UniTriSplat: A Unified 3D Gaussian Splatting Framework with Uniform Spherical Rasterization for Universal Cameras

Yipeng Zhu[1], Huajian Huang[2†], Tristan Braud[1], and Sai-Kit Yeung[1]

[1] The Hong Kong University of Science and Technology, Hong Kong, China
[2] Beijing Institute of Technology, Beijing, China
[†]Corresponding author

Abstract. Existing 3D Gaussian Splatting (3DGS) frameworks rely on camera-specific rasterization, suffering from inconsistent solid-angle sampling and degraded performance across heterogeneous camera models (e.g., perspective, fisheye, omnidirectional). To address this limitation, we propose UniTriSplat, a unified 3DGS framework for universal cameras that reformulates Gaussian splatting on the unit sphere via HEALPix discretization. Leveraging the equal-area property of HEALPix, we construct a spherical sampling grid aligned with the angular resolution of input images. We derive the forward rendering and gradient propagation of Gaussians directly in the spherical radian domain, yielding uniform optimization behavior from narrow-FoV images to full 360-degree panoramas. To enhance perceptual reconstruction quality, we additionally introduce a HEALPix-aware SSIM loss that respects spherical neighborhood structure. Extensive experiments across diverse camera models demonstrate that UniTriSplat consistently improves cross-camera generalization while preserving geometric fidelity and rendering quality. Project page: https://yipengzhu0809.github.io/UniTriSplat/

Keywords: 3D Gaussian Splatting · Omnidirectional Reconstruction · Novel View Synthesis

1 Introduction

3D Gaussian Splatting [23] (3DGS) achieves high-fidelity 3D reconstruction by employing explicit primitives and a differentiable tile-based rasterization pipeline. However, increasing demand for wide-field-of-view (FoV) perception in urban-scale digital twins [38,42] and immersive virtual reality [21,34] requires a shift from perspective rendering to omnidirectional synthesis. This transition necessitates that 3DGS natively supports heterogeneous imaging models across arbitrary-FoV configurations, including fisheye lenses and 360° imagery.

Despite its efficiency, adapting 3DGS to diverse camera geometries remains challenging. Existing frameworks typically rely on camera-specific projection functions [29] and their corresponding Jacobian matrices [43], which leads to

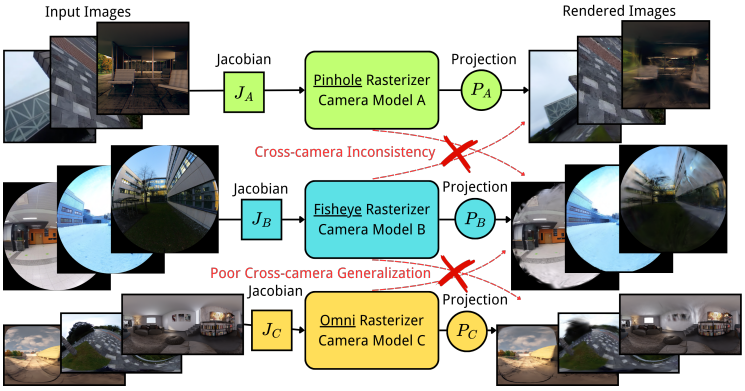

Fig. 1: Limitations of camera-specific 3DGS pipelines. Existing methods require separate rasterizers and projection models for different camera types, resulting in fragmented implementations and poor cross-camera generalization.

fragmented implementations and limits their versatility. Crucially, the tight coupling between the rasterizer and the pinhole model prevents the learned Gaussian scene from maintaining geometric consistency when rendered across different camera types, thereby compromising cross-camera generalization (Fig. 1). An alternative is to map diverse camera models onto a unified intermediate space, such as equirectangular panoramas [12, 19, 28]. Nevertheless, this approach is limited by the requirement for 360° coverage and struggles with partial-FoV inputs. Furthermore, the non-uniform pixel density of equirectangular projection induces imbalanced sampling between polar and equatorial zones [20], resulting in biased gradient propagation and suboptimal convergence (Fig. 3).

