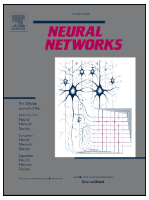
Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Full Length Article

Metamon-GS: Enhancing representability with variance-guided densification and light encoding

Junyan Su[1], Baozhu Zhao[1], Xiaohan Zhang, Qi Liu[*]

Department of Future Technology, South China University of Technology, Guangzhou, 511400, China

ARTICLE INFO

Keywords:
3D scene reconstruction
Novel view synthesis
Differentiable rendering

ABSTRACT

The introduction of 3D Gaussian Splatting (3DGS) has advanced novel view synthesis by utilizing Gaussians to represent scenes. Recent anchor-based 3DGS variants have significantly enhanced reconstruction performance by encoding Gaussian point features via anchor embeddings. Despite this progress in reconstruction, it remains challenging to further boost rendering performance. Feature embeddings have difficulty accurately representing colors from different perspectives under varying lighting conditions, which leads to a washed-out appearance. Another reason is the lack of a proper densification strategy that prevents Gaussian point growth in sparsely initialized areas, resulting in blurriness and needle-shaped artifacts. To address them, we propose Metamon-GS, from innovative viewpoints of variance-guided densification strategy and multi-level hash grid. The densification strategy guided by variance specifically targets Gaussians with high gradient variance in pixels and compensates for the importance of regions with extra Gaussians to improve reconstruction. The multi-level hash grid, by contrast, encodes implicit global lighting conditions, enabling accurate color reproduction across different viewpoints and feature embeddings. Our thorough experiments on publicly available datasets show that Metamon-GS surpasses its baseline model and other variants, delivering superior quality in rendering novel views. The source code of our method is available at https://github.com/sato-imo/metamon-gs.

1. Introduction

Advances in computer graphics and 3D vision have greatly improved the capability to create detailed 3D scenes from 2D images. This advancement is based on a long history of 3D reconstruction methods, such as Structure-from-Motion (SfM) (Snavely et al., 2006; Ye et al., 2024a) and Multi-View Stereo (MVS) (Furukawa and Ponce, 2010; Li et al., 2023; Xu et al., 2024). One of the notable advancements is 3D Gaussian Splatting (3DGS) introduced by Kerbl et al. (2023), which presents a different way to represent 3D scenes by utilizing elliptical Gaussian functions, also known as Gaussians. 3DGS extends the idea of representing 3D scenes using primitives similar to other point-based methods (Aliev et al., 2020; Gross and Pfister, 2007), which consider Gaussian functions as primitives. A Gaussian point is defined by a set of learnable features, such as spherical harmonic components, scale, rotation, position and opacity. This method allows for a smooth and continuously changing representation of the scene, making rendering more efficient and serving as a useful tool for reconstructing high-quality 3D scenes. Furthermore, a well-reconstructed point cloud can also support a variety of downstream tasks including keypoint extraction (Shao et al., 2024), 3D object detection (Ding et al., 2024; Li et al., 2025), and point cloud encryption (Yang et al., 2024a).

Challenges still exist in enhancing the quality of reconstruction under specific conditions (Kheradmand et al., 2024; Wei et al., 2024; Ye et al., 2024b), despite the strides made in creating high-quality new view images. The main problem that reduces the quality of reconstruction is the failure to densify certain areas with sparse initial point clouds sufficiently. Insufficient Gaussians cannot adequately represent these areas, causing the model to get stuck in local minima and resulting in blurred and needle-shaped artifacts.

The other challenge is that when light condition changes too sharply across different view directions, the appearance in these areas shows color degradation and loss of detail.

To address these challenges, we propose Metamon-GS. Our approach involves using a variance-guided densification technique to pinpoint areas that need more Gaussians by analyzing the variance of color gradients. This technique identifies areas that have a high variation in color but a low gradient variation in position, effectively pinpointing

[*] corresponding author.
E-mail addresses: ft_su.junyan@mail.scut.edu.cn (J. Su), 202320163293@mail.scut.edu.cn (B. Zhao), ftxiaohanzhang@mail.scut.edu.cn (X. Zhang), drliuqi@scut.edu.cn (Q. Liu).
[1] Equal Contribution

https://doi.org/10.1016/j.neunet.2025.108349
Received 10 February 2025; Received in revised form 9 November 2025; Accepted 17 November 2025
Available online 21 November 2025

Gaussians that need to be made denser. By emphasizing color differences rather than just position gradients, we can achieve better representation in areas that were previously not fully reconstructed, addressing the shortcomings of smoothing gradients across rendered pixels.

Furthermore, we also address the task of effectively interpreting color depending on various perspectives (Gao et al., 2024; Jiang et al., 2024; Shi et al., 2025; Yang et al., 2024b). Taking inspiration from Instant-NGP (Müller et al., 2022), our suggestion is to utilize a hash grid for encoding view-dependent features. We consider lighting conditions as a global attribute and incorporate directional information, originally stored in the anchor embeddings of our baseline, into the hash grid. In the MLP input, the view direction vector is replaced with the hash grid encoding of the direction vector. This approach leads to more accurate view-dependent color decoding.

We conducted extensive experiments on the Mip-NeRF 360 (Barron et al., 2022), NeRF Synthetic (Mildenhall et al., 2020), and Tanks & Temples (Knapitsch et al., 2017) datasets, aiming to showcase the advantages of our model over the baseline models. We also performed ablation studies to validate the effectiveness of our proposed methods. Here are our contributions:

- We propose to use a hash grid to encode lighting conditions, which enhances the quality of reconstruction in scenes with intricate lighting.
- We propose a novel densification strategy guided by variance of pixel gradients to address problems arising from the gradient smoothing of rendered pixels. This method is primarily implemented with CUDA within the part of code of Gaussian rasterizer.
- Experiments on various datasets show that our approach successfully tackles these challenges and outperforms the baseline model.

2. Related work

2.1. Neural radiance field

Neural Radiance Fields (NeRF) represent a revolutionary technique that has demonstrated exceptional performance in novel view synthesis tasks (Mildenhall et al., 2020; Verbin et al., 2022). NeRF employs Multi-Layer Perceptrons (MLPs), to implicitly represent 3D scenes. By estimating a radiance field and utilizing volumetric rendering (Drebin et al., 1988; Levoy, 1990), NeRF can generate high-quality images from new viewpoints.

In recent years, NeRF has achieved significant advancements in multi-scale anti-aliasing and efficient scene representations, further enhancing rendering quality and practicality. For multi-scale anti-aliasing, studies such as Barron et al. (2021, 2022, 2023), Zhang et al. (2020) have introduce novel ray sampling strategies and multi-scale feature field construction to overcome certain artifacts. Meanwhile, works like (Chen et al., 2022; Fridovich-Keil et al., 2023, 2022; Liu et al., 2020) leverage tensor factorization and hierarchical sparse voxel structures to achieve improvements in training efficiency while preserving reconstruction fidelity. These advancements further advance NeRF models' rendering quality and practicality significantly. However, NeRF and its variants highly rely on ray marching and sampling, as well as the use of MLPs to estimate color and opacity, resulting in slow training and inference speeds. To address these issues, researchers have proposed various optimization methods (Chen et al., 2023; Hedman et al., 2021; Müller et al., 2022; Sun et al., 2022) aiming to improve the training and inference efficiency while maintaining rendering quality.

