Advanced Rendering & Modeling with 3D Gaussians

Minglun Gong
In collaboration with Zhentao Huang
School of Computer Sci., Univ. of Guelph

Outline

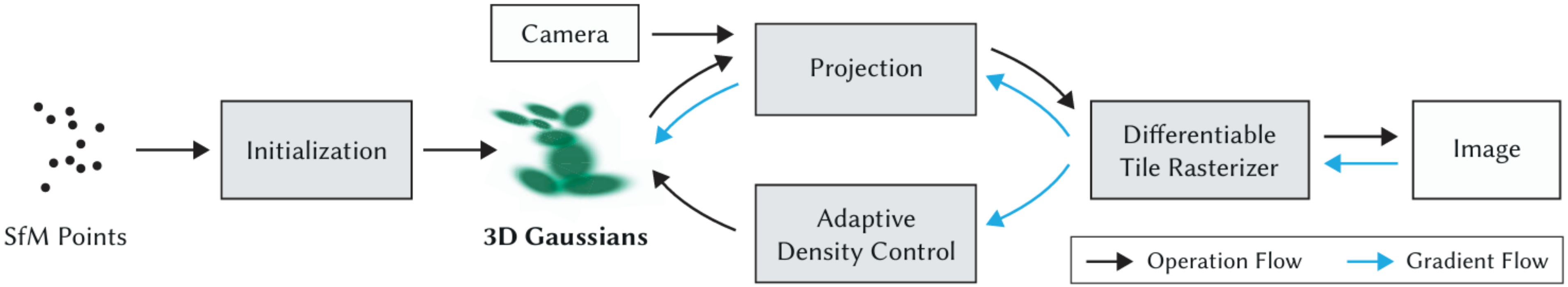
- Introduction
  - Overview of the 3D Gaussian Splatting Approach
- Enhancements on Scene Rendering
  - Mini-Splatting Technique
  - Proposed Textured-GS
- Enhancements on Scene Modeling
  - Planar-based GS for Reconstruction Technique
  - Proposed Gaussian Set Surface Reconstruction
- Conclusions

3D Gaussian Splatting (SIGGRAPH 2023)

3D Gaussian Splatting for Real-Time Radiance Field Rendering

BERNHARD KERBL*, Inria, Université Côte d'Azur, France
GEORGIOS KOPANAS*, Inria, Université Côte d'Azur, France
THOMAS LEIMKÜHLER, Max-Planck-Institut für Informatik, Germany
GEORGE DRETTAKIS, Inria, Université Côte d'Azur, France

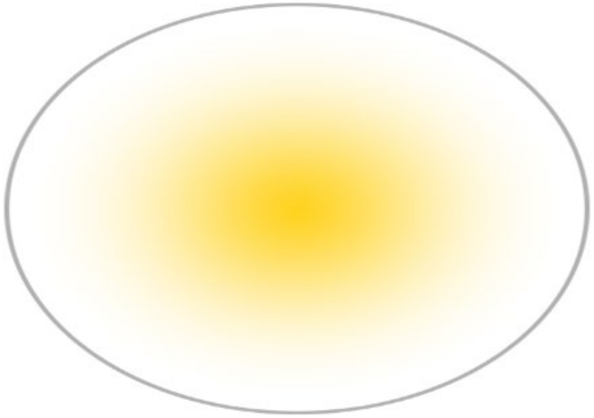
3D Gaussian Splatting Overview

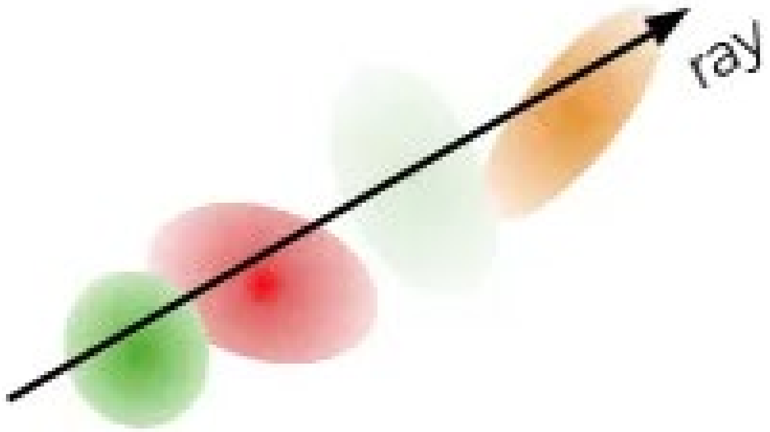


- Gaussian Representation
  - Each scene point is represented by a Gaussian.
  - Facilitates smooth transitions between data points.
- Splatting Process
  - Gaussians are ‘splat’ onto the viewing plane.
  - Each Gaussian projects its properties (color, density) based on its geometric relation to the viewpoint.

3D Gaussian Representation

- Position (3 parameters)
  - Coordinates: X, Y, Z
- Scale (3 parameters)
  - Scale along X, Y, Z axes
- Orientation (4 parameters)
  - Quaternion: w, x, y, z
- Color (3 to 48 parameters)
  - RGB/Spherical Harmonics
- Opacity (1 parameter)
  - Transparency level





Visualization of Gaussians (7 Million)

Rendering result

Fully opaque

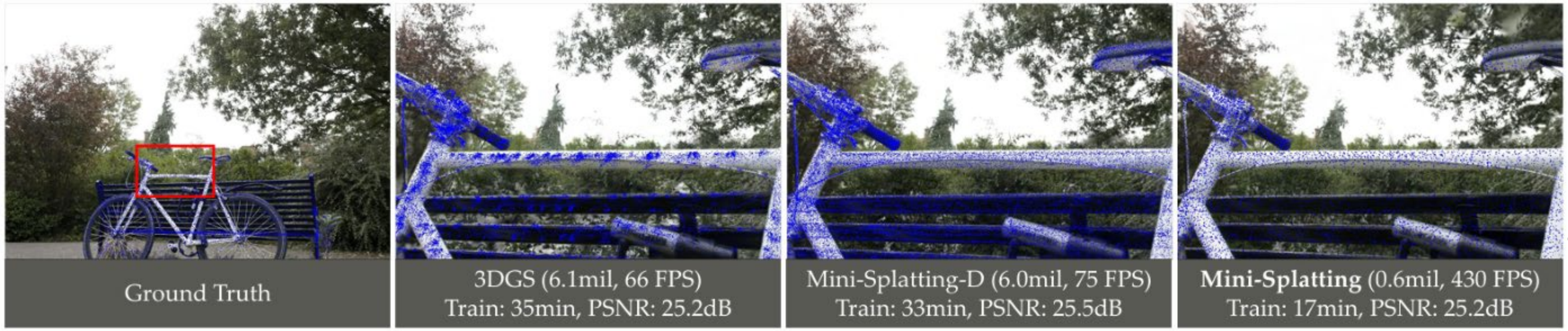
Enhancements on Rendering

Mini-Splatting (ECCV 2024)