Fundamentally, all central camera models can be unified through a shared spherical geometry. In this work, we exploit this principle to develop **UniTriSplat**, a unified 3DGS framework featuring a camera-agnostic spherical rasterizer. By projecting 3D Gaussians directly onto a standardized spherical grid, UniTriSplat decouples the rendering process from camera-specific projection, enabling consistent synthesis across arbitrary FoVs and lens models. Specifically, we adopt HEALPix [11] (Hierarchical Equal Area iso-Latitude Pixelization) as the underlying spherical discretization scheme. HEALPix partitions the sphere into a hierarchical grid of equal-area pixels, inherently eliminating the sampling bias present in equirectangular representations [3]. To accommodate varying camera configurations, we adjust the HEALPix resolution by matching the grid's solid angle to the local angular resolution of the input images. Building upon this representation, we reformulate the forward and backward rasterization of 3DGS within the arc-length coordinate system on the unit sphere, and employ HEALPix for uniform spherical tessellation. Subsequently, we redesign the tile query, depth sorting, and density control modules to operate natively on the HEALPix grid. In particular, the tile query supports both sequential scanning and quadtree traversal via different indexing schemes, integrating depth sort-

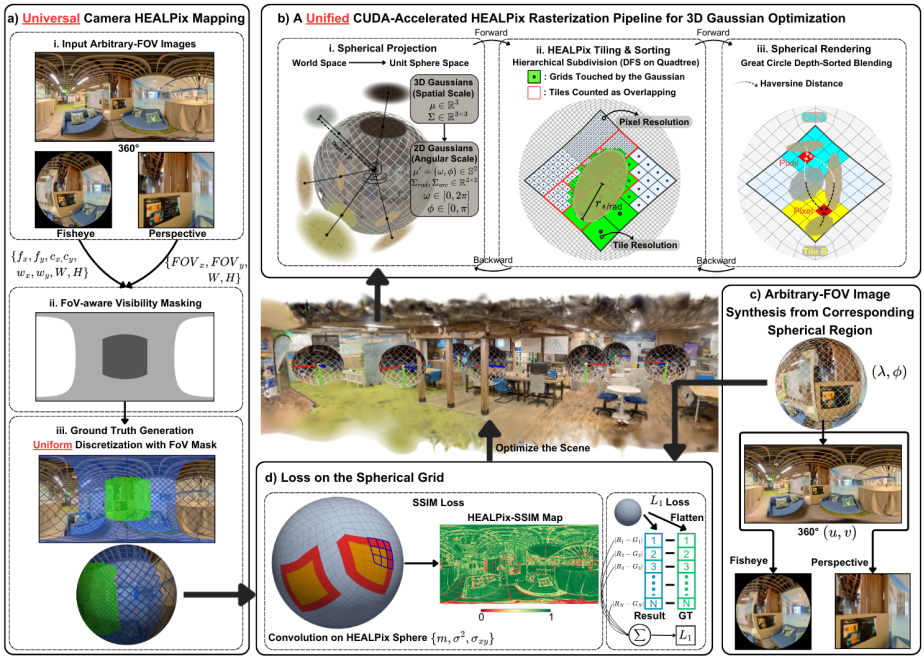

Fig. 2: Overview of UniTriSplat. (a) Unified mapping from heterogeneous camera models to the HEALPix domain with FoV-specific masking. (b) Spherical rasterization within the radian space: projection, tile query, and depth-sorted blending. (c) Sampling the rendered HEALPix grids into 2D images for evaluation. (d) Optimization guided by the HEALPix-aware SSIM and L_1 losses.

ing by radial distance and density control driven by angular-gradient thresholds. During rendering, the 3D Gaussian scene is rasterized onto a spherical domain corresponding to the target FoV, followed by resampling onto the image plane of the target camera. To further extend spherical perceptual quality [15] to HEALPix, we introduce a HEALPix-aware SSIM (Structural Similarity Index) loss that computes structural similarity over spherical neighborhoods, improving reconstruction fidelity in distorted regions. The pipeline of UniTriSplat is illustrated in Fig. 2. Our contributions can be summarized as follows:

- Proposing UniTriSplat, the first 3DGS framework supporting arbitrary camera models and FoVs via a unified HEALPix-based rasterization pipeline.
- Deriving the gradients for splatting 3D Gaussians onto HEALPix spherical grids and developing a custom rasterizer for effective training and rendering.
- Designing a HEALPix-SSIM loss for spherical geometry in structural similarity computation, facilitating spatially uniform Gaussian optimization.
- Comprehensive experiments demonstrating state-of-the-art spherical reconstruction metrics, competitive planar metrics, improved cross-camera generalization, and stable reconstruction quality for arbitrary-FoV inputs.

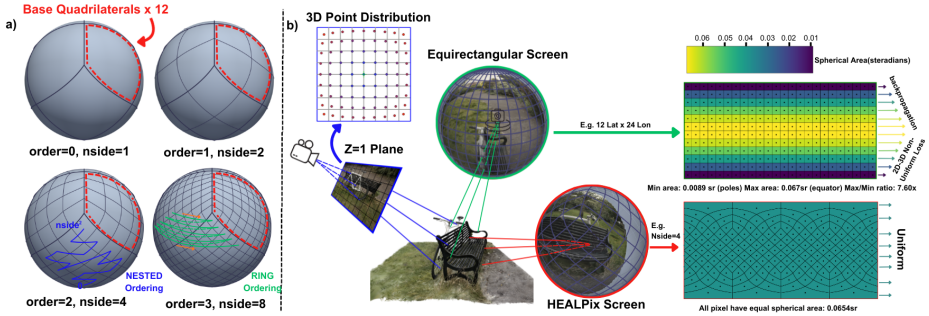

Fig. 3: HEALPix Tessellation and Uniform Rasterization. (a) Hierarchical structure and indexing. (b) Equirectangular projection shows 7.6× pixel area variation, while HEALPix ensures uniform spherical sampling.

2 Related Work

2.1 3D Gaussian Splatting Beyond Perspective Projection

Wide-FoV cameras are essential for VR, robotics, and autonomous systems, yet vanilla 3DGS relies on perspective projection, introducing severe distortions for such inputs (Fig. 3). Recent works [28, 29] derive camera-specific Jacobians for equidistant and equirectangular projections. Others employ tangent-plane projections [1, 19, 25], splatting Gaussians onto local tangent planes then warping to the sphere, though this introduces linearization artifacts at patch boundaries. Several works [19, 27, 37] redesign the EWA affine approximation [43] for wide-FoV imaging; notably, Wu et al. [37] replace it with the Unscented Transform for more accurate nonlinear projection under arbitrary distortions. Ito et al. [20] apply distortion-aware reweighting to mitigate polar oversampling, while Shin et al. [32] jointly optimize with dual-fisheye calibration for seamless 360° renderings. Deng et al. [10] propose a cross-FoV method that extends perspective projection to a cubemap representation, enabling the processing of large-FoV images. However, all these methods require camera-specific modifications and remain confined to equirectangular or planar rasterization, leaving nonuniform sampling unaddressed. Our work performs rasterization directly on the sphere, providing a camera-agnostic framework with inherently uniform coverage.

2.2 Spherical Image Representation

Projecting visual signals onto a sphere aligns naturally with omnidirectional perception. Since omnidirectional imaging was widely adopted [4], equirectangular projection [33] has become popular for its simplicity, supporting various vision tasks [8,14,17,18,41], but suffers from severe polar oversampling. To address this, uniform tessellation schemes have been explored [6], including cubemap [12], icosahedral grids [31], and quasi-uniform Yin–Yang grid [22]. Choi et al. [7] use Yin-Yang grid for balanced ray sampling in scene reconstruction, while Lee et

al. [26] leverage it for quasi-uniform patch decomposition enabling pretrained networks on 360° images. We adopt HEALPix as our spherical representation (Fig. 3). Unlike quasi-uniform schemes, HEALPix guarantees strictly equal-area pixels with a hierarchical multi-resolution structure. Carlsson et al. [3] exploit this hierarchy for efficient spherical self-attention, while Krachmalnicoff et al. [24] and Cheng et al. [5] provide foundational spherical CNN implementations. Despite existing efforts for HEALPix-based applications [5], HEALPix's irregular geometry has limitations in 3D tasks. We present the first CUDA implementation integrating HEALPix directly into the 3DGS rasterization pipeline.