2.2. 3D Gaussian splatting and variants

Recently, the field of novel view synthesis has witnessed significant advancements, with 3DGS emerging as a particularly promising technique. Unlike traditional point-based methods (Guo et al., 2024; Insafutdinov and Dosovitskiy, 2018; Kopanas et al., 2021; Lassner and Zollhofer, 2021; Sandström et al., 2023; Yifan et al., 2019; Zhang et al., 2022), 3DGS represents scene elements using ellipsoids defined by Gaussian functions, encapsulating both shape and color information. These ellipsoids, referred to as Gaussians, are characterized by a set of learnable features including spherical harmonic components, scale, rotation, position, and opacity. The 3DGS pipeline typically begins with the initialization of Gaussians, derived from sparse point clouds estimated by Structure-from-Motion (SfM) methods like COLMAP (Schönberger and Frahm, 2016). During training, an adaptive densification mechanism is employed to split or clone these Gaussians, allowing for a more detailed scene representation. This process helps generate more Gaussians to model finer details in the scene, enhancing the quality of the synthesized views. As a result, 3DGS provides faster rendering speeds and produces higher-quality results in novel view synthesis.

Building upon the success of standard 3DGS, researchers have continued to further enhance its capabilities. One notable advancement was the development of anchor-based lightweight variants, such as Scaffold-GS (Lu et al., 2024) and Octree-GS (Ren et al., 2024). For an anchor restricted inside a voxel, there are multiple offset Gaussians associated with it, with the features of individual Gaussians being embedded into one feature embedding. The densification of Gaussians in these methods is equivalent to the densification of anchors, which leverages the sparsity of local features of Gaussians, reducing the number of trainable parameters by decoding the feature embeddings of offset Gaussians with several MLPs.

2.3. Densification strategy

For point-based methods, the initial point cloud often lacks sufficient points to fully model complex scenes, thus a densification strategy is needed to generate additional points. A probable situation is that the initial point cloud is imperfect, which presents the need for a proper densification strategy.

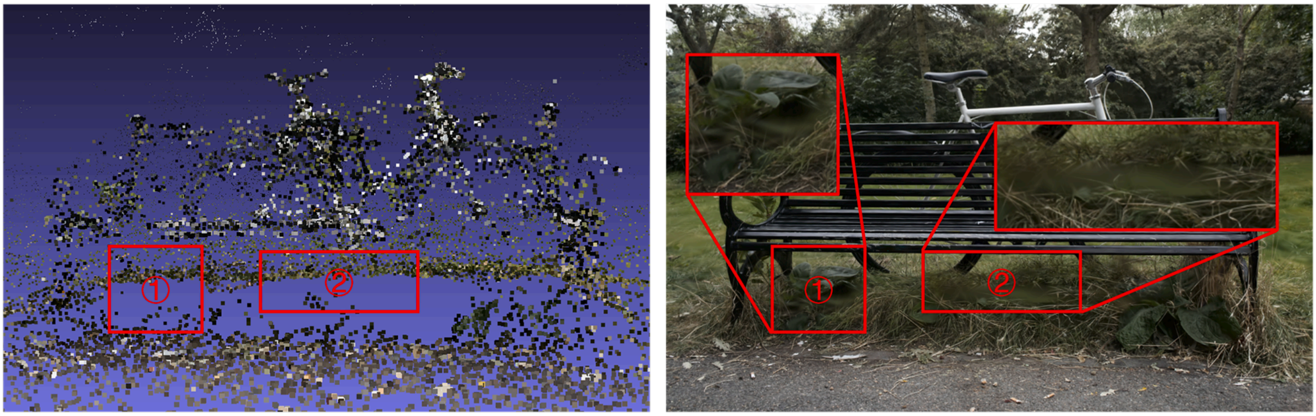
3DGS (Kerbl et al., 2023) was designed with an adaptive densification strategy utilizing the gradient from the position of points in the NDC coordinate. This process involves splitting or cloning operations on target Gaussians, enabling more effective modeling of fine features and significantly improving the fidelity of reconstructed scenes. Effective though, this strategy still struggles to densify Gaussians in areas with intricate textures, where SfM methods produce insufficient initial points, as illustrated in Fig. 1. Several studies have focused on improving densification strategies. For instance, FreGS (Zhang et al., 2024a) enhanced the gradient magnitude in high-frequency areas by incorporating frequency domain supervision. This allows for more opportunities for Gaussians to be densified in areas with intricate details but may struggle with low-contrast regions. On the other hand, Pixel-GS (Zhang et al., 2024b) improved upon the original approach by adjusting the average gradient based on the number of pixels rendered by each Gaussian point. This approach prioritizes larger Gaussians for densification, improving overall scene coverage, but potentially overlooking smaller, yet significant features.

Our Metamon-GS approaches the concept of reconstruction differently, using pixel color gradients for densification. The method we offer can be used in conjunction with current methods, potentially creating new possibilities for future research on adaptive densification techniques.

2.4. View-dependent color

The appearance of a 3D scene can vary significantly depending on the viewing angle and lighting conditions, particularly due to surface properties like roughness. This phenomenon of view-dependent color poses a significant challenge in 3D scene reconstruction and rendering. Recent advancements in neural rendering and point-based reconstruction have made substantial progress in addressing this issue.

NeRF (Mildenhall et al., 2020) incorporated camera orientation as an input to its MLP to generate view-dependent colors. This approach,



(a) Initial Sparse Point Cloud **(b) Blurring Novel View Rendering**

Fig. 1. In certain areas, the Colmap-generated SfM point cloud is relatively sparse, as indicated by the red box in **(a)**. Utilizing this point cloud as the starting Gaussians and implementing the original clone and split density control strategy may lead to certain areas lacking enough Gaussians, ultimately causing an inadequate scene reconstruction, as demonstrated in the corresponding region highlighted by the red box in **(b)**. This greatly reduces the overall quality of novel view synthesis, especially in areas with intricate geometry or delicate details.

while effective, can be computationally intensive and may struggle with large-scale scenes. Several Point-based methods (Fridovich-Keil et al., 2022; Kerbl et al., 2023; Li et al., 2024; Rakhimov et al., 2022; Sloan et al., 2023) utilized spherical harmonics as point features to represent view-dependent colors efficiently. Spherical harmonics provide a compact representation of directional data, allowing these methods to capture color variations across viewing angles with relatively low computational overhead. Anchor-based variants of 3DGS such as Scaffold-GS (Lu et al., 2024) and OctreeGS (Ren et al., 2024) used feature embedding of an anchor instead of explicit features to describe Gaussians. In these methods, color is decoded by an MLP with feature embedding and view direction as input. Effective to some extent though, the use of feature embedding and MLPs for encoding view-dependent colors has shown limitations in capturing the complexity of lighting conditions.

To address this, our Metamon-GS develops a novel approach inspired by Instant-NGP (Müller et al., 2022), utilizing a hash grid for encoding view-dependent features. We treat lighting conditions, which are previously stored in anchor embeddings, as a global attribute and incorporate directional information into the hash grid. The view direction vector in the MLP input is substituted with the hash grid encoding of the direction vector within our approach. Our method improves view-dependent color decoding accuracy by incorporating lighting conditions into a hash grid.

