful based on sufficient evidence. A high failure ratio f_j indicates that a Gaussian is frequently involved in under-rendered tiles, suggesting that its current representation is insufficient for capturing the local structure or appearance. By additionally requiring the Gaussian to appear in a minimum number of tiles ($a_j > \tau_{\text{min}}$), we ensure that the decision is statistically meaningful and not based on sparse evidence.

In contrast to gradient-based densification, which selects Gaussians based on the magnitude of their accumulated position gradients regardless of where or how they visually contribute, our criterion focuses directly on the rendering outcome with localized perceptual guidance. This enables the model to identify and refine the Gaussians that are persistently responsible for poor visual quality, particularly in regions with complex textures, fine details, or repetitive patterns.

Conversely, we prune Gaussians that exhibit persistently low activity across tiles:

$$a_j < \tau_{\text{prune}}, \tag{8}$$

indicating that they have limited spatial influence. Notably, since larger Gaussians naturally overlap with more tiles, they tend to accumulate higher a_j values, while small Gaussians with weak contribution and limited support are more likely to remain inactive. This criterion thus acts as a spatial relevance filter, selectively removing Gaussians that are unlikely to contribute to any meaningful region.

Tile-Level Structural Loss

While our tile-guided densification strategy adaptively improves the spatial distribution of Gaussians, it operates separately from the main end-to-end optimization loop. However, the rendering quality is also fundamentally determined by how Gaussians are trained to match appearance. In standard 3DGS training, the supervision is based on a combination of ℓ_1 loss and a D-SSIM term, which jointly encourage pixel-wise accuracy and global structural alignment. While effective in maintaining overall perceptual quality, this global loss formulation lacks fine-grained control over localized structural discrepancies. In particular, regions with high-frequency details, such as object boundaries, textures, and geometric edges, may still suffer from over-smoothing or underfitting, as the loss does not explicitly focus on spatially localized structure.

To address this issue, we propose a tile-level structural loss that evaluates the perceptual quality of small image patches independently. By computing SSIM within each tile, we provide region-aware supervision that complements the global D-SSIM term and enhances the model's ability to preserve localized features.

Tile-Level Structural Loss. To better guide structural learning during training, we propose a tile-level structural loss with adaptive weighting based on perceptual quality. Given a rendered image divided into tiles $\mathcal{T} = \{T_1, T_2, \ldots, T_N\}$, we compute the Structural Similarity Index (SSIM) (Wang et al. 2004) between each rendered tile R_i and its ground-truth counterpart G_i. Instead of using uniform weighting, we assign higher importance to tiles with lower SSIM scores via an exponential reweighting scheme:

$$w_i = \frac{\exp(-\alpha \cdot s_i)}{\sum_{j=1}^{N} \exp(-\alpha \cdot s_j)}, \quad s_i = \text{SSIM}(R_i, G_i), \quad (9)$$

where $\alpha > 0$ is a temperature-like parameter that controls the sharpness of the weighting distribution: larger values of α result in more aggressive emphasis on poorly rendered tiles.

The tile-level structural loss is then defined as:

$$\mathcal{L}_{\text{tile}} = \sum_{T_i \in \mathcal{T}} w_i \cdot (1 - s_i). \quad (10)$$

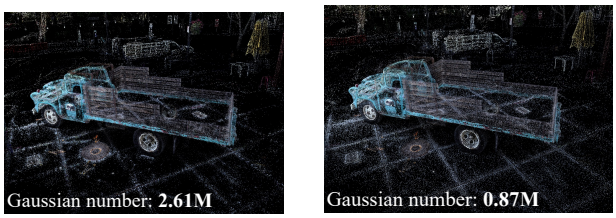
This formulation encourages the network to focus its learning capacity on structurally deficient areas, such as edges, fine textures, or regions with occlusion artifacts, which are often underrepresented in pixel-wise losses.