Mini-Splatting: Representing Scenes with a Constrained Number of Gaussians

Guangchi Fang and Bing Wang*

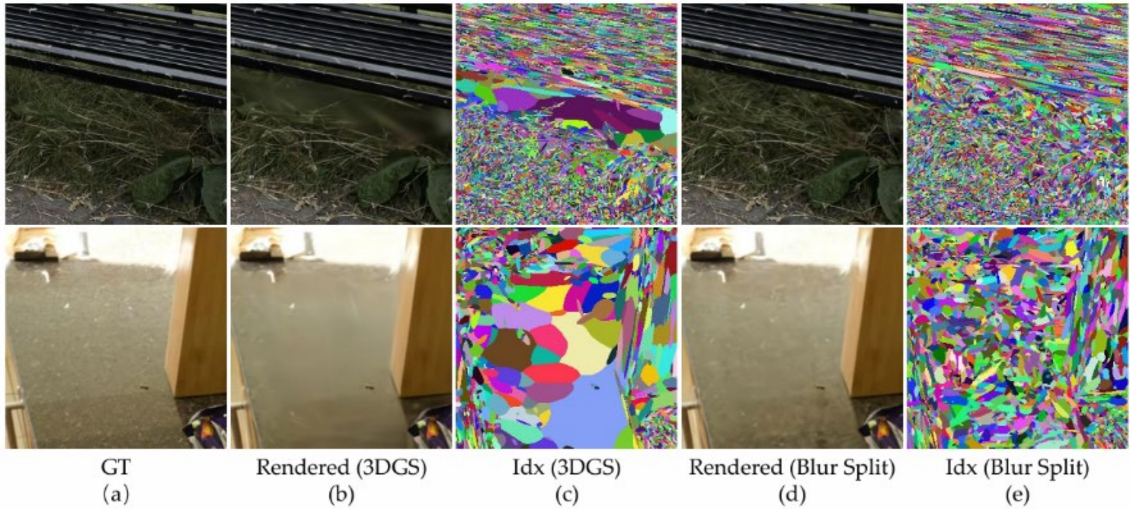
Spatial Intelligence Group, The Hong Kong Polytechnic University
guangchi.fang@gmail.com, bingwang@polyu.edu.hk



Mini-Splatting Overview

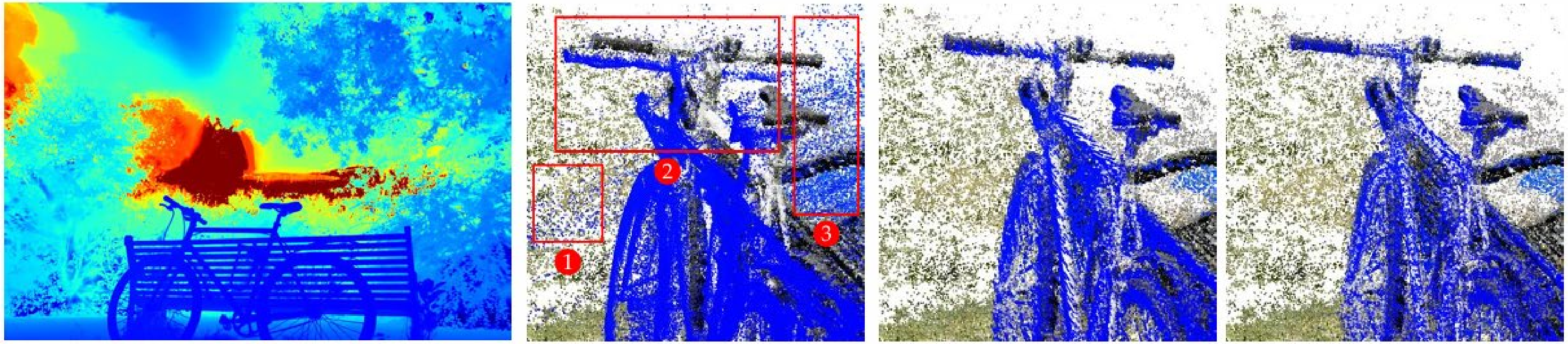
- Limitations Addressed:
  - Efficient Representation: Represent scenes with fewer Gaussians while maintaining rendering quality.
  - Improved Gaussian Distribution: More uniform and efficient Gaussian distribution via strategic densification and simplification.
  - Balanced Trade-Off: Optimizes rendering quality against resource and storage efficiency.
- Densification:
  - Blur Split: Enhances detail by splitting Gaussians in blurry areas.
  - Depth Reinitialization: Improves geometry using merged depth points.
- Simplification:
  - Intersection Preserving: Retains Gaussians critical to image impact.
  - Importance-weighted Sampling: Ensures geometric integrity and rendering quality.

Blur Split Strategy



- **Objective:**
  - Targets the reduction of Gaussian blur artifacts to enhance overall image clarity.
- **Implementation:**
  - Render the index of the Gaussian with the maximum contribution per pixel.
  - Highlight the connection between blurry areas and large Gaussians.

Depth Reinitialization Strategy



Blending Depth (a) | Depth Points (Blending) (b) | Depth Points (Center) (c) | Depth Points (Mid) (d)

- **Objective:**
  - Use dense depth to address 'overlapping' and 'under-reconstruction' issues.
- **Depth Calculation:**
  - Blending Depth: Replaces Gaussian color with depth, leading to issues such as depth collapse, object misalignment, and blending boundaries.
  - Mid-point Depth: Calculates the midpoint between two intersection points of an input ray and the Gaussian ellipsoid.

Simplification Techniques

- **Objective:**
  - Enhance Gaussian representation efficiency while maintaining rendering quality after densification.
- **Intersection Preserving:**
  - Concept: Discards non-intersecting Gaussians to preserve rendering quality.
  - Implementation: Focuses on Gaussians that directly intersect with scene rays, specifically using mid-point depth calculations.
- **Importance-weighted Sampling:**
  - Implementation: Assigns importance values to Gaussians, using stochastic sampling to maintain geometric integrity.
    - $P_i = \frac{I_i}{\sum_{i=1}^{N} I_i}$
- **Problem with Direct Pruning:**
  - Can degrade rendering quality by disrupting local geometry.
  - Gaussians with similar importance are kept or removed together.

Open Issues

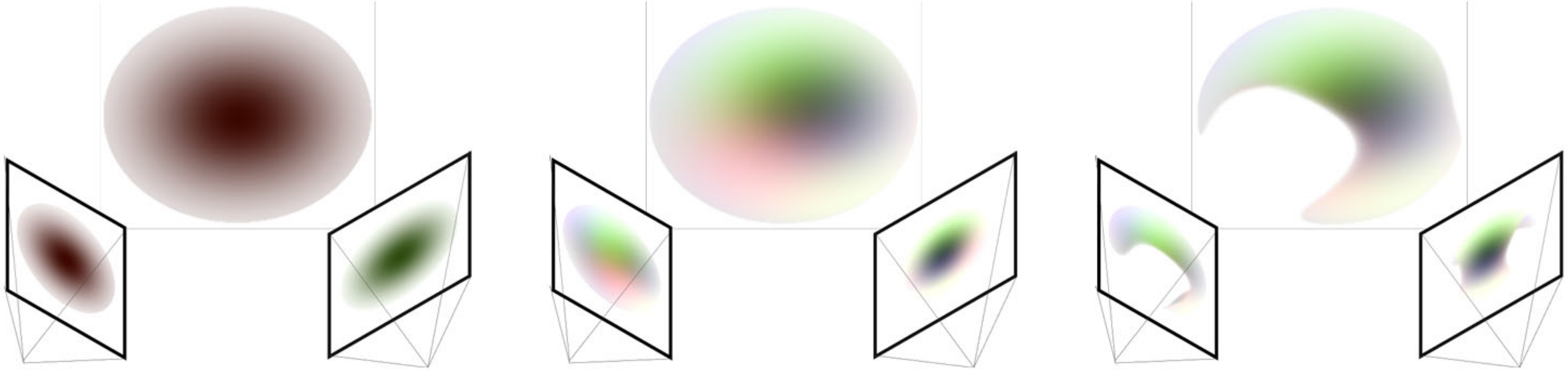
- Lack of Spatially Dependent Texturing:
  - Color Definition: Utilizes Spherical Harmonics (SH) coefficients to assign color attributes to each Gaussian.
  - View-Dependent Texturing: Each Gaussian exhibits only one color under a given viewing angle.
  - Lacking Spatial Variation: Requires a large number of Gaussians to approximate spatial variations, hindering efficiency and realism.
- Challenges with Sharp Boundaries:
  - Gaussian Shape: The inherently rounded shape of Gaussians in 3DGS complicates the accurate portrayal of sharp object boundaries.
  - Edge Definition: Difficulty in defining crisp edges and detailed textural transitions, potentially reducing detail and the overall quality of rendered scenes.
  - Recent CVPR 2025 paper: "3D Half-Gaussian Splatting"