3. Method

Here, we propose Metamon-GS to address the aforementioned limitations, where Fig. 2 provides an overview of our approach. We first briefly review the original 3DGS densification strategy, conducting a pre-experiment to analyze the mean and variance of the color gradient of Gaussians. Then we introduce our Variance-Guided Densification strategy, explaining how it leverages the variance of color gradients. Finally, we present our Lighting Hash Encoder, which employs a hash grid to encode lighting information, enabling more accurate modeling of complex lighting conditions compared to original view direction inputs.

3.1. *Densification in 3D Gaussian splatting*

The original Gaussian densification strategy assumes that when a Gaussian point cannot fit the rendered pixels well, the point tends to shift its position due to a high backpropagated gradient on the Normalized Device Coordinates (NDC) position. The decision to densify the Gaussian point depends on the average gradient magnitude across visible viewpoints during the densification period. This strategy can be described as:

$$\bar{g}_{\text{norm}} = \frac{\sum_{k=1}^{M}\sqrt{\left(\frac{\partial L_k}{\partial x_k}\right)^2+\left(\frac{\partial L_k}{\partial y_k}\right)^2}}{M} \qquad \text{where } \frac{\partial L_k}{\partial x_k} = \sum_{p} G_{p,k} \tag{1}$$

where M is the number of viewpoints, $G_{p,k}$ is the gradient from the p th pixel the Gaussian k rendered and (x, y) is the NDC coordinate of the point. A Gaussian is selected to split and clone when its $\bar{g}_{\text{norm}}$ exceeds the threshold τ_{th}.

This approach is effective in many cases, but it does not effectively concentrate when the backpropagated gradients from various pixels within the Gaussian's covered area evenly affect its position in different directions. The resulting gradient from adding up these individual pixel gradients is quite small to reach the threshold, preventing the Gaussian point from splitting.

We illustrate these two instances of under-reconstructed that occur during the implementation of adaptive Gaussian densification using positional gradient. In the first scenario, as depicted in Fig. 3(a), the pixel gradients converge coherently, leading to a significant Gaussian positional gradient that facilitates densification. However, as illustrated in Fig. 3(b), this assumption that under-reconstructed Gaussians exhibit high gradients fails to hold in certain scenarios. The gradients diverge, causing the Gaussian positions to be optimized in different directions. This is because the Gaussian covers a specific area with intricate textures. The divergence of gradients causes a decrease in the overall positional gradient following the weighted sum, preventing the Gaussian from reaching the necessary threshold for proper densification. The quality of rendering descends when there are no enough representative Gaussians.

To validate our hypothesis, we conduct experiments by adjusting the CUDA differential rasterization pipeline. We compute the mean and variance of the color gradients of Gaussians to track how these statistics change during training process. Results are shown in Fig. 4. The findings indicate that many Gaussians still exhibit diverse color gradients, even when the training process reaches 15,000 steps, suggesting that the rendered pixels do not fit well.

This analysis highlights the difficulties presented by intricate textures, as the inconsistency in pixel gradients makes it difficult to densify and fit effectively. This observation also intuitively matches the characteristics of areas that have not been fully reconstructed as perceived by humans. Although current densification methods prioritize improving gradients in high-frequency or inadequately fitted regions to facilitate

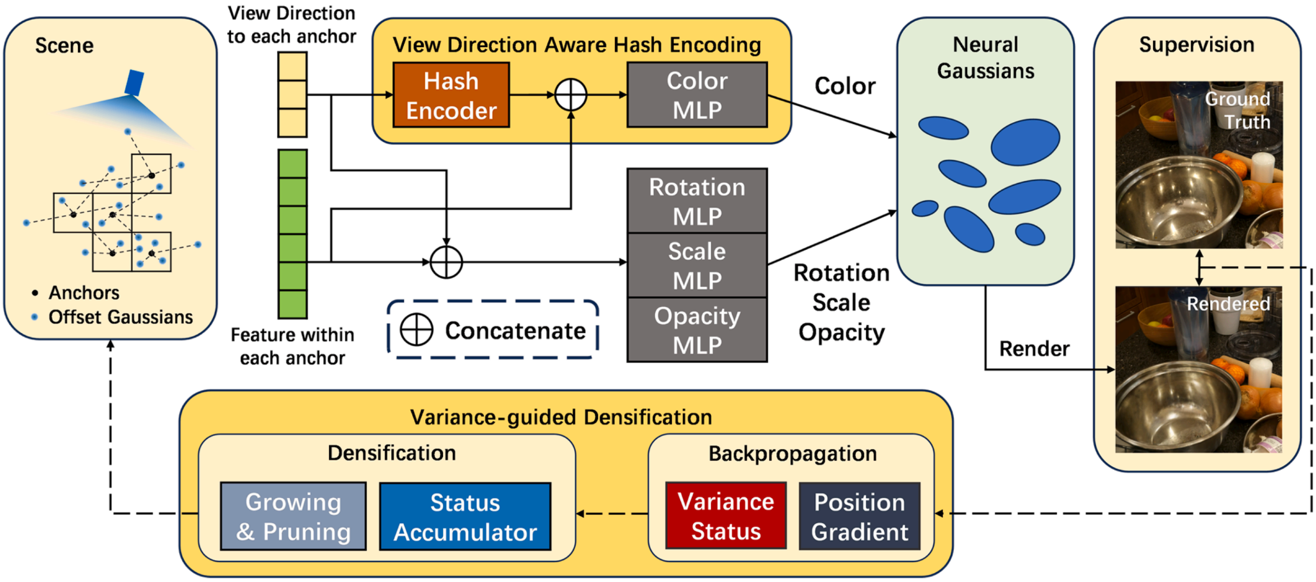


Fig. 2. Overview of Metamon-GS. Our method enhances 3DGS with two key innovations: (1) Integration of view-direction aware hash encoding to learn view-dependent features, e.g., lighting condition, and (2) A variance-guided densification strategy based on variance of color gradient during backpropagation. We first interpolate view-dependent feature embedding from a hash grid, concatenate them with anchor embeddings, and then feed them into the color MLP. Other features are decoded similarly to Scaffold-GS. This strategy, combined with our novel densification strategy, results in more accurate color representation and efficient Gaussians for scene representation.

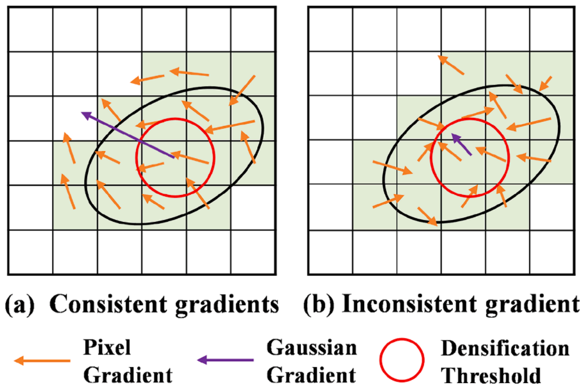


Fig. 3. Pixel-wise gradients and overall gradients of Gaussians under two different cases of under-reconstruction. (a) In an ideal scenario, gradients of pixels consistently converge, forming a high Gaussian positional gradient, which allows effective densification. (b) In contrast, in a scenario with complex textures, gradients diverge in different directions, resulting in a lower Gaussian positional gradient, hindering proper densification.

the satisfaction of densification criteria, there is still a lack of statistical approach to tackling this problem.