3 HEALPix-based 3D Gaussian Rasterization

In this paper, we present a unified 3DGS framework that rasterizes directly on the HEALPix spherical grid. The following sections detail HEALPix adaptation for visual representation (Sec. 3.1), latitude-aware spherical splatting with gradient derivations (Sec. 3.2), efficient tile query algorithms exploiting HEALPix's indexing (Sec. 3.3), and image synthesis for arbitrary camera models (Sec. 3.4).

3.1 Adapting HEALPix for Visual Representation

Our rasterizer operates on the HEALPix grid, which partitions the unit sphere into 12 base quadrilaterals, each recursively subdivided with resolution parameter N_{side} (power of two), yielding $N_{\text{pix}} = 12N_{\text{side}}^2$ equal-area pixels. This equal-area property ensures uniform solid angle $\Omega_{\text{pix}} = 4\pi/N_{\text{pix}}$ per pixel, providing balanced gradient contributions during backpropagation.

To dynamically match the input angular resolution, we compute the adaptive HEALPix resolution $N_{\text{side}}^* = \sqrt{4\pi WH/(12\Omega_{\text{in}})}$ and round it to the nearest power of two. The solid angle Ω_{in} is defined as 4π for Equirectangular, $2\pi(1-\cos\theta)$ for Fisheye, and $4\arcsin(\sin\theta_x \sin\theta_y)$ for Perspective, where $\theta, \theta_x, \theta_y$ denote the respective half-FoVs. For a camera model $\mathcal{C}$, let $\mathcal{D}_\mathcal{C} \subseteq [0, W) \times [0, H)$ denote its valid image domain. We define the image-to-sphere mapping $\mathcal{P}_\mathcal{C} : \mathcal{D}_\mathcal{C} \to \mathbb{S}^2$, which maps each valid image pixel (u, v) to its spherical direction (ω, ϕ). The corresponding visible spherical region is $\mathcal{R}_\mathcal{C} = \mathcal{P}_\mathcal{C}(\mathcal{D}_\mathcal{C})$, with the HEALPix visibility mask $\mathcal{M}_\mathcal{C}(p) = \mathbf{1}[(\omega_p, \phi_p) \in \mathcal{R}_\mathcal{C}]$. Ground-truth images are resampled onto the adaptive-resolution HEALPix grid within $\mathcal{R}_\mathcal{C}$, enabling unified supervision across camera models, as shown in Fig. 2(a).

HEALPix provides two indexing schemes: RING ordering arranges pixels along iso-latitude rings, while NESTED ordering follows the hierarchical quadtree structure, as illustrated in Fig. 3. We adopt NESTED indexing with (x, y, f) parameterization, where $x, y \in [0, N_{\text{side}})$ are local coordinates within base quadrilateral $f \in [0, 12)$. This maps naturally to GPU parallelization: each base quadrilateral is partitioned into $\lceil N_{\text{side}}/B \rceil^2$ tiles of $B \times B$ threads, yielding $12\lceil N_{\text{side}}/B \rceil^2$ parallel tiles with hierarchical spatial coherence for efficient memory access.

3.2 Latitude-Aware Splatting on the Unit Sphere

We now formulate how 3D Gaussians are splatted onto the HEALPix spherical screen space, parameterized in radians rather than pixels. Let $\mathbf{t} = \mathbf{W}\boldsymbol{\mu} + \mathbf{b}$ denote the camera-space position of a Gaussian center $\boldsymbol{\mu}$, where $\mathbf{W} \in \mathbb{R}^{3\times3}$ and $\mathbf{b}$ are the rotation and translation of the view matrix.