Textured-GS (GI 2025)

Textured-GS: Gaussian Splatting with Spatially Defined Color and Opacity

Zhentao Huang
University of Guelph
School of Computer Science
zhentao@uoguelph.ca

Minglun Gong
University of Guelph
School of Computer Science
minglun@uoguelph.ca



(a) 3D Gaussian (b) Gaussian with Color Texture (c) Gaussian with Color & Opacity Texture

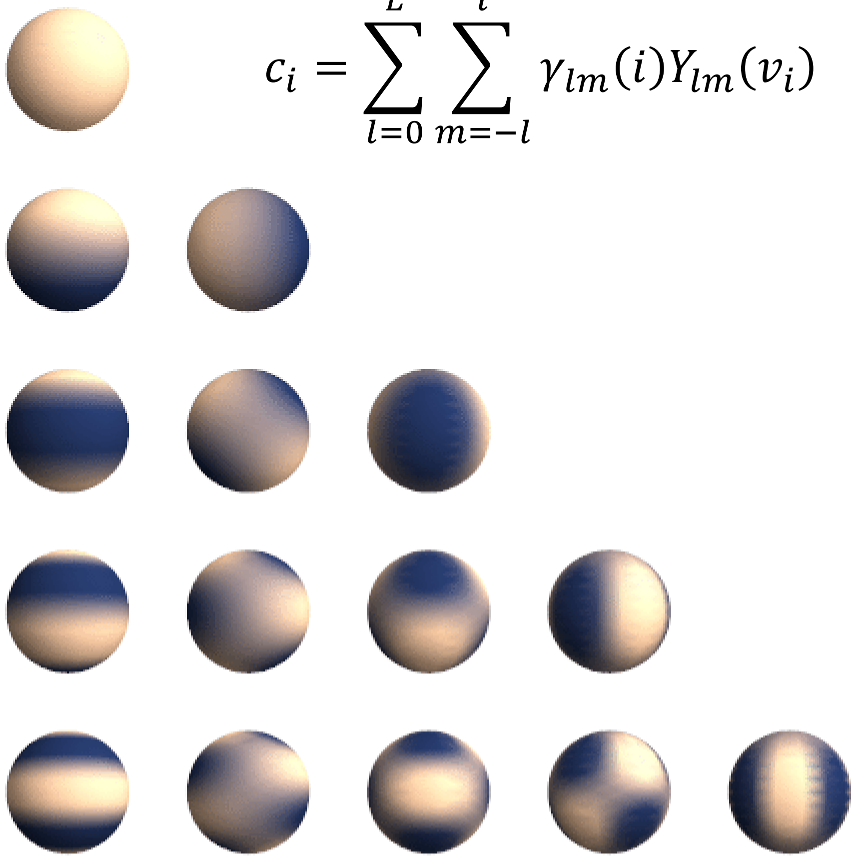
Textured-GS Overview

- Objectives:
  - Innovative Texturing Method: Introduces a novel approach to enhance rendering details, leveraging SH without increasing parameter overhead.
  - Enhanced Gaussian Representation: Enriches each Gaussian with spatially defined color and opacity variations, enabling more detailed and diverse surface appearances.
- Proposed Approach:
  - Color Variation: Facilitates dynamic color changes across different areas of a Gaussian's ellipsoidal surface, not just based on the viewing angle.
  - Opacity Integration: Incorporates an opacity channel within the SH framework to accurately model transparency variations across the Gaussian surface.

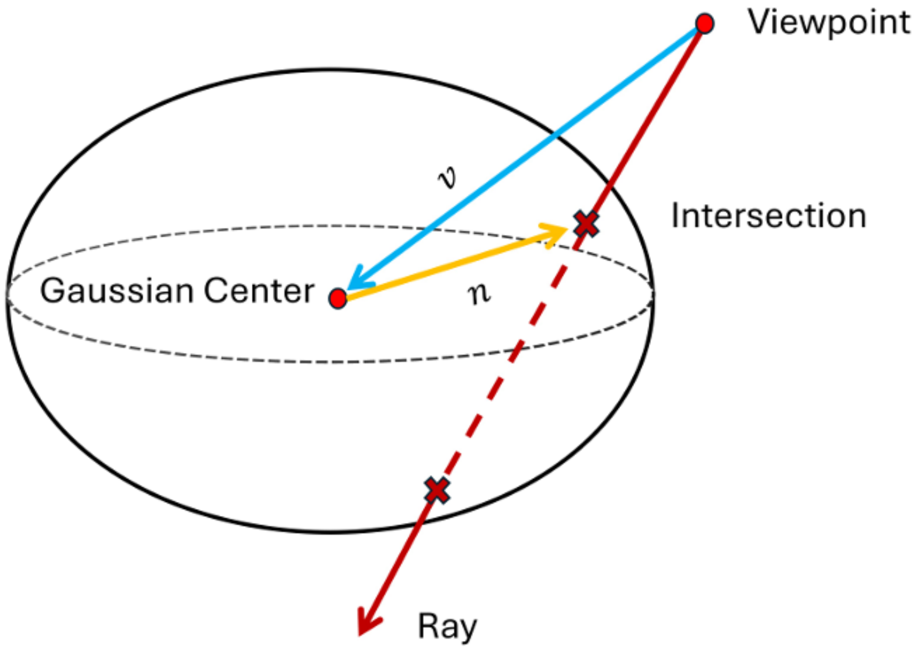
Spherical Harmonics

- Definition
  - Special functions defined on the surface of a sphere.
- Properties
  - Orthogonal functions that form a complete set.
  - Any function on the sphere's surface can be represented as a sum of spherical harmonics.
- Applications in 3DGS
  - Used for modeling illumination and reflection.
  - Input: Viewing angle.
  - Output: RGB color.

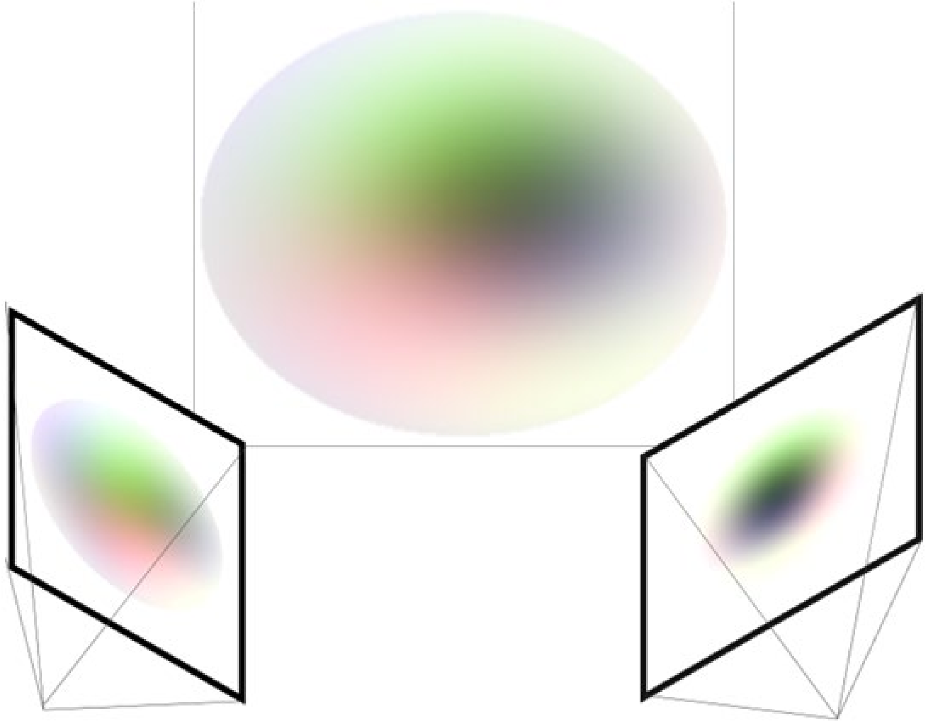
$$c_i = \sum_{l=0}^{L} \sum_{m=-l}^{l} \gamma_{lm}(i) Y_{lm}(v_i)$$