3.2. *Variance-guided densification*

Conventional densification in 3DGS relies on positional gradient magnitude to trigger Gaussian splitting or cloning. However, this approach faces limitations in regions with intricate textures or complex color patterns. In such areas, although individual pixels may exhibit high color variance, the positional gradients tend to average out across variant pixels, thereby masking the underlying need for densification and potentially leaving textured regions under-represented.

To address this limitation, we propose a Variance-Guided Densification (VGD) approach that directly monitors color gradient variations during the rendering process. Our key insight is that regions requiring densification often exhibit higher variance in pixels' color gradients, even when their averaged positional gradients appear relatively low. By tracking this variance, we can better identify areas where a single Gaussian point inadequately represents the local color distribution complexity.

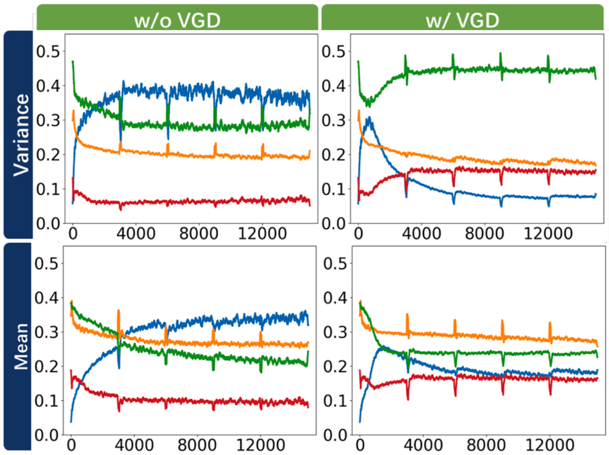
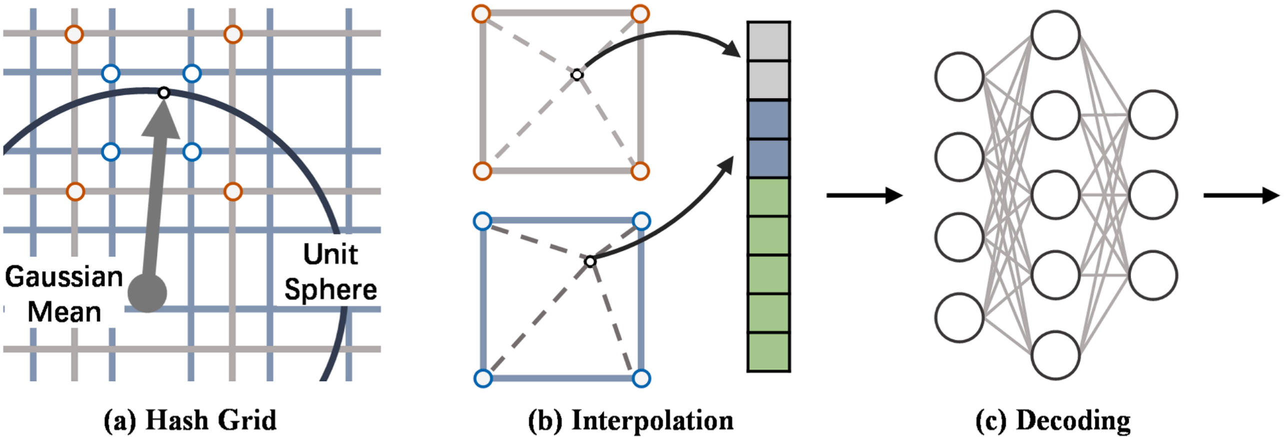


Fig. 4. Pre-experiment on 3D-GS with our proposed VGD strategy. We employ an identical starting point for the reconstruction process. After 600 iterations, the curves for 3D-GS with VGD begin to disperse significantly from those without VGD. For 3D-GS w/o VGD, the curves of high gradient variance (blue and yellow) remain relatively high as the densification strategy progresses. For 3D-GS without VGD, the high gradient variance curves (blue and yellow) remain relatively elevated as the densification strategy progresses. In contrast, for 3D-GS with VGD, the corresponding high gradient variance curves decrease rapidly, while the low gradient variance curves (green and red) show an increase. These results show that our suggested method of densification successfully decreases color discrepancies among pixels in Gaussians.

3.2.1. *Variance computation during backpropagation*

Our method computes the mean and variance of pixel gradients for each Gaussian point during the rasterization backpropagation process. We employ an iterative updating scheme that efficiently accumulates



(a) Hash Grid (b) Interpolation (c) Decoding

Fig. 5. Illustration of Lighting Hash Encoder projected into 2D. (a) We index the hash grid by Distant Gaussians are normalized and projected onto a unit sphere. The vertices of the hash grid cells intersected by this unit sphere represent the hash grid parameters that will be optimized during training. This allows for the effective encoding of view-dependent lighting information.

statistics across all pixels rendered by each Gaussian:

$$\hat{\mu}_{n+1} = \hat{\mu}_n + \beta_{n+1}(g_{n+1} - \hat{\mu}_n) \tag{2}$$

$$\hat{\sigma}^2_{n+1} = (1 - \beta_n)\hat{\sigma}^2_n + \beta_{n+1}(g_{n+1} - \hat{\mu}_n)^2 \tag{3}$$

Here, n denotes the index of the pixel rendered by the Gaussian, $\beta_n = 1/n$ provides the incremental weighting, and g_n represents the backpropagated gradient of the n-th pixel. We perform these calculations separately for each RGB channel to capture color-specific variations, then compute the total variance by summing across all channels to obtain a comprehensive measure of the Gaussian's metric for densification.

Our appraoch leverages the existing CUDA rasterizer implementation, computing gradient statistics during the standard backpropagation pass with minimal computational overhead. This design ensures that variance tracking integrates naturally into the 3DGS pipeline without significant performance impact.

3.2.2. *Integration with adaptive densification*

To seamlessly integrate our variance-based criterion with the existing densification framework, we combine it with the original positional gradient threshold through a scaled comparison. The final densification criterion becomes:

$$\gamma \bar{D} + \bar{g}_{\text{norm}} > \tau\text{th} \tag{4}$$

where the average variance $\bar{D}$ is computed across multiple views:

$$\bar{D} = \frac{\sum_{k=1}^{M} \hat{\sigma}^2}{M} \tag{5}$$

Here, M represents the number of views rendered within the densification interval, and γ is a scaling factor that balances the contribution of variance-based and gradient-based criteria.

3.3. *Lighting hash encoder*

Existing neural rendering approaches face significant challenges in modeling complex view-dependent appearance effects. Point-based methods, such as 3DGS, employ spherical harmonics to represent color variations across viewing directions, effectively capturing various photometric effect.

Anchor-based methods like Scaffold-GS decoding implicit feature embeddings alongside viewing directions to generate view-dependent colors for attached Gaussians. While the constrained dimensionality of anchor feature embeddings fundamentally limits the model's ability to accurately represent complex lighting conditions across diverse viewpoints.

Table 1
Comparison with different novel view synthesis methods across three public datasets (Mip-NeRF 360 Barron et al., 2022, Tanks&Temples Knapitsch et al., 2017, and Deep Blending Hedman et al., 2018). Results are presented in SSIM (Wang et al., 2004), PSNR, and LPIPS (Zhang et al., 2018).