Forward Projection. The spherical projection $\Pi : \mathbb{R}^3 \to \mathbb{S}^2$ maps $\mathbf{t} = (t_x, t_y, t_z)^\top$ to longitude-latitude coordinates:

$$\boldsymbol{\mu}' = \Pi(\mathbf{t}) = (\omega, \phi)^\top, \quad \omega = \operatorname{atan2}(t_x, t_z), \quad \phi = \operatorname{atan2}\left(t_y, \sqrt{t_x^2 + t_z^2}\right), \tag{1}$$

with radial depth $r = \|\mathbf{t}\|$ for depth sorting. The 3D covariance $\boldsymbol{\Sigma}_{\text{3D}}$ is projected to radian-space covariance $\boldsymbol{\Sigma}_{\text{rad}}$ via the Jacobian of Π composed with the rotation, characterizing the Gaussian's angular extent for spherical rasterization.

Since the HEALPix screen measures arc length rather than angular displacement, and equal angular increments yield different arc lengths at different latitudes, we apply latitude-dependent scaling:

$$\boldsymbol{\Sigma}_{\text{arc}} = \mathbf{S}_\phi \boldsymbol{\Sigma}_{\text{rad}} \mathbf{S}_\phi^\top, \quad \mathbf{S}_\phi = \operatorname{diag}(\cos\phi, 1). \tag{2}$$

Conic parameters are computed from $\boldsymbol{\Sigma}_{\text{arc}}^{-1}$, and the spherical bounding radius $r_s = 3\sqrt{\lambda_{\max}(\boldsymbol{\Sigma}_{\text{arc}})}$ determines tile overlap.

Backward Propagation. Gradients propagate through this pipeline to update $\boldsymbol{\mu}$, scale $\mathbf{s}$, and rotation $\mathbf{q}$. Let $\mathcal{L}_X \coloneqq \frac{\partial \mathcal{L}}{\partial X}$.

Covariance gradients. Since $\mathbf{S}_\phi$ depends on ϕ, backpropagating through Eq. (2) yields:

$$\mathcal{L}_{\boldsymbol{\Sigma}_{\text{rad}}} = \mathbf{S}_\phi^\top \mathcal{L}_{\boldsymbol{\Sigma}_{\text{arc}}} \mathbf{S}_\phi, \quad \mathcal{L}_\phi\big|_{\mathbf{S}_\phi} = -\sin\phi \left(2\cos\phi\, \sigma_{\omega\omega}\, \mathcal{L}_{\sigma_{\omega\omega}^{\text{arc}}} + \sigma_{\omega\phi}\, \mathcal{L}_{\sigma_{\omega\phi}^{\text{arc}}}\right), \tag{3}$$

where $\boldsymbol{\Sigma}_{\text{rad}} = \left(\begin{smallmatrix} \sigma_{\omega\omega} & \sigma_{\omega\phi} \\ \sigma_{\omega\phi} & \sigma_{\phi\phi} \end{smallmatrix}\right)$. The first term is the standard covariance gradient; the second captures how projected Gaussian shape varies with latitude. Gradients then propagate through $\boldsymbol{\Sigma}_{\text{rad}} = \mathbf{T}\boldsymbol{\Sigma}_{\text{3D}}\mathbf{T}^\top$ to update $\mathbf{s}$ and $\mathbf{q}$.

Position gradients. Beyond standard paths through $\mathbf{J}$ and $\boldsymbol{\mu}'$, spherical projection introduces an additional path through $\mathbf{S}_\phi$:

$$\left.\frac{\partial \mathcal{L}}{\partial \mathbf{t}}\right|_{\mathbf{J}} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{\Sigma}_{\text{rad}}} \frac{\partial \boldsymbol{\Sigma}_{\text{rad}}}{\partial \mathbf{J}} \frac{\partial \mathbf{J}}{\partial \mathbf{t}}, \quad \left.\frac{\partial \mathcal{L}}{\partial \mathbf{t}}\right|_{\mathbf{S}_\phi} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{\Sigma}_{\text{arc}}} \frac{\partial \boldsymbol{\Sigma}_{\text{arc}}}{\partial \mathbf{S}_\phi} \frac{\partial \mathbf{S}_\phi}{\partial \phi} \frac{\partial \phi}{\partial \mathbf{t}}. \tag{4}$$