Joint Training Objective. We incorporate the tile-based structural loss as an auxiliary term in the overall optimization objective. The final loss is formulated as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{tile}} \cdot \mathcal{L}_{\text{tile}}, \quad (11)$$

where $\mathcal{L}_{\text{recon}}$ is the original photometric loss (typically a combination of ℓ_1), and λ_{tile} is a hyperparameter balancing the contribution of tile-level perceptual supervision.

By introducing structure-aware supervision at the tile level, our method bridges the gap between global fidelity and local realism. The model is incentivized to preserve geometrically meaningful details and region-level consistency, particularly in areas where traditional losses tend to oversmooth or blur fine features. As shown in our experiments, this results in sharper reconstructions and more faithful reproduction of complex textures and high-frequency content.



(a) **3DGS** (b) **3DGS w/ Ours**

Figure 3: Point distribution comparison between 3DGS and our method. The original 3DGS (a) tends to produce overly concentrated Gaussian distributions, resulting in redundant point clusters that limit expressiveness and waste storage. In contrast, our method (b) adaptively guides Gaussians to distribute more broadly and evenly across the scene, achieving better coverage with fewer points. This not only improves rendering quality in structurally important regions but also reduces memory overhead by avoiding redundancy.

Experiment

Experimental Setup

We conduct comprehensive comparisons against multiple representative Gaussian Splatting variants, including 3DGS (Kerbl et al. 2023), AbsGS (Ye et al. 2024), Pixel-GS (Zhang et al. 2024b), Mip-Splatting (Yu et al. 2024), Mini-Splatting-D (Fang and Wang 2024), and ResGS (Lyu et al. 2024). All models are evaluated on three standard datasets of varying scale and complexity: *Mip-NeRF 360* (Barron et al. 2022), *Tanks and Temples* (Knapitsch et al. 2017), and *Deep Blending* (Hedman et al. 2018).

Evaluation is performed from two perspectives: rendering quality and model size. Rendering quality is assessed using PSNR, SSIM(Wang et al. 2004), and LPIPS(Zhang et al. 2018), while model size is measured by the memory consumption of the final point cloud (in MB). To ensure fair comparisons, all models are trained for 30,000 iterations. All experiments are conducted on the same hardware setup.

Implementation Details

We set the tile size to 16×16 pixels by default, following the original tile-based rendering pipeline. For tile-guided densification, the failure ratio threshold is set to $\tau_{\text{fail}} = 0.999$, meaning that a Gaussian is selected for refinement if more than 99.9% of its associated tiles are under-rendered. We use the following hyperparameter settings in experiments: the minimum appearance threshold τ_{min} is set to 500; Gaussians with fewer than $\tau_{\text{prune}} = 200$ tile appearances are pruned; the perceptual quality threshold is set to $\tau_{\text{SSIM}} = 0.6$; and the tile loss weight is fixed at $\lambda_{\text{tile}} = 0.2$. Following 3DGS, we perform densification every 500 iterations, except during the first 1000 iterations.