Method	Mip-NeRF360			Tanks&Temples			Deep Blending		
	SSIM↑	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓
INGP-Base	0.671	25.30	0.371	0.723	21.72	0.330	0.797	23.62	0.423
INGP-Big	0.699	25.59	0.331	0.745	21.92	0.305	0.817	24.96	0.390
M-NeRF 360	0.792	27.69	0.237	0.759	22.22	0.257	0.901	29.40	0.245
3DGS	0.870	29.07	0.184	0.841	23.14	0.183	0.881	28.92	0.287
Scaffold	0.848	28.84	0.220	**0.853**	23.96	0.177	0.906	30.21	0.254
Ours	**0.876**	**29.52**	**0.171**	0.850	**24.03**	**0.176**	**0.912**	**30.40**	**0.230**

Table 2
Ablation studies on our proposed light encoding and densification strategy. Variance-guided densification adds sufficient Gaussians to better reconstruct the scene and Hash Grid Encoded Lighting enables better color representation considering different view directions.

Method	SSIM↑	PSNR↑	LPIPS↓
Base	0.848	28.84	0.220
Base + LHE	0.870	29.34	0.187
Base + LHE + VGD	**0.876**	**29.52**	**0.171**

To address these limitations, We propose leveraging hash grid encoding to efficiently learn scene lighting information as illustrated in Fig. 5. Hash grids provide a compact yet expressive representation by mapping complex spatial data into hash tables, enabling efficient storage and rapid querying of lighting parameters.

3.3.1. *Implementation of hash grid*

Our hash grid follows a computational approach similar to other works, consisting of three steps:

1) **View Direction Computation:** For each Gaussian anchor point i, we compute the normalized view direction vector as:

$$\mathbf{v} = \frac{(X_{cam} - X_{anchor})}{\|X_{cam} - X_{anchor}\|} \tag{6}$$

where $\mathbf{X}cam$ represents the camera position and $\mathbf{X}anchor, i$ denotes the anchor position. The resulting vector $\mathbf{v}_i$ lies on the unit sphere ($\mathbf{v}_i \in [-1, 1]^3, |\mathbf{v}_i| = 1$), defining the domain of our lighting function.

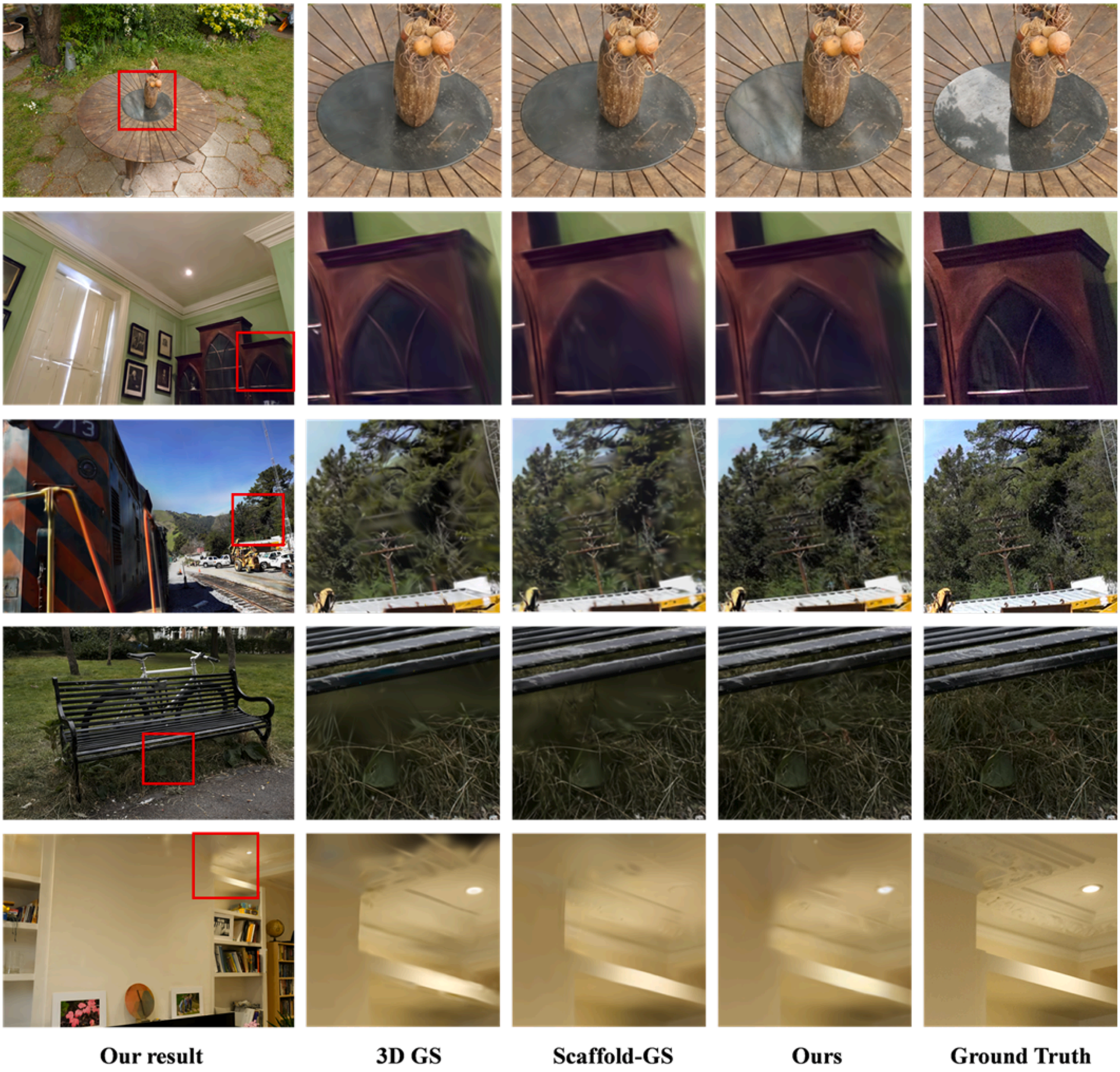


Fig. 6. Comparison on multiple datasets. We conducted extensive experiments on diverse datasets including indoor, outdoor, and object-centric datasets. Results show that our method outperforms the predecessor, providing better rendering effects for texture details.

2) **Spatial Hashing:** We discretize the view direction space and apply a hash function to convert continuous vectors into discrete indices:

$$h(\mathbf{x}) = \left(\left\lfloor \frac{x}{s_x} \right\rfloor, \left\lfloor \frac{y}{s_y} \right\rfloor, \left\lfloor \frac{z}{s_z} \right\rfloor \right) \bmod M \tag{7}$$

where s_x, s_y, and s_z are the minimal resolutions of current level and M is the size of the hash table. These indices can quickly locate relevant cells in the hash grid.

3) **Trilinear Interpolation:** To obtain smooth lighting values at arbitrary view directions, we employ trilinear interpolation among the eight nearest grid vertices. This ensures continuous color transitions and eliminates discretization artifacts.

Once the interpolated embeddings are generated, they are concatenated with the anchor embeddings and fed into the decoder MLP to produce the final Gaussian features.

3.3.2. Regularization

Although the hash grid is effective for encoding light conditions, it tends to overfit the training views. Overfitting can result in inadequate generalization when used on test samples, which can reduce the model's reliability and precision. We add random noise to the view direction vector to ensure the generalization of our model. This helps to achieve a more seamless and consistent color response within the actual viewing directions, thus mitigating the risk of overfitting and promoting better performance on test data.