The second term is a geometry-dependent gradient path unique to spherical projection. The total gradient, including $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}'}\mathbf{J}$, transforms to world space via $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \mathbf{W}^\top \frac{\partial \mathcal{L}}{\partial \mathbf{t}}$. For depth supervision, we use inverse radial depth r^{-1}, whose gradient is $\frac{\partial r^{-1}}{\partial \mathbf{t}} = -\frac{\mathbf{t}}{r^3}$.

3.3 Efficient Tile Query on HEALPix

Given the projected spherical covariances, we perform tile-based culling to restrict each Gaussian's contribution to its intersecting HEALPix regions, ensuring

high-performance rendering. In vanilla 3DGS, each 2D Gaussian's rectangular footprint is intersected with a regular tile grid in $O(1)$ time. Since HEALPix pixels do not form a rectangular lattice, we develop two alternative query strategies exploiting its NESTED and RING indexing, which are illustrated in Fig. 8.

NESTED Quadtree Traversal. HEALPix's NESTED indexing forms a quadtree hierarchy. Given a spherical disc with center (ω, ϕ) and radius r_s, we perform depth-first search (DFS) from the 12 base quadrilaterals, pruning subtrees whose centers lie entirely outside the disc. Let N^q_{side} denote the query resolution, P the number of Gaussians, and K the average touched pixels per Gaussian. This achieves $O(K + \log N^q_{\text{side}})$ time per Gaussian, but requires $O(P \cdot \log N^q_{\text{side}})$ stack memory and suffers from irregular memory access on the GPU.

RING Sequential Scan. HEALPix's RING indexing arranges pixels in isolatitude rings, each characterized by $z = \cos\vartheta$ where $\vartheta \in [0, \pi]$ is the colatitude measured from the north pole. We compute the z-range intersecting the disc, then analytically solve for the longitude range within each ring:

$$\cos \Delta\omega = \frac{\cos r_s - z_c \cdot z_{\text{ring}}}{\sqrt{1 - z_c^2} \cdot \sqrt{1 - z_{\text{ring}}^2}}, \tag{5}$$

where $z_c = \sin\phi_c$ is the z-coordinate of the disc center at latitude ϕ_c, and $\Delta\omega$ is the half-width in longitude. This yields $O(K)$ time with $O(1)$ memory. Both schemes capture distinct trade-offs: NESTED prioritizes precision, while RING achieves a 1.7× speedup at the expense of quality (see Sec. 5.3).

3.4 Rendering on Spherical Grid and Image Synthesis

To ensure correct visibility across the full sphere, we perform per-tile alpha compositing by sorting Gaussians based on their radial distance $r = |\mathbf{t}|$ from the camera origin, rather than conventional planar z-depth. Each HEALPix pixel center is determined by its NESTED index (x, y, f), which maps to spherical coordinates $(\omega_{\text{pix}}, \phi_{\text{pix}})$ via the HEALPix indexing scheme. During per-pixel blending, we compute the great-circle distance d_{gc} between each Gaussian center (ω, ϕ) and the HEALPix pixel center using the Haversine formula [35], as shown in Fig. 2 (b), then decompose it into local tangent-plane offsets (d_x, d_y) via the spherical bearing angle. The per-pixel blending weight is computed as $\alpha = o \cdot \exp\big(-\frac{1}{2}\mathbf{d}^\top \boldsymbol{\Sigma}^{-1}_{\text{arc}}\mathbf{d}\big)$, where o is the Gaussian opacity and $\mathbf{d} = (d_x, d_y)^\top$.