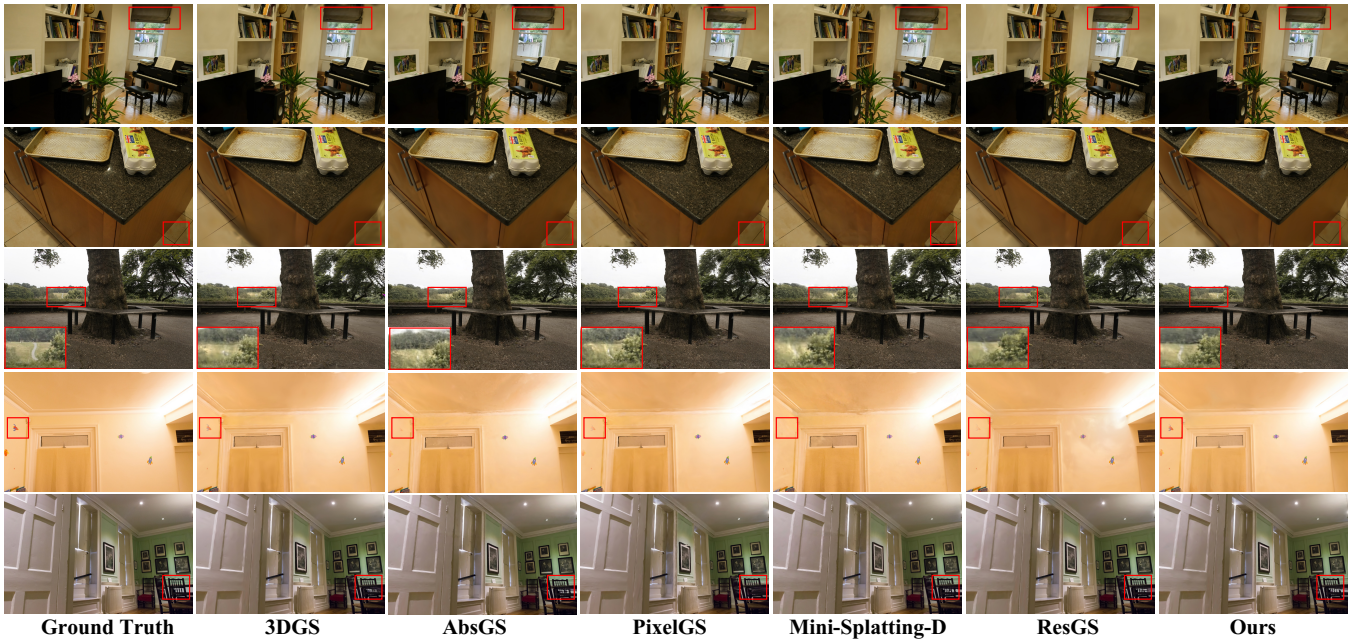


Figure 4: Qualitative comparison of novel view synthesis results. We compare the visual quality of our method (rightmost) against 3DGS, AbsGS, PixelGS, Mini-Splatting-D and ResGS on various challenging scenes. Our TileGS produces results that are closer to ground truth and better capture high-frequency details while avoiding common artifacts observed in baselines.

Quantitative and Qualitative Evaluation

As shown in Table 1, TileGS consistently outperforms prior methods in perceptual quality. By leveraging tile-wise structural cues, our method enhances both geometric fidelity and fine-scale texture reconstruction across a range of datasets. Compared to previous approaches that rely on global statistics or heuristic sampling, TileGS achieves more targeted and accurate densification, resulting in superior visual quality. For instance, it delivers state-of-the-art PSNR and LPIPS on Mip-NeRF 360 and Deep Blending, and the best SSIM and LPIPS on the geometry-challenging Tanks and Temples dataset.

In addition, we visualize the final point cloud distributions in Fig. 3. Compared to 3DGS, which tends to generate overly concentrated Gaussian clusters and redundant point clouds, our enhanced 3DGS variant promotes broader and more uniform Gaussian placement guided by tile-level quality signals, leading to improved coverage with fewer points and reduced redundancy. As a result, our method not only enhances expressiveness in key structural regions, but also avoids wasted storage on uninformative or oversampled areas.

To better understand the perceptual impact of our approach, we present visual comparisons against other approaches. Representative results are shown in Fig. 4. Our method reconstructs significantly sharper details and cleaner boundaries, especially in geometrically complex or highly textured regions. Notably, our method preserves fine-grained features such as repetitive structures, thin edges, and subtle material variations that are often smoothed out or lost in other methods. In contrast to the baseline, which tends to over-smooth or blur intricate geometry, TileGS produces visually crisp and spatially coherent reconstructions, resulting in a more faithful rendering experience under novel views. These qualitative differences are especially prominent in scenes with strong structural anisotropy or high-frequency texture.