4. Experiments

4.1. Experimental setup

Dataset We evaluate our method on a diverse range of scenes from three primary datasets: Mip-NeRF 360 (Barron et al., 2022), Tanks & Temples (Knapitsch et al., 2017). and DeepBlending (Hedman et al.,

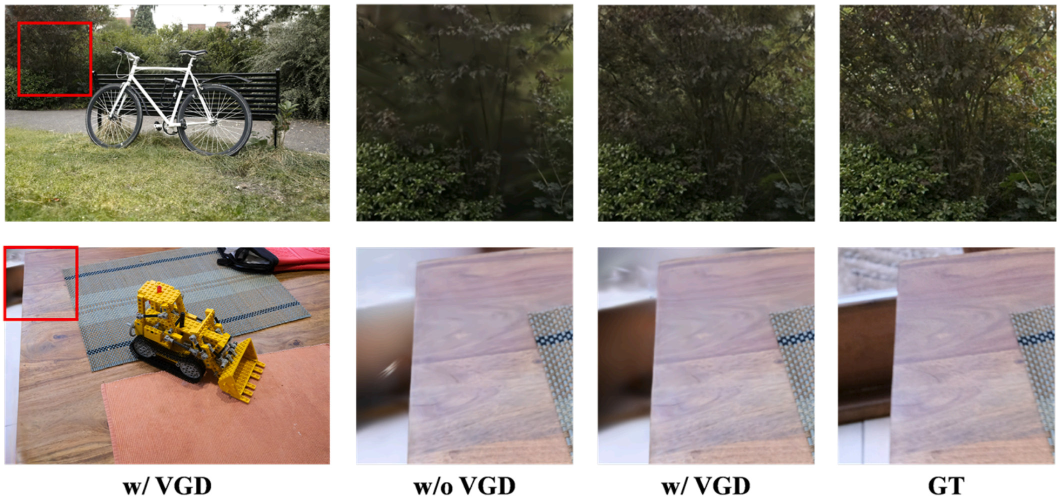


Fig. 7. Qualitative results on variance guided densification. After incorporating our proposed VGD method, most of the previously blurred plant details in the background and the lower edge of the door frame have been reconstructed and restored. Our method significantly suppresses visually prominent reconstruction artifacts, thereby improving overall image quality and visual fidelity.

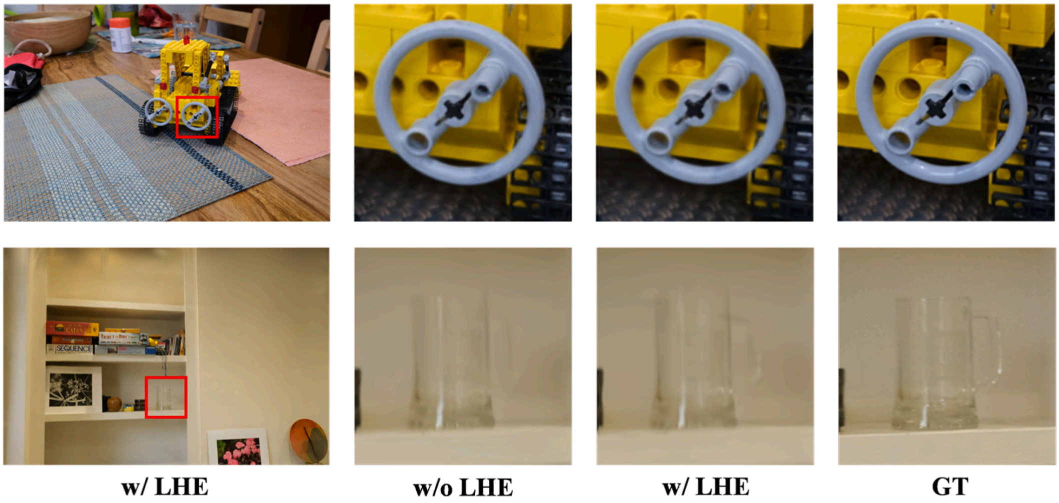


Fig. 8. Qualitative results on light hash encoder. The incorporation of LHE significantly enhances the visual quality of the LEGO construction vehicle model. Specifically, the glossiness rendering on the upper and lower surfaces of the handwheel appears more realistic and natural. Regarding color reproduction, our method exhibits superior accuracy with reduced color deviation. For instance, the gray color of the wheel rim is rendered as a realistic light gray rather than a dark gray in our approach, while the yellow building blocks in the background display a true bright yellow tone instead of an orange-tinted yellow, demonstrating improved color fidelity.

2018). This comprehensive selection includes both real-world and synthetic environments, encompassing indoor and outdoor scenes with varying levels of complexity. The MipNeRF-360 dataset provides seven real-world scenes with challenging view distributions, while NeRF Synthetic offers controlled synthetic scenes ideal for baseline comparisons. Tanks & Temples contributes large-scale real-world scenes with intricate geometries, and DeepBlending adds diversity with its unique capture methodology. These varied datasets allow us to test the generalization ability of our method across different scene types and capture conditions. Following the protocol established in Mip-NeRF360 (Barron et al., 2022), we designate every eighth image as part of the test set, utilizing the remaining images for training, ensuring a consistent and fair evaluation across all methods.

Metrics To quantitatively assess the performance of our novel view synthesis method, we employ three complementary metrics: SSIM (Wang et al., 2004), PSNR, and LPIPS (Zhang et al., 2018). SSIM evaluates the structural and perceptual similarity between the synthesized views and ground truth, capturing local patterns of pixel intensities. PSNR provides a standard measure of reconstruction quality, particularly sensitive to per-pixel differences. LPIPS offers a perceptual metric that aligns well with human visual perception, evaluating differences in feature space rather than pixel space. Together, these metrics provide a comprehensive evaluation of our method's ability to generate high-quality, perceptually accurate novel views.

Implementation We conducted all experiments on an RTX 3090 GPU, maintaining consistent hyperparameters across all scenes to demonstrate the robustness and generalizability of our method. Our experimental setup encompasses both quantitative and qualitative comparisons with existing state-of-the-art novel view synthesis methods, as well as detailed ablation studies on our key technical components. These studies aim to validate the effectiveness of each component in our proposed Metamon-GS. For modeling lighting, we use an 8-level hash grid with a base resolution of 8, the maximum hash map size is 19. The coefficient γ we adopted is 2^{11}. We set τ_{th} to 0.0004 following our baseline model.

4.2. Results

In our comparative experiments, we evaluated our approach against several leading methods across 11 real-world scenes sourced from three aforementioned datasets, providing a comprehensive benchmark of our method's capabilities in diverse real-world scenarios. We compare our model with 4 predecessors, including Mip-NeRF 360, Instant-NGP (Müller et al., 2022), 3DGS (Kerbl et al., 2023), and Scaffold-GS (Lu et al., 2024). As shown in Table 1, our method outperforms these predecessor approaches. The results demonstrate that our method surpasses the state-of-the-art model by 0.45 dB in PSNR on Mip-NeRF360 dataset.

Rendering results presented in Fig. 6 clearly demonstrate that our model exhibits remarkable perception capabilities for environmental light changes, such as the reflections of the environment on the surfaces of table and cabinet, as well as for complex textures, like the sharp edges on leaves and grassland. Our model shows a strong capability in modeling intricate details, and its scene reconstruction ability has significantly surpassed that of the baseline and other advanced models.