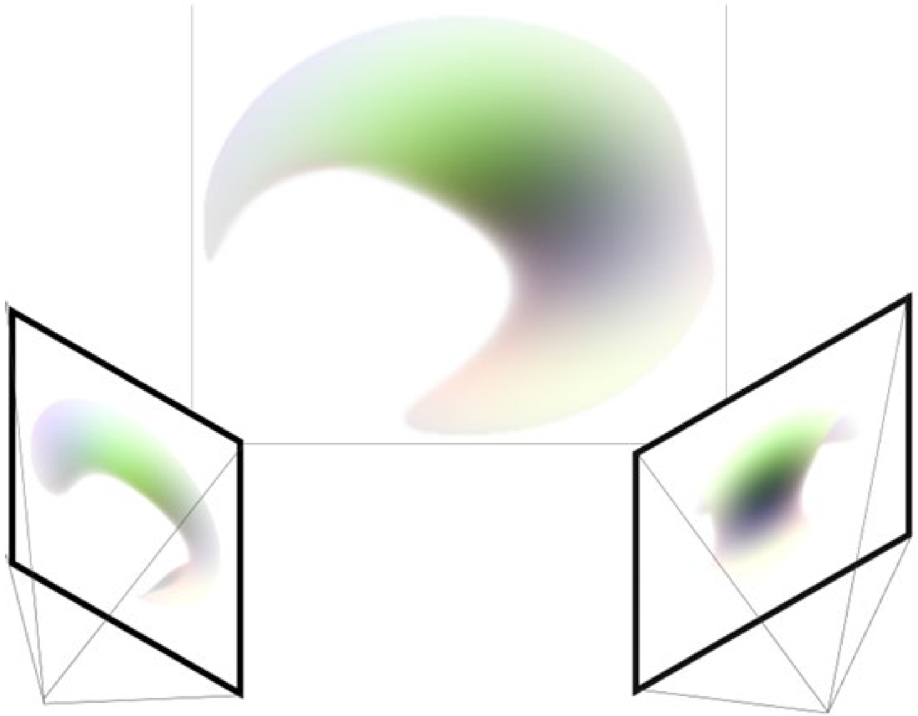
Adjust SH Sampling Parameters



Replacing v with n



Add Opacity Channel



- Opacity Modeling:
  - Integrates an opacity channel into each SH coefficient.
- Advantages:
  - Variable Opacity: Enhances shape perception, improving the modeling of sharp object boundaries.
  - Unidirectional Gaussians: Supports creating Gaussians that are opaque on one side and transparent on the other.

Visual Comparison

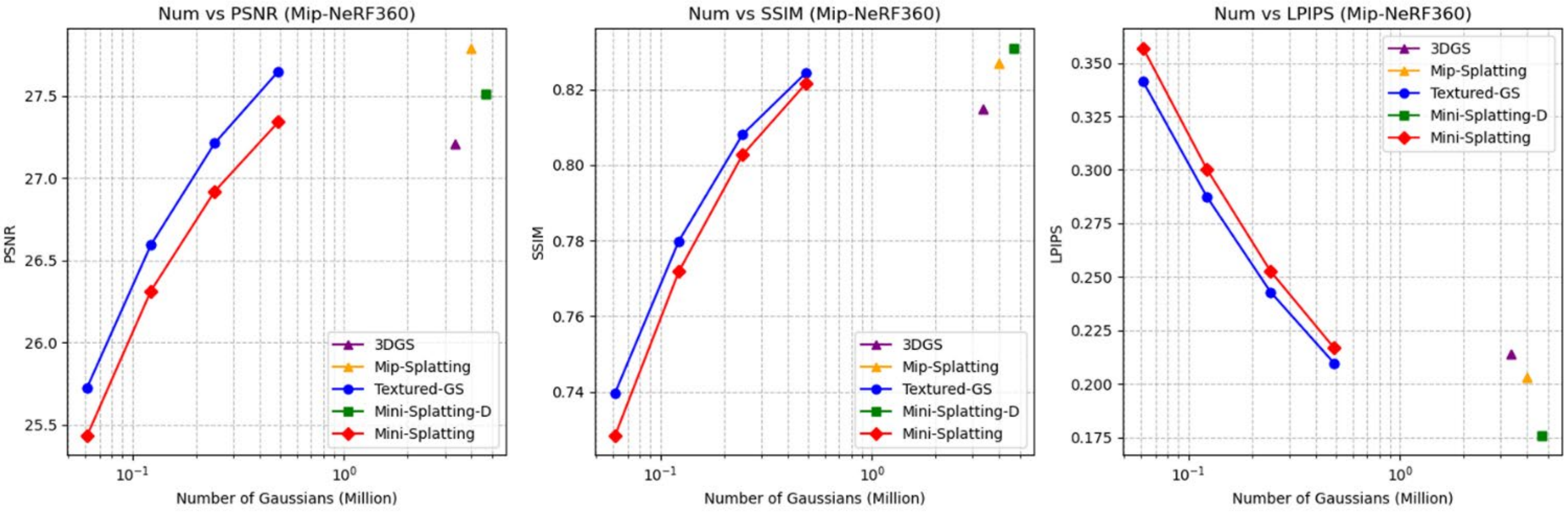
Visual Comparison (Cont'd)

(a) Ground Truth (b) 3DGS [11] (c) Mini-Splatting [5] (d) Ours

Quantitative Evaluation

Method | Metric	Mip-NeRF 360				Tanks&Temples				Deep Blending			
	SSIM ↑	PSNR ↑	LPIPS ↓	Num	SSIM ↑	PSNR ↑	LPIPS ↓	Num	SSIM ↑	PSNR ↑	LPIPS ↓	Num
Plenoxels [6]	0.626	23.08	0.463	-	0.719	21.08	0.379	-	0.795	23.06	0.510	-
INGP-Big [19]	0.699	25.59	0.331	-	0.745	21.92	0.305	-	0.817	24.96	0.390	-
mip-NeRF 360 [1]	0.792	27.69	0.237	-	0.759	22.22	0.257	-	0.901	29.40	0.245	-
Zip-NeRF [2]	0.828	28.54	0.189	-	-	-	-	-	-	-	-	-
3DGS [11]	0.815	27.21	0.214	**3.36**	0.841	23.14	0.183	**1.78**	0.903	29.41	0.243	**2.98**
Mip-Splatting [25]	0.827	27.79	0.203	**3.97**	-	-	-	-	-	-	-	-
3DGS-MCMC [12]	-	-	-	-	0.860	24.29	0.190	**1.78**	0.890	29.67	0.320	**2.98**
Mini-Splatting-D [5]	0.831	27.51	0.176	**4.69**	0.853	23.23	0.140	**4.28**	0.906	29.88	0.211	**4.63**
Mini-Splatting (30K) [5]	0.822	27.34	0.217	**0.49**	0.835	23.18	0.202	**0.20**	0.908	29.98	0.253	**0.35**
Mini-Splatting (44K) [5]	0.822	27.34	0.213	**0.49**	0.840	23.34	0.203	**0.20**	0.906	29.80	0.250	**0.35**
Textured-GS	0.825	27.64	0.209	**0.49**	0.843	23.49	0.191	**0.20**	0.909	30.02	0.248	**0.35**