Consequently, the HEALPix sphere is rendered through front-to-back α-blending. For a HEALPix pixel p, let $\mathcal{S}_p$ denote the set of overlapping Gaussians, sorted by increasing radial depth r_i. Its rendered color is

$$I_{\text{H}}(p) = \sum_{i \in \mathcal{S}_p} \mathbf{c}_i \, \alpha_i(p) \prod_{\substack{j \in \mathcal{S}_p \\ r_j < r_i}} (1 - \alpha_j(p)), \tag{6}$$

where $\mathbf{c}_i$ is the view-dependent color of Gaussian i, and $\alpha_i(p)$ is the blending weight, evaluated for Gaussian i at pixel p. Using the camera mapping $\mathcal{P}_\mathcal{C}$ and

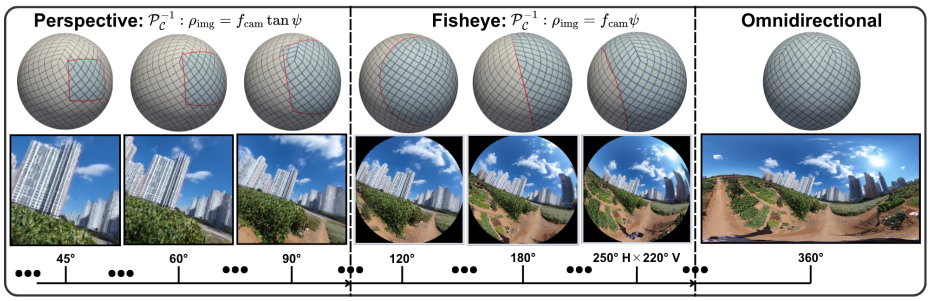

Fig. 4: Rendering across camera models and FoVs. UniTriSplat synthesizes perspective, fisheye, and omnidirectional images by changing only the image-to-sphere mapping and valid FoV mask. The underlying spherical rasterization remains unchanged across camera configurations.

valid image domain $\mathcal{D}_{\mathcal{C}}$ defined in Sec. 3.1, the final image is obtained by interpolating neighboring HEALPix samples:

$$I_{\mathcal{C}}(u,v) = \text{Interp}_{\text{H}}(I_{\text{H}}, \mathcal{P}_{\mathcal{C}}(u,v)), \qquad (u,v) \in \mathcal{D}_{\mathcal{C}}. \tag{7}$$

For partial-FoV cameras, rendering and optimization are restricted to HEALPix pixels satisfying $\mathcal{M}_{\mathcal{C}}(p) = 1$, avoiding computation outside the visible spherical region $\mathcal{R}_{\mathcal{C}}$. Figure 4 illustrates image synthesis across different camera models and FoVs while keeping the underlying HEALPix rasterization unchanged.

4 Training on the HEALPix Grid

Having described the spherical rasterization pipeline, we now detail the training procedure. Two components require adaptation: the structure-aware loss must respect HEALPix's non-planar topology (Sec. 4.1), and the density control thresholds must operate in radian space rather than pixel counts (Sec. 4.2).

4.1 HEALPix-SSIM Loss

The original 3DGS combines $\mathcal{L}_1$ photometric loss and SSIM [36] structure-aware loss. However, standard SSIM assumes a regular 2D grid, making it inapplicable to HEALPix tessellation. We propose HEALPix-SSIM (HSSIM), a structural similarity metric for the HEALPix grid with a fully differentiable CUDA implementation. Within each base quadrilateral, the NESTED (x, y, f) parameterization approximates a local 2D grid, enabling separable convolution to compute local statistics μ, σ^2, and covariance σ_{12} between rendered and ground-truth HEALPix images for the SSIM index. At quadrilateral boundaries, topological singularities cause some corner pixels to have only 7 neighbors; we detect such cases and renormalize kernel weights accordingly. Furthermore, since equal pixel offsets correspond to different angular distances at varying latitudes, we