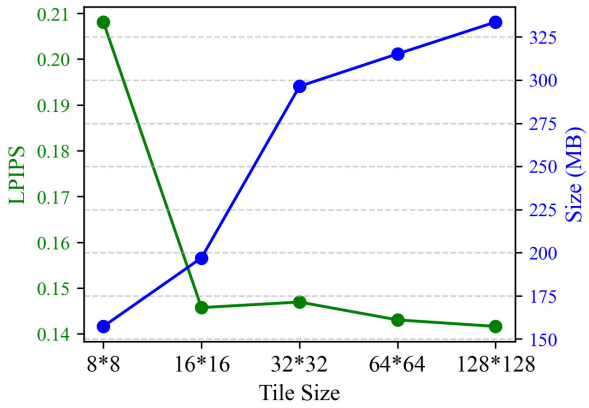


Figure 5: Effect of tile size on rendering quality and model size.

Compatibility Experiments

To demonstrate the plug-and-play nature of our approach, we integrate it into several representative baselines, including PixelGS (Zhang et al. 2024b), AbsGS (Ye et al. 2024), and mini-splatting-D (Fang and Wang 2024). As shown in

Dataset	Mip-NeRF 360				Tanks and Temples				Deep Blending			
Methods|Metrics	PSNR ↑	SSIM ↑	LPIPS ↓	Mem ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Mem ↓	PSNR ↑	SSIM ↑	LPIPS ↓	Mem ↓
3DGS	27.21	0.815	0.214	734MB	23.14	0.841	0.183	411MB	29.41	0.903	0.243	676MB
3DGS†	27.40	0.813	0.217	784MB	23.67	0.845	0.178	438MB	29.36	0.899	0.247	660MB
AbsGS†	27.43	0.819	0.198	742MB	23.62	0.852	0.162	304MB	29.63	0.901	0.236	457MB
PixelGS†	27.48	0.822	0.191	1310MB	23.83	0.853	0.151	1062MB	28.85	0.892	0.250	1075MB
Mip-Splatting†	27.97	0.831	0.175	955MB	23.68	0.859	0.1562	568MB	29.45	0.903	0.238	840MB
Mini-Splatting-D†	27.51	0.831	0.176	1100MB	23.28	0.851	0.140	1010MB	**29.82**	0.901	**0.212**	1089MB
ResGS†	28.09	**0.834**	0.177	770MB	24.20	0.865	0.145	296MB	29.73	**0.907**	0.229	582MB
Ours	**28.19**	**0.834**	**0.174**	**422MB**	**24.33**	**0.867**	**0.138**	**197MB**	29.78	0.906	0.223	**309MB**

Table 1: Quantitative comparison across datasets. Our TileGS achieves competitive or superior performance with significantly lower memory usage across all datasets. † indicates results reproduced using official code on the same hardware and training protocol for fair comparison. Best results are shown in **bold**, and second-best results are underlined in each column.

Dataset	Deep Blending		
Method|Metrics	PSNR ↑	Mem ↓	FPS ↑
PixelGS	28.85	1075	171
PixelGS + Ours	29.27	578	260
AbsGS	29.63	457	246
AbsGS + Ours	29.66	240	390
mini-splatting-D	29.82	1089	163
mini-splatting-D + Ours	29.91	623	237

Table 2: Comparison of methods with and without our optimization. We report PSNR, model size, training time and inference speed.

Table 2, our method brings consistent improvements across all settings: enhancing rendering quality, reducing memory usage, and accelerating inference. This indicates that our tile-guided strategy can be seamlessly incorporated into diverse Gaussian-based pipelines. The ability to generalize across architectures further underscores the practical value and broad applicability of our design.