Performance under Different Settings

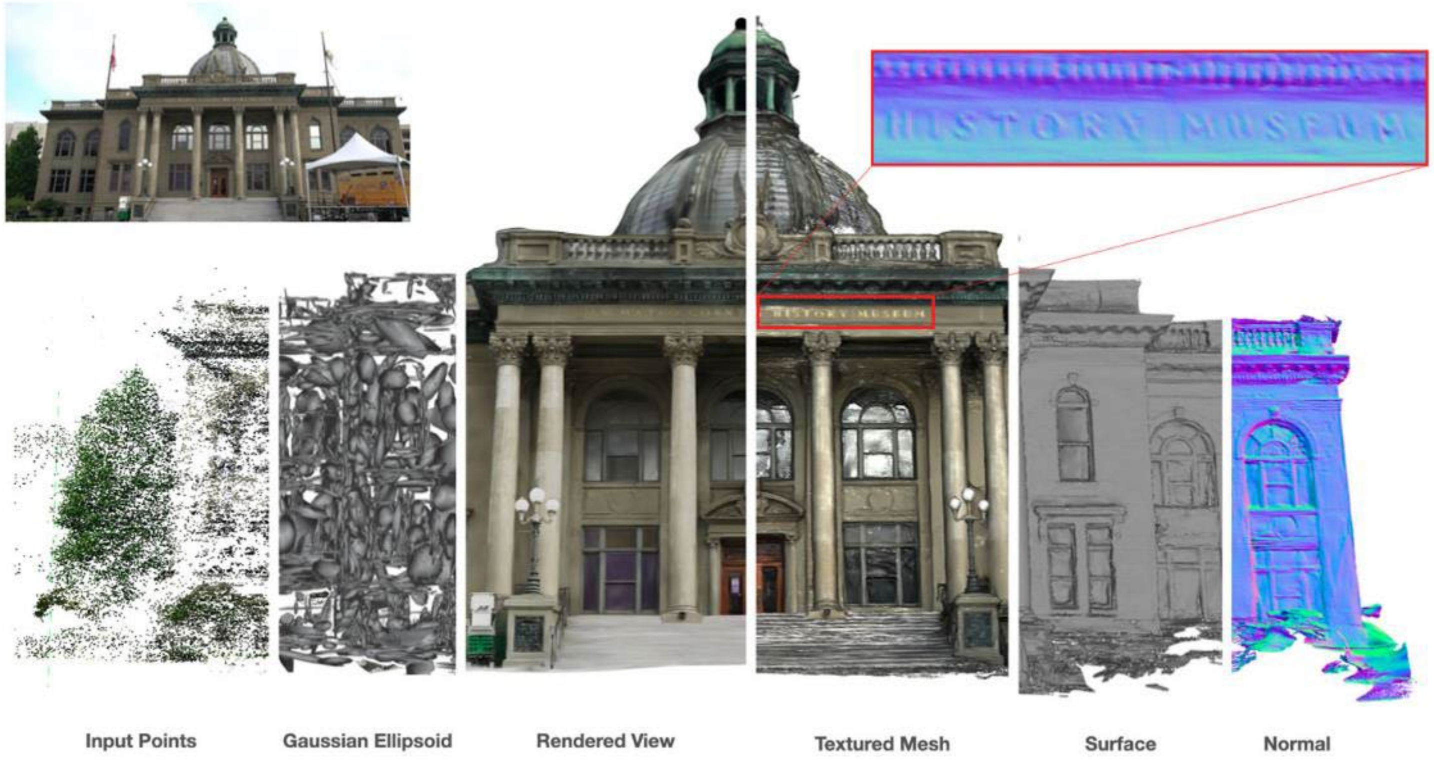


ENHANCEMENTS ON MODELING

PGSR (TVCG 2024)

PGSR: Planar-based Gaussian Splatting for Efficient and High-Fidelity Surface Reconstruction

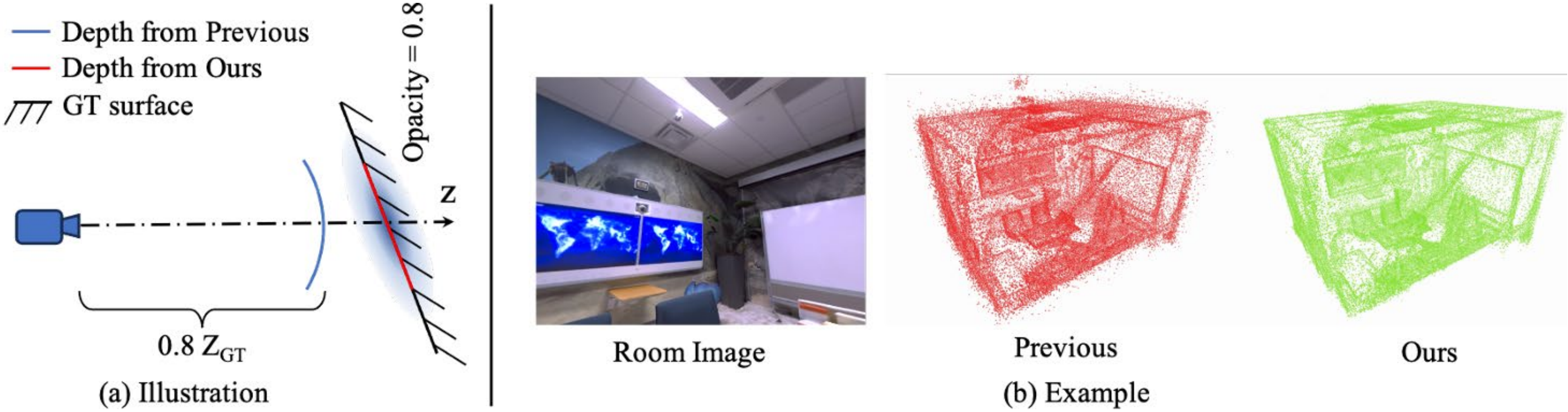
Danpeng Chen, Hai Li, Weicai Ye, Yifan Wang, Weijian Xie, Shangjin Zhai,
Nan Wang, Haomin Liu, Hujun Bao, Guofeng Zhang



PGSR Overview

- Limitations Addressed:
  - 3DGS is fast for training & rendering, but has poor geometric accuracy.
  - Unstructured Gaussian point clouds make it difficult to ensure surface consistency.
  - Existing methods struggle with accurate mesh extraction.
- Proposed Approach:
  - Unbiased Depth Rendering: Converts Gaussian points into planar representations for accurate depth estimation.
  - Single-View & Multi-View Regularization: Improves global geometric consistency.
  - Exposure Compensation: Handles illumination variations for better reconstruction.

Unbiased Depth Rendering



- **Normal Determination:**
  - Flattens Gaussian shapes and utilizes the direction of the minimum scale factor as the normal for each Gaussian.
- **Depth Map Calculation:**
  - Computes depth maps by intersecting rays with these flattened Gaussian planes, avoiding the issue of producing curved depths.

Single-View & Multi-View Regularization

- Single-View Regularization:
  - Depth and Normal Consistency: Utilizes rendered depth maps to calculate normals, minimizing its difference with the rendered normal map.
- Multi-View Regularization:
  - Geometric Consistency: Applies homography, correlating pixels between reference and neighboring frames.
  - Photometric Consistency: Use NCC to match colors across views.