4.3. Ablation study

We conducted an ablation study to evaluate the effectiveness of the separate parts of our proposed method on the Mip-NeRF 360 (Barron et al., 2022) dataset. Mip-NeRF 360 is a widely-used dataset in the field of 3D reconstruction, featuring diverse scenes and complex textures, which provides a challenging environment to test the robustness and effectiveness of our method.

We evaluated the variance-guided densification strategy and found that it effectively facilitated the Gaussians to densify in areas with complex textures. In complex-textured regions, the variance of pixel color gradients is typically high. By analyzing this variance, our strategy can precisely identify these areas and allocate more Gaussians, enhancing the overall fitting to the scene. This improvement also led to an increase in rendering quality. Rendering results are demonstrated in Fig. 7.

Results are presented in Table 2. In terms of evaluation metrics, the proposed LHE achieved an improvement in PSNR by 0.18 dB, and a reduction in LPIPS by 0.016. Qualitative result is shown in Fig. 8. These improvements can be attributed to the unique design of LHE. By using a hash grid to encode lighting information and considering lighting conditions as a global attribute with directional information, LHE can better capture the complex lighting variations in the scene. This enables more accurate color representation under different view directions, thus contributing to the overall improvement in rendering quality.

Overall, our ablation study demonstrates the effectiveness of both the variance-guided densification strategy and the Lighting Hash Encoder in enhancing the performance of our proposed method on the challenging Mip-NeRF 360 dataset.

5. Conclusion

We have introduced a new method for identifying Gaussians that need to be densified, which is based on the variance of color gradients in pixels generated. This method corresponds to how humans perceive blurriness in areas that are not well-fitted. Furthermore, a view-dependent hash grid feature is implemented to substitute the view direction vector input of the color MLP, reducing the uncertainty in modeling intricate lighting for Gaussian anchors. The results of the experiment show that our new method is effective, outperforming other methods in novel view synthesis tasks. Our model generates high-quality images with enhanced detail and fewer defects when compared to current methods.

However, our approach exhibits some limitations. The view-dependent modeling remains sensitive to extreme viewpoint variations beyond the training distribution, occasionally causing color inconsistency in under-observed regions. Additionally, the densification criteria require careful parameter tuning for scenes with complex multi-scale structures, as aggressive densification could accidentally lead to unnecessary Gaussian growth in textureless regions. Furthermore, with the modification of MLP and the introduction of hash grid in LHE, the original SIBR_viewer is no longer compatible with Metamon-GS. To visualize the reconstruction, a dedicated version of the viewer still needs to be developed.

Future work will focus on two directions: first, developing a geometry-aware optimization to improve robustness against extreme perspective variations, and second, designing self-adaptive densification criteria capable of automatically adjusting to local structural complexity. These advancements will further strengthen the practicality of Gaussian splatting in real-world 3D Reconstruction.

CRediT authorship contribution statement

Junyan Su: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation, Conceptualization; **Baozhu Zhao:** Writing – review & editing, Visualization, Validation, Supervision; **Xiaohan Zhang:** Writing – review & editing, Visualization, Supervision; **Qi Liu:** Writing – review & editing, Supervision, Resources, Project administration, Investigation, Funding acquisition.

Declaration of competing interest

This work was supported in part by the National Natural Science Foundation of China under Grant 62202174, in part by the GJYC program of Guangzhou under Grant 2024D01J0081, and in part by the ZJ program of Guangdong under Grant 2023QN10X455, and in part by the Fundamental Research Funds for the Central Universities under Grant 2025ZYGXZR053.

References

Aliev, K.-A., Sevastopolsky, A., Kolos, M., Ulyanov, D., & Lempitsky, V. (2020). Neural point-based graphics. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer vision – ECCV 2020* (pp. 696–712). Cham: Springer International Publishing.

Barron, J. T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., & Srinivasan, P. P. (2021). Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 5855–5864).

Barron, J. T., Mildenhall, B., Verbin, D., Srinivasan, P. P., & Hedman, P. (2022). Mip-neRF 360: Unbounded anti-aliased neural radiance fields. *CVPR*, .

Barron, J. T., Mildenhall, B., Verbin, D., Srinivasan, P. P., & Hedman, P. (2023). Zip-nerf: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 19697–19705).

Chen, A., Xu, Z., Geiger, A., Yu, J., & Su, H. (2022). TensoRF: Tensorial radiance fields. In *European conference on computer vision (ECCV)*.

Chen, Z., Funkhouser, T., Hedman, P., & Tagliasacchi, A. (2023). MobileneRF: Exploiting the polygon rasterization pipeline for efficient neural field rendering on mobile architectures. In *The conference on computer vision and pattern recognition (CVPR)*.

Ding, B., Xie, J., Nie, J., Wu, Y., & Cao, J. (2024). C2BG-Net: Cross-modality and cross-scale balance network with global semantics for multi-modal 3d object detection. *Neural Networks*, *179*, 106535. https://doi.org/https://doi.org/10.1016/j.neunet.2024.106535

Drebin, R. A., Carpenter, L., & Hanrahan, P. (1988). Volume rendering. *ACM Siggraph Computer Graphics*, *22*(4), 65–74.

Fridovich-Keil, S., Meanti, G., Warburg, F. R., Recht, B., & Kanazawa, A. (2023). K-Planes: Explicit radiance fields in space, time, and appearance. In *Cvpr*.

Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., & Kanazawa, A. (2022). Plenoxels: Radiance fields without neural networks. In *Cvpr*.

Furukawa, Y., & Ponce, J. (2010). Accurate, dense and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(8), 1362–1376. https://doi.org/10.1109/TPAMI.2009.161

Gao, J., Gu, C., Lin, Y., Li, Z., Zhu, H., Cao, X., Zhang, L., & Yao, Y. (2024). Relightable 3d gaussians: Realistic point cloud relighting with brdf decomposition and ray tracing. In *European conference on computer vision* (pp. 73–89). Springer.

Gross, M., & Pfister, H. (2007). Point-Based Graphics. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Guo, S., Wang, Q., Gao, Y., Xie, R., Li, L., Zhu, F., & Song, L. (2024). Depth-guided robust point cloud fusion neRF for sparse input views. *IEEE Transactions on Circuits and Sys-*

tems for Video Technology, *34*(9), 8093–8106. https://doi.org/10.1109/TCSVT.2024.3385360

Hedman, P., Philip, J., Price, T., Frahm, J.-M., Drettakis, G., & Brostow, G. (2018). Deep blending for free-viewpoint image-based rendering, . *ACM Transactions on Graphics*, *37*(6), 257:1–257:15.

Hedman, P., Srinivasan, P. P., Mildenhall, B., Barron, J. T., & Debevec, P. (2021). Baking neural radiance fields for real-time view synthesis. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)* (pp. 5875–5884).

Insafutdinov, E., & Dosovitskiy, A. (2018). Unsupervised learning of shape and pose with differentiable point clouds. *Advances in neural information processing systems*, *31*, 2802–2812.

Jiang, Y., Tu, J., Liu, Y., Gao, X., Long, X., Wang, W., & Ma, Y. (2024). Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5322–5332).