Hyperparameter Analysis

Tile Size. We conducted experimental analysis and comparison on various tile sizes to investigate their impact on rendering quality and model size. Increasing tile size leads to a decrease in LPIPS, which measures perceptual loss, indicating improved rendering quality. However, this improvement comes with a trade-off, with the model size increasing significantly, as shown in Fig. 5. Although larger tile sizes offer diminishing returns in LPIPS, the model size grows substantially. Therefore, a 16×16 tile size provides the optimal balance, delivering substantial quality improvements without excessive memory overhead.

Ablation Study

To assess the effectiveness of each component in TileGS, we conduct an ablation study on the Tanks and Temples dataset. As shown in Table 3, we start from a base model and incrementally incorporate our two key contributions: Tile-Guided Densification (TGD) and Tile-Level Structural Loss (TSL).

Dataset	Tank&Temple			
Method|Metrics	PSNR ↑	SSIM ↑	LPIPS ↓	Mem ↓
Base	23.67	0.845	0.178	438MB
Base + TSL	23.74	0.847	0.176	278MB
Base + TGD	24.10	0.852	0.159	364MB
Base + TGD + TSL (full)	24.33	0.867	0.138	197MB

Table 3: Ablation study on the Tanks and Temples dataset. TGD denotes proposed Tile-Guided Densification. TSL denotes the Tile-Level Structural Loss.

Each component of **TGD** and **TSL** individually contributes to improving rendering quality and reducing model size, but they do so with different strengths. **TGD**, which emphasizes tile-guided densification, primarily improves the perceptual quality of the renderings. By focusing on areas where the model underperforms, TGD enhances fine-grained details, as shown by the increase in PSNR and the reduction in LPIPS when applied alone. On the other hand, **TSL** excels at improving structural consistency and optimizing memory usage. It reduces memory significantly by ensuring a more compact and efficient representation, while still contributing to moderate improvements in perceptual quality. The best performance, achieved when both components are included, demonstrates that the combination offers a balanced approach to improving both perceptual fidelity and computational efficiency.

Conclusion

We presented TileGS, a tile-wise perceptually guided framework for enhancing 3D Gaussian Splatting. By incorporating local rendering quality into the densification process and introducing a tile-level structural loss, TileGS enables more accurate scene representation. Our method is lightweight, plug-and-play, and demonstrates consistent improvements in both perceptual quality and structural fidelity across diverse scenes.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No. 62302297, 72192821, 62272447, 62472285, 62472282), the Fundamental Research Funds for the Central Universities (YG2023QNB17, YG2024QNA44), National Key R&D Program of China (2024YFE0115500), Young Elite Scientists Sponsorship Program by CAST (2022QNRC001), Beijing Natural Science Foundation (L222117).

References

Adamkiewicz, M.; Chen, T.; Caccavale, A.; Gardner, R.; Culbertson, P.; Bohg, J.; and Schwager, M. 2022. Vision-Only Robot Navigation in a Neural Radiance World. arXiv:2110.00168.

Aliev, K.-A.; Sevastopolsky, A.; Kolos, M.; Ulyanov, D.; and Lempitsky, V. 2020. Neural Point-Based Graphics.

Barron, J. T.; Mildenhall, B.; Verbin, D.; Srinivasan, P. P.; and Hedman, P. 2022. Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields. *CVPR*.

Barron, J. T.; Mildenhall, B.; Verbin, D.; Srinivasan, P. P.; and Hedman, P. 2023. Zip-NeRF: Anti-Aliased Grid-Based Neural Radiance Fields. *ICCV*.

Boss, M.; Braun, R.; Jampani, V.; Barron, J. T.; Liu, C.; and Lensch, H. P. 2021. NeRD: Neural Reflectance Decomposition from Image Collections. In *IEEE International Conference on Computer Vision (ICCV)*.

Chen, A.; Xu, Z.; Geiger, A.; Yu, J.; and Su, H. 2022a. TensoRF: Tensorial Radiance Fields. In *European Conference on Computer Vision (ECCV)*.