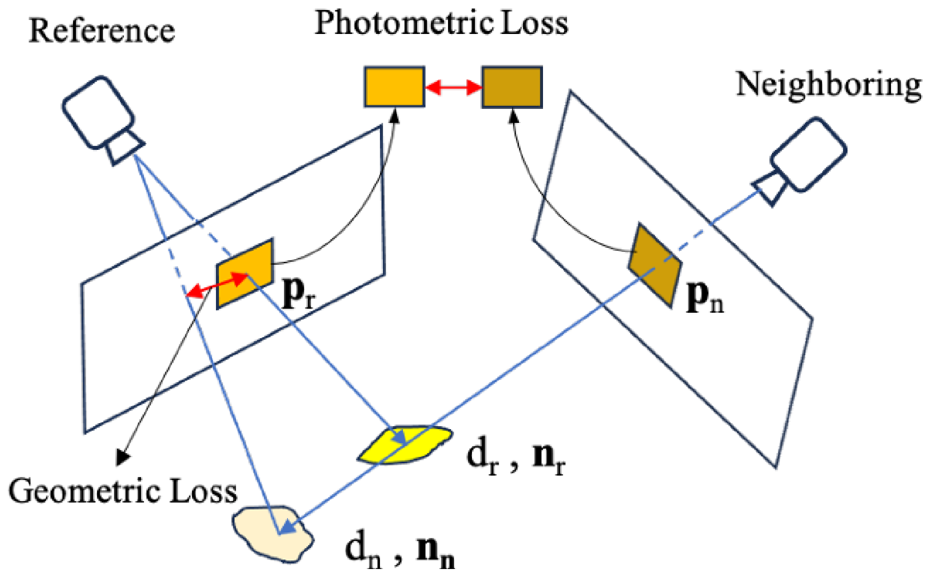


Fig. 9: **Multi-view photometric and geometric loss.**

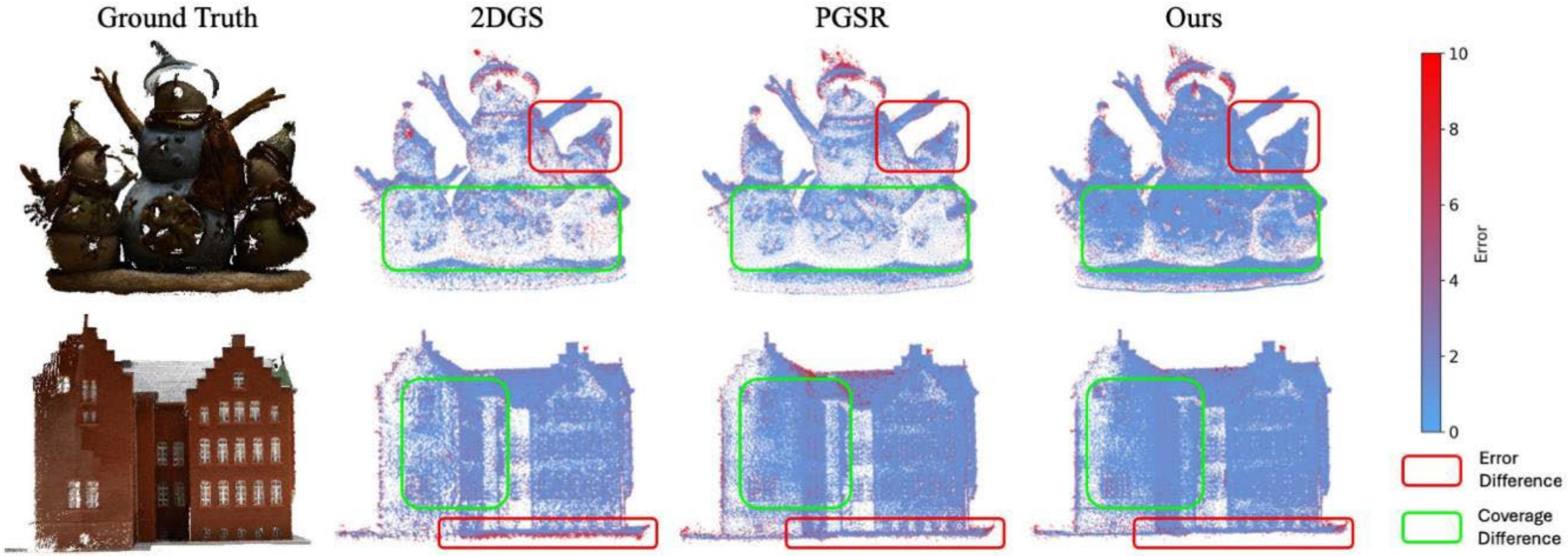
Open Issues

- Rely on Depth & Normal Maps:
  - Computing loss using blended depth and normal maps is less effective for gradient descent on individual Gaussians.
  - Generates 3D models from blended maps, then combining these maps into a point cloud, rather than directly utilizing 3D Gaussians.
- Cannot directly edit Gaussians
  - Limits applications, such as scene editing and dynamic object tracking.
- NCC Loss Limitations:
  - Fail to account for high matching costs caused by occlusions and view-dependent lighting effects.
- Gaussian Optimization:
  - Does not directly optimize Gaussian distribution on object surfaces.
  - Does not directly optimize the scales of Gaussians.
  - The optimization does not have impact on Gaussians that are positioned behind the object surface due to occlusion.

Gaussian Set Surface Reconstruction

Gaussian Set Surface Reconstruction through Per-Gaussian Optimization

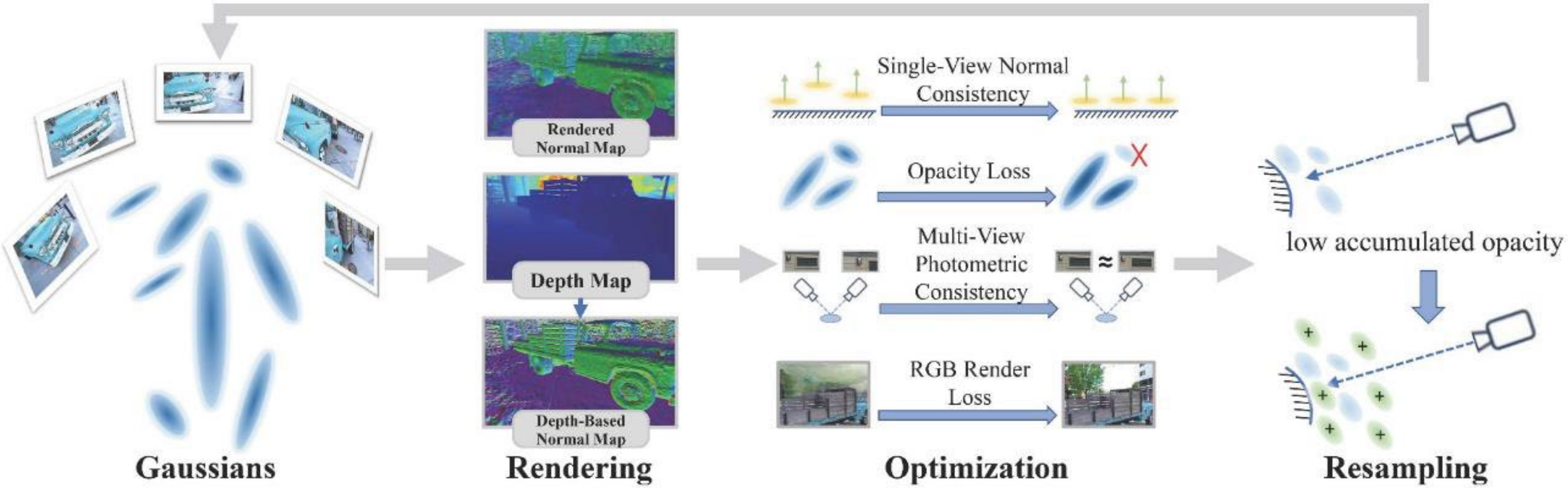
Zhentao Huang, University of Guelph
Di Wu, University of Macau
Zhenbang He, UBC Okanagan
Minglun Gong, University of Guelph