Kerbl, B., Kopanas, G., Leimkühler, T., & Drettakis, G. (2023). 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, *42*(4). https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

Kheradmand, S., Rebain, D., Sharma, G., Sun, W., Tseng, Y.-C., Isack, H., Kar, A., Tagliasacchi, A., & Yi, K. M. (2024). 3D gaussian splatting as markov chain monte carlo. *Advances in Neural Information Processing Systems*, *37*, 80965–80986.

Knapitsch, A., Park, J., Zhou, Q.-Y., & Koltun, V. (2017). Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, *36*(4).

Kopanas, G., Philip, J., Leimkühler, T., & Drettakis, G. (2021). Point-based neural rendering with per-view optimization. In *Computer graphics forum* (pp. 29–43). Wiley Online Library (*vol. 40*).

Lassner, C., & Zollhofer, M. (2021). Pulsar: Efficient sphere-based neural rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 1440–1449).

Levoy, M. (1990). Efficient ray tracing of volume data. *ACM Transactions on Graphics (ToG)*, *9*(3), 245–261.

Li, C., Zhou, L., Jiang, H., Zhang, Z., Xiang, X., Sun, H., Luan, Q., Bao, H., & Zhang, G. (2023). Hybrid-MVS: Robust multi-view reconstruction with hybrid optimization of visual and depth cues. *IEEE Transactions on Circuits and Systems for Video Technology*, *33*(12), 7630–7644. https://doi.org/10.1109/TCSVT.2023.3276753

Li, Y., Li, Q., Gao, C., Gao, S., Wu, H., & Liu, R. (2025). Pfenet: Towards precise feature extraction from sparse point cloud for 3d object detection. *Neural Networks*, *185*, 107144. https://doi.org/https://doi.org/10.1016/j.neunet.2025.107144

Li, Z., Zhang, Y., Wu, C., Zhu, J., & Zhang, L. (2024). Ho-gaussian: Hybrid optimization of 3d gaussian splatting for urban scenes. In *European conference on computer vision* (pp. 19–36). Springer.

Liu, L., Gu, J., Lin, K. Z., Chua, T.-S., & Theobalt, C. (2020). Neural sparse voxel fields. *NeurIPS*, .

Lu, T., Yu, M., Xu, L., Xiangli, Y., Wang, L., Lin, D., & Dai, B. (2024). Scaffold-GS: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 20654–20664).

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., & Ng, R. (2020). NeRF: Representing scenes as neural radiance fields for view synthesis. In *Eccv*.

Müller, T., Evans, A., Schied, C., & Keller, A. (2022). Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics*, *41*(4), 102:1–102:15. https://doi.org/10.1145/3528223.3530127

Rakhimov, R., Ardelean, A.-T., Lempitsky, V., & Burnaev, E. (2022). Npbg + +: Accelerating neural point-based graphics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 15969–15979).

Ren, K., Jiang, L., Lu, T., Yu, M., Xu, L., Ni, Z., & Dai, B. (2024). Octree-GS: Towards consistent real-time rendering with LOD-structured 3d gaussians. https://arxiv.org/abs/2403.17898.

Sandström, E., Li, Y., Van Gool, L., & Martin, R. O. (2023). Point-SLAM: Dense neural point cloud-based SLAM. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*.

Schönberger, J. L., & Frahm, J.-M. (2016). Structure-from-motion revisited. In *2016 IEEE Conference on computer vision and pattern recognition (CVPR)* (pp. 4104–4113). https://doi.org/10.1109/CVPR.2016.445

Shao, Y., Tan, A., Wang, B., Yan, T., Sun, Z., Zhang, Y., & Liu, J. (2024). Ms23d: A 3d object detection method using multi-scale semantic feature points to construct 3d feature layer. *Neural Networks*, *179*, 106623. https://doi.org/https://doi.org/10.1016/j.neunet.2024.106623

Shi, Y., Wu, Y., Wu, C., Liu, X., Zhao, C., Feng, H., Zhang, J., Zhou, B., Ding, E., & Wang, J. (2025). Gir: 3d gaussian inverse rendering for relightable scene factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, .

Sloan, P.-P., Kautz, J., & Snyder, J. (2023). Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2* , pp. . New York, NY, USA: Association for Computing Machinery. (1st ed.). https://doi.org/10.1145/3596711.3596749.

Snavely, N., Seitz, S. M., & Szeliski, R. (2006). Photo tourism: Exploring photo collections in 3d. In *Acm siggraph 2006 papers* (pp. 835–846).

Sun, C., Sun, M., & Chen, H.-T. (2022). Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5459–5469).

Verbin, D., Hedman, P., Mildenhall, B., Zickler, T., Barron, J. T., & Srinivasan, P. P. (2022). Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on computer vision and pattern recognition (CVPR)* (pp. 5481–5490). IEEE.

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

Wei, M., Wu, Q., Zheng, J., Rezatofighi, H., & Cai, J. (2024). Normal-GS: 3d gaussian splatting with normal-involved rendering. *arXiv preprint arXiv:2410.20593*

Xu, H., Chen, W., Sun, B., Xie, X., & Kang, W. (2024). RobustMVS: Single domain generalized deep multi-view stereo. *IEEE Transactions on Circuits and Systems for Video Technology*, *34*(10), 9181–9194. https://doi.org/10.1109/TCSVT.2024.3399458

Yang, W., Huang, J., He, X., & Wen, S. (2024a). Fixed-time synchronization of complex-valued neural networks for image protection and 3d point cloud information protection. *Neural Networks*, *172*, 106089. https://doi.org/https://doi.org/10.1016/j.neunet.2023.12.043

Yang, Z., Gao, X., Sun, Y.-T., Huang, Y., Lyu, X., Zhou, W., Jiao, S., Qi, X., & Jin, X. (2024b). Spec-gaussian: Anisotropic view-dependent appearance for 3d gaussian splatting. *Advances in Neural Information Processing Systems*, *37*, 61192–61216.

Ye, Z., Bao, C., Zhou, X., Liu, H., Bao, H., & Zhang, G. (2024a). Ec-sfm: Efficient covisibility-based structure-from-motion for both sequential and unordered images. *IEEE Transactions on Circuits and Systems for Video Technology*, *34*(1), 110–123. https://doi.org/10.1109/TCSVT.2023.3285479

Ye, Z., Li, W., Liu, S., Qiao, P., & Dou, Y. (2024b). Absgs: Recovering fine details in 3d gaussian splatting. In *Proceedings of the 32nd ACM international conference on multimedia* (pp. 1053–1061).

Yifan, W., Serena, F., Wu, S., Öztireli, C., & Sorkine-Hornung, O. (2019). Differentiable surface splatting for point-based geometry processing. *ACM Transactions on Graphics (TOG)*, *38*(6), 1–14.

Zhang, J., Zhan, F., Xu, M., Lu, S., & Xing, E. (2024a). FreGS: 3d gaussian splatting with progressive frequency regularization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 21424–21433).

Zhang, K., Riegler, G., Snavely, N., & Koltun, V. (2020). NeRF + +: Analyzing and improving neural radiance fields. *arXiv:2010.07492*

Zhang, Q., Baek, S.-H., Rusinkiewicz, S., & Heide, F. (2022). Differentiable point-based radiance fields for efficient view synthesis. In *Siggraph asia 2022 conference papers* SA '22. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3550469.3555413

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 586–595).

Zhang, Z., Hu, W., Lao, Y., He, T., & Zhao, H. (2024b). Pixel-GS: Density control with pixel-aware gradient for 3d gaussian splatting. *arXiv:2403.15530*