Chen, Y.; Chen, Z.; Zhang, C.; Wang, F.; Yang, X.; Wang, Y.; Cai, Z.; Yang, L.; Liu, H.; and Lin, G. 2023. GaussianEditor: Swift and Controllable 3D Editing with Gaussian Splatting. arXiv:2311.14521.

Chen, Z.; Wang, C.; Guo, Y.-C.; and Zhang, S.-H. 2022b. StructNeRF: Neural Radiance Fields for Indoor Scenes with Structural Hints. arXiv:2209.05277.

Fang, G.; and Wang, B. 2024. Mini-Splatting: Representing Scenes with a Constrained Number of Gaussians. arXiv:2403.14166.

Feng, Y.; Feng, X.; Shang, Y.; Jiang, Y.; Yu, C.; Zong, Z.; Shao, T.; Wu, H.; Zhou, K.; Jiang, C.; and Yang, Y. 2024. Gaussian Splashing: Unified Particles for Versatile Motion Synthesis and Rendering. *arXiv preprint arXiv:2401.15318*.

Hahlbohm, F.; Franke, L.; Kappel, M.; Castillo, S.; Eisemann, M.; Stamminger, M.; and Magnor, M. 2025. INPC: Implicit Neural Point Clouds for Radiance Field Rendering. In *International Conference on 3D Vision*.

Haque, A.; Tancik, M.; Efros, A.; Holynski, A.; and Kanazawa, A. 2023. Instruct-NeRF2NeRF: Editing 3D Scenes with Instructions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Hedman, P.; Philip, J.; Price, T.; Frahm, J.-M.; Drettakis, G.; and Brostow, G. 2018. Deep Blending for Free-viewpoint Image-based Rendering. 37(6): 257:1–257:15.

Jiang, Y.; Yu, C.; Xie, T.; Li, X.; Feng, Y.; Wang, H.; Li, M.; Lau, H.; Gao, F.; Yang, Y.; and Jiang, C. 2024. VR-GS: A Physical Dynamics-Aware Interactive Gaussian Splatting System in Virtual Reality. *arXiv preprint arXiv:2401.16663*.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4).

Knapitsch, A.; Park, J.; Zhou, Q.-Y.; and Koltun, V. 2017. Tanks and Temples: Benchmarking Large-Scale Scene Reconstruction. *ACM Transactions on Graphics*, 36(4).

Kwon, O.; Park, J.; and Oh, S. 2023. Renderable Neural Radiance Map for Visual Navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9099–9108.

Li, Z.; Chen, Z.; Li, Z.; and Xu, Y. 2024. Spacetime Gaussian Feature Splatting for Real-Time Dynamic View Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8508–8520.

Luiten, J.; Kopanas, G.; Leibe, B.; and Ramanan, D. 2024. Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis. In *3DV*.

Luo, G.; Xu, T.-X.; Liu, Y.-T.; Fan, X.-X.; Zhang, F.-L.; and Zhang, S.-H. 2024. 3D Gaussian Editing with A Single Image. arXiv:2408.07540.

Lyu, Y.; Cheng, K.; Kang, X.; and Chen, X. 2024. ResGS: Residual Densification of 3D Gaussian for Efficient Detail Recovery. In *International Conference on Computer Vision*.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*.

Müller, T.; Evans, A.; Schied, C.; and Keller, A. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *ACM Trans. Graph.*, 41(4): 102:1–102:15.

Rota Bulò, S.; Porzi, L.; and Kontschieder, P. 2024. Revising Densification in Gaussian Splatting. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LXIII*, 347–362. Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-031-73035-1.

Srinivasan, P. P.; Deng, B.; Zhang, X.; Tancik, M.; Mildenhall, B.; and Barron, J. T. 2021. NeRV: Neural Reflectance and Visibility Fields for Relighting and View Synthesis. In *CVPR*.