GSSR Overview

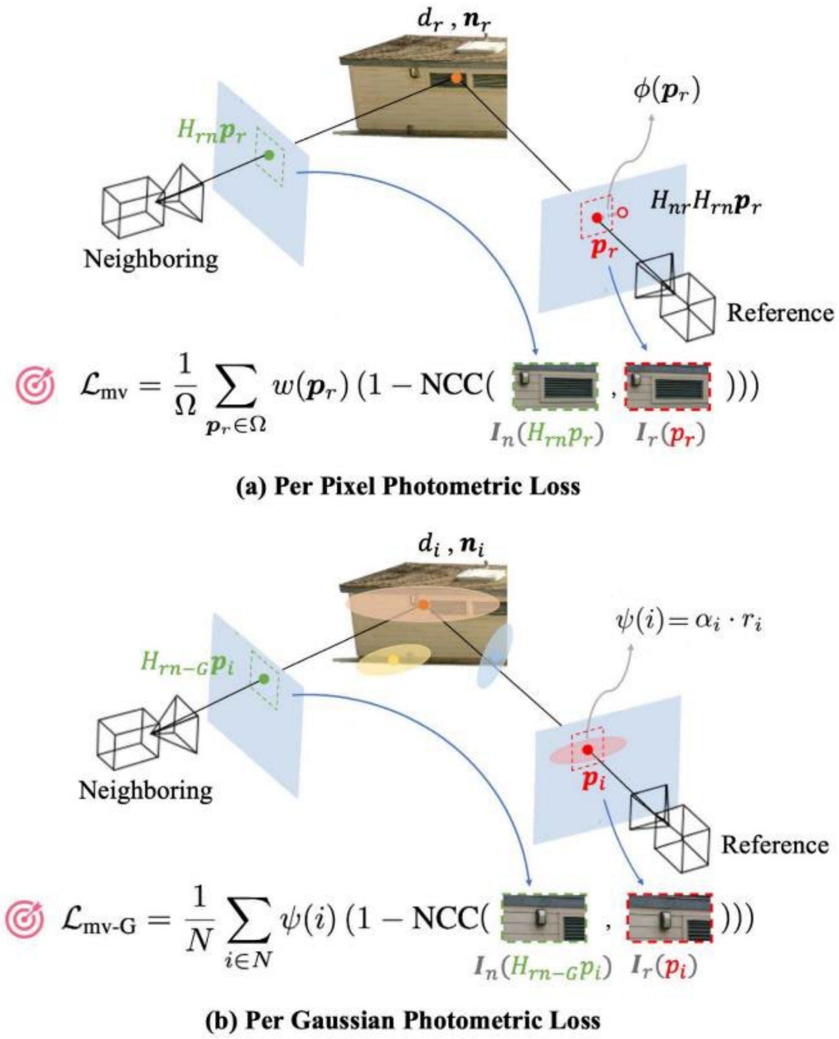
- A Gaussian Set Surface (GSS) is a collection of Gaussians that:
  - Centered on latent scene surfaces;
  - Dominant normals aligned with the surface normal;
  - Uniformly distributed along the latent surface.
- Gaussian Set Surface Reconstruction (GSSR) enforces fine-grained geometric alignment of 3D Gaussians through:
  - Pixel-level and Gaussian-level single-view normal consistency;
  - Multi-view photometric consistency;
  - Opacity regularization for eliminating redundant Gaussians;
  - Depth- and normal-guided Gaussian reinitialization for cleaner, more uniform distribution.
- Combines the geometric manipulability of Point Set Surfaces with 3DGS's strengths in photorealistic novel view synthesis.

Algorithm Pipeline



- Render the depth map, depth-based normals and alpha-blended normals.
- The optimization stage includes four major components:
  - Single-view normal consistency, Multi-view photometric consistency, RGB rendering loss, Opacity regularization
- Gaussians are periodically resampled

Photometric Loss at Pixel & Gaussian Levels

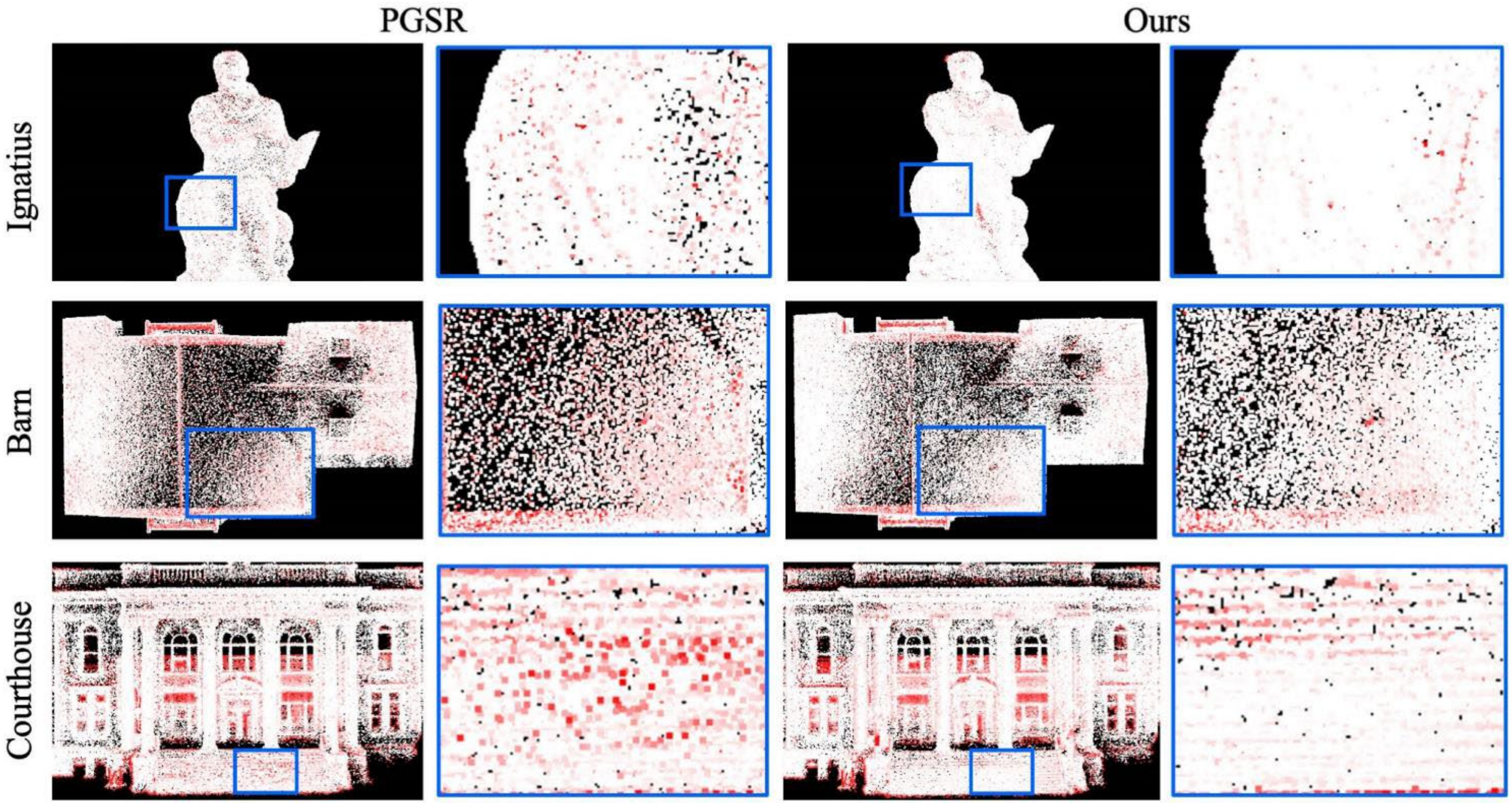


(a) Per Pixel Photometric Loss

(b) Per Gaussian Photometric Loss

- $\boldsymbol{L_{mv}} = \frac{1}{|\Omega|}\sum_{p_r \in \Omega} \boldsymbol{w(p_r)} \times (1 - \boldsymbol{NCC}(I_r(p_r), I_n(H_{rn}p_r)))$
  - Loss calculated for each pixel ($\boldsymbol{p_r}$) in a reference image $\boldsymbol{r}$
  - Loss are backpropagated to all Gaussians that project to $\boldsymbol{p_r}$
- $\boldsymbol{L_{mv-G}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\psi(i)} \times (1 - \boldsymbol{NCC}(I_r(p_r), I_n(H_{rn-G}p_r)))$
  - Loss calculated for the $\boldsymbol{i^{th}}$ Gaussian
  - Directly impact the parameters for the $\boldsymbol{i^{th}}$ Gaussian

Comparison of Gaussian Centroid Precision



Gaussian Density Control

- Observations:
  - Gaussians generated by 3DGS or PGSR form a thick layer surrounding the latent surface
  - Many of them are semi-transparent.
- Depth filtered opacity loss
  - Encourages each Gaussian's opacity to converge toward either 0 (fully transparent and removable) or 1 (fully opaque)
  - Filtered by depth map for proper occlusion handling
  - $L_o = \frac{1}{N}\sum_{i=1}^{N}(\log \alpha_i + \log(1 - \alpha_i))$