Verbin, D.; Hedman, P.; Mildenhall, B.; Zickler, T.; Barron, J. T.; and Srinivasan, P. P. 2021. Ref-NeRF: Structured View-Dependent Appearance for Neural Radiance Fields. arXiv:2112.03907.

Wang, Y.; Wang, X.; Yi, R.; Fan, Y.; Hu, J.; Zhu, J.; and Ma, L. 2025. 3D Gaussian Head Avatars with Expressive Dynamic Appearances by Compact Tensorial Representations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 21117–21126.

Wang, Z.; Bovik, A.; Sheikh, H.; and Simoncelli, E. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612.

Wen, H.; Kang, H.; Ma, J.; Huang, J.; Yang, Y.; Lin, H.; Lai, Y.-K.; and Li, K. 2025. DyCrowd: Towards Dynamic Crowd Reconstruction from a Large-scene Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–14.

Wu, G.; Yi, T.; Fang, J.; Xie, L.; Zhang, X.; Wei, W.; Liu, W.; Tian, Q.; and Wang, X. 2024. 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 20310–20320.

Xie, T.; Zong, Z.; Qiu, Y.; Li, X.; Feng, Y.; Yang, Y.; and Jiang, C. 2023. PhysGaussian: Physics-Integrated 3D Gaussians for Generative Dynamics. *arXiv preprint arXiv:2311.12198*.

Xu, Q.; Xu, Z.; Philip, J.; Bi, S.; Shu, Z.; Sunkavalli, K.; and Neumann, U. 2022. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5438–5448.

Yang, Y.; Liu, F.; Lu, Y.; Zhao, Q.; Wu, P.; Zhai, W.; Yi, R.; Cao, Y.; Ma, L.; Zha, Z.-J.; and Dong, J. 2025. SIGMAN:Scaling 3D Human Gaussian Generation with Millions of Assets. arXiv:2504.06982.

Ye, Z.; Li, W.; Liu, S.; Qiao, P.; and Dou, Y. 2024. AbsGS: Recovering Fine Details for 3D Gaussian Splatting. arXiv:2404.10484.

Yu, A.; Li, R.; Tancik, M.; Li, H.; Ng, R.; and Kanazawa, A. 2021. PlenOctrees for Real-time Rendering of Neural Radiance Fields. In *ICCV*.

Yu, H.; Julin, J.; Milacski, Z. A.; Niinuma, K.; and Jeni, L. A. 2023. CoGS: Controllable Gaussian Splatting. *arXiv*.

Yu, Z.; Chen, A.; Huang, B.; Sattler, T.; and Geiger, A. 2024. Mip-Splatting: Alias-free 3D Gaussian Splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 19447–19456.

Yuan, Y.-J.; Sun, Y.-T.; Lai, Y.-K.; Ma, Y.; Jia, R.; and Gao, L. 2022. NeRF-editing: geometry editing of neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18353–18364.

Zhang, J.; Zhan, F.; Xu, M.; Lu, S.; and Xing, E. 2024a. FreGS: 3D Gaussian Splatting with Progressive Frequency Regularization. *arXiv preprint arXiv:2403.06908*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. arXiv:1801.03924.

Zhang, Z.; Hu, W.; Lao, Y.; He, T.; and Zhao, H. 2024b. Pixel-GS: Density Control with Pixel-aware Gradient for 3D Gaussian Splatting. In *ECCV*.

Zheng, Z.; Zhao, X.; Zhang, H.; Liu, B.; and Liu, Y. 2023. AvatarReX: Real-time Expressive Full-body Avatars. arXiv:2305.04789.

Zhu, E.; Levy, M.; Gwilliam, M.; and Shrivastava, A. 2025. NeRF-Aug: Data Augmentation for Robotics with Neural Radiance Fields. arXiv:2411.02482.

Zwicker, M.; Pfister, H.; van Baar, J.; and Gross, M. 2001. EWA volume splatting. In *Proceedings Visualization, 2001. VIS '01.*, 29–538.