- Observations:
  - Gaussians generated by 3DGS or PGSR are not uniformly distributed
  - Poorly covered areas have low opacity
- View-based, opacity-guided resampling
  - The sampling probability at each pixel p is set to $1 - \alpha_{acc}(p)$

Comparison on Gaussian Density Distribution

Reconstruction on DTU Dataset

	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
2DGS* [15]	0.49	0.79	0.34	0.42	0.95	0.95	0.83	1.25	1.24	0.64	0.62	1.34	0.44	0.69	0.48	0.76
GOF* [34]	0.49	0.83	0.36	0.38	1.33	0.87	0.73	1.24	1.32	0.66	0.73	1.26	0.52	0.82	0.51	0.80
PGSR* [3]	0.34	0.55	0.39	0.35	0.78	0.58	0.49	1.09	0.63	0.59	0.47	0.50	0.30	0.37	0.34	0.52
GausSurf† [30]	0.35	0.55	0.34	0.34	0.77	0.58	0.51	1.10	0.69	0.60	0.43	0.49	0.32	0.40	0.37	0.52
Ours	0.33	0.57	0.37	0.33	0.78	0.62	0.51	1.11	0.68	0.59	0.48	0.55	0.30	0.38	0.35	0.53
2DGS*	0.66	1.02	0.55	0.60	0.88	1.03	0.92	0.55	0.95	0.47	0.52	0.99	0.63	0.42	0.41	0.71
GOF*	0.98	1.28	0.99	0.85	1.11	1.29	1.08	0.77	1.10	0.62	0.65	1.05	0.79	0.48	0.61	0.91
PGSR*	0.58	0.68	0.85	0.71	0.74	0.64	0.68	0.51	0.74	0.45	0.46	0.60	0.46	0.37	0.37	0.59
Ours	0.32	0.47	0.46	0.29	0.56	0.42	0.39	0.43	0.52	0.33	0.26	0.33	0.21	0.26	0.25	0.37
2DGS*	0.92	0.87	1.02	0.72	1.06	1.47	1.25	1.94	1.70	1.11	1.56	1.51	0.81	1.33	1.09	1.22
GOF*	0.71	0.72	0.81	0.59	0.96	1.33	1.12	1.78	1.70	0.99	1.38	1.14	0.69	1.19	0.92	1.07
PGSR*	0.75	0.72	0.88	0.64	0.82	1.16	0.97	1.69	1.44	0.96	1.30	0.90	0.68	1.05	0.86	0.99
Ours	0.62	0.61	0.58	0.51	0.56	0.88	0.67	1.52	1.17	0.82	0.85	0.54	0.46	0.64	0.55	0.73

* *Reproduced results using the authors' official implementation.*
† *The source code is not available; only mesh-based evaluation results are reported.*

Reconstruction on Tanks and Temples Dataset

	Barn	Caterpillar	Courthouse	Ignatius	Meetingroom	Truck	Mean
2DGS* [15]	0.46	0.24	0.15	0.49	0.19	0.45	0.33
GOF* [34]	0.55	0.39	0.28	0.71	0.26	0.57	0.46
PGSR* [3]	0.65	0.45	0.21	0.81	0.33	0.62	0.51
GausSurf† [30]	0.50	0.42	0.30	0.73	0.39	0.65	0.50
Ours	0.64	0.42	0.22	0.80	0.32	0.59	0.50
2DGS*	0.65	0.59	0.62	0.71	0.47	0.71	0.62
GOF*	0.65	0.60	0.62	0.77	0.49	0.70	0.64
PGSR*	0.63	0.58	0.56	0.70	0.49	0.69	0.61
Ours	0.66	0.66	0.52	0.83	0.47	0.78	0.65
2DGS*	0.08	0.07	0.04	0.09	0.04	0.11	0.07
GOF*	0.12	0.09	0.05	0.15	0.05	0.16	0.10
PGSR*	0.16	0.13	0.08	0.16	0.09	0.20	0.14
Ours	0.20	0.19	0.08	0.28	0.07	0.32	0.19

Novel View Synthesis on Mip-NeRF360 dataset

Method	Indoor scenes			Outdoor scenes			Average on all scenes		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
2DGS [15]	30.40	0.916	0.195	24.34	0.717	0.246	27.37	0.817	0.221
GOF [34]	30.79	0.924	0.184	24.82	0.750	0.202	27.81	0.837	0.193
PGSR* [3]	30.20	0.930	0.158	24.77	0.752	0.204	27.49	0.841	0.181
GausSurf [30]	30.05	0.920	0.183	25.09	0.753	0.212	27.57	0.837	0.198
Ours	29.79	0.924	0.168	24.47	0.724	0.250	27.13	0.824	0.209

- Followed experimental setup in 3DGS
  - Conduct validation on the Mip-NeRF360 dataset
  - Delivers competitive results in novel view rendering

Ablation Study

Method	Chamfer Distance ↓	Precision ↓	Completeness ↓
w/o Resampling	0.57	0.48	0.92
w/o L_{normal}	0.53	0.40	0.68
w/o $L_{normal-G}$	0.54	0.39	0.74
w/o L_{mv}	0.63	0.43	0.85
w/o L_{mv-G}	0.51	0.47	0.73
Full model	0.53	0.37	0.73

Conclusions

Scene Rendering:

- Enhanced Gaussian Representation:
  - Spherical Harmonics for both color & opacity
  - Captures spatial-dependent texture/material properties
- Geometry-Texture Decoupling:
  - Complex appearances via textures
  - Fewer Gaussians needed
- Memory Efficiency:
  - High fidelity without increasing Gaussian count

Scene Modeling:

- Geometrically-Precise 3D Gaussians
  - Centered on latent surface
  - Dense & uniformly distributed
- Novel Optimization Framework
  - Multi-scale (pixel + Gaussian-level) constraints
  - Opacity regularization for pruning
- State-of-the-Art Performance
  - Superior geometric consistency
  - Preserved photorealistic rendering

Future Directions:

- Enhanced Rendering Quality
  - Surface-constrained Gaussians degrades rendering fidelity
  - Relax Gaussian location to preserve:
    - *Complex view-dependent effects*
    - *Material reflectance properties*
    - *High-frequency details*
- Dynamic Scene Extension
  - Extract 3D Gaussian flow for:
    - *Temporal consistency for animation*
    - *Motion tracking & prediction*
    - *Editable dynamic representations*
- Gaussians as Primitives (GSS)
  - Establish 3D Gaussians as foundational elements for:
    - *Scene modeling & editing*
    - *Geometric skeletonization*
    - *Adaptive level-of-detail*

QUESTIONS?

Abstract

- 3D Gaussian Splatting (3DGS) enables efficient novel view synthesis through a flexible representation, but it struggles to capture fine spatial textures and accurately reconstruct scene geometry. This leads to the need for a large number of Gaussians to reproduce detailed appearances, and their positions often deviate from the true surface.
- In the first half of this talk, we present Textured-GS, a novel rendering method that augments 3DGS by introducing spatially varying color and opacity via Spherical Harmonics (SH). This enhancement allows each Gaussian to express richer surface details, improving visual quality without increasing Gaussian count. We integrate Textured-GS into the Mini-Splatting pipeline and demonstrate consistent improvements in rendering fidelity across several real-world datasets.
- In the second half, we explore using Gaussians as primitives for scene modeling. Inspired by Point Set Surface methods, we propose Gaussian Set Surface Reconstruction (GSSR), which distributes Gaussians evenly along the latent surface while aligning their dominant axes with surface normals. GSSR achieves fine-grained alignment through a combination of pixel-level and Gaussian-level single-view normal consistency and multi-view photometric consistency, optimizing both local accuracy and global coherence. Our results show significantly improved geometric precision, enabling intuitive scene editing and efficient generation of Gaussian-based 3D environments.