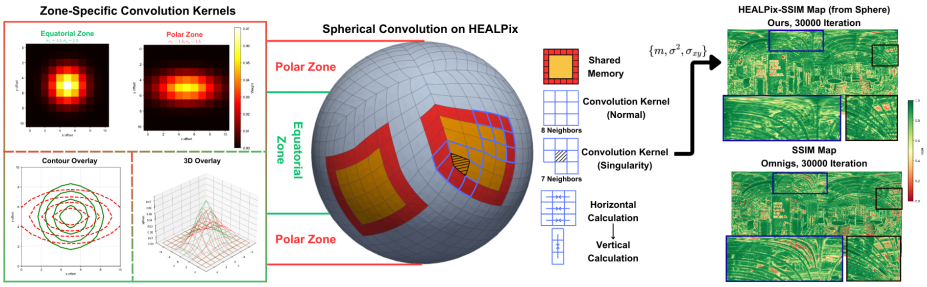

Fig. 5: HEALPix-SSIM. Zone-specific convolution kernels (left), spherical convolution with shared memory and singularity-aware boundary handling (middle), and the comparison of the resulting HSSIM map and SSIM map (right).

employ zone-specific Gaussian kernels: isotropic for the equatorial zone where pixels are roughly square, and anisotropic for polar zones where pixel geometry is longitudinally compressed (Fig. 5). Our CUDA implementation leverages shared memory with cross-face padding and caches intermediate derivatives for efficient gradient propagation via transposed convolution. The final training loss is $\mathcal{L} = (1-\lambda)\mathcal{L}_1^{\mathrm{H}} + \lambda\mathcal{L}_{\text{H-SSIM}}$, where $\mathcal{L}_1^{\mathrm{H}}$ is the per-pixel ℓ_1 distance on the HEALPix grid and λ balances photometric and perceptual terms.

4.2 Radian-Space Density Control

Adaptive density control in 3DGS clones or splits high-gradient Gaussians and prunes low-opacity or oversized ones. Directly applying pixel-based thresholds to HEALPix is problematic because spherical projection maps Gaussians to angular coordinates (ω, ϕ), requiring size-based criteria to operate in radian-space. For each visible Gaussian, the spherical projection produces a 2D covariance Σ_{arc} on the sphere, from which we derive an angular radius r_s representing the Gaussian's spherical footprint. We maintain a per-Gaussian buffer that records the maximum r_s observed across training iterations, updated via $r_{s,\max} \leftarrow \max(r_{s,\max}, r_s)$ whenever the Gaussian is visible. This radian-space tracking replaces pixel-based screen size and ensures scale comparisons remain consistent on the HEALPix grid. The pruning threshold for oversized Gaussians is similarly converted from pixels to radians based on the HEALPix resolution N_{side}.

In order to reduce the number of primitives and time consumption, we also explore multi-view consistent pruning [30], which removes Gaussians contributing minimally to reconstruction across sampled views. This is an exploratory configuration that is not used in the final framework due to quality degradation. However, ablation results confirm that it provides a speed-quality trade-off.

5 Experiment

In this section, we evaluate UniTriSplat's performance in terms of multi-FoV reconstruction fidelity, cross-camera generalization, and computational efficiency.

Table 1: Quantitative Results of Multi-FoV Evaluation. We evaluate UniTriSplat on perspective (Pp.), fisheye (Fe.), and omnidirectional (Omni.) datasets. FoV denotes horizontal × vertical field of view. Training time is reported for 30,000 iterations. Blue and light blue indicate the best and second-best results, respectively.

Dataset	FoV	Method	PSNR (dB)↑	SSIM↑	HSSIM↑	LPIPS↓	Train (min)↓
Pp. Mip-NeRF 360	52° × 36°– 68° × 47°	3DGS	25.59	0.801	0.726	0.258	**25.2**
		OP43DGS	**28.03**	**0.879**	0.741	**0.231**	45.6
		Ours	27.57	0.848	**0.806**	0.239	47.2
Fe. ScanNet++	127° × 85°	OP43DGS	21.82	0.728	0.650	0.355	**30.5**
		Fisheye-GS	29.13	0.920	0.801	0.186	43.5
		Ours	**29.75**	**0.928**	**0.883**	**0.179**	46.4
Fe. FIORD	120° × 120°– 170° × 170°	OP43DGS	21.64	0.717	0.634	0.321	50.1
		Fisheye-GS	21.15	0.704	0.615	0.350	**25.6**
		Ours	**24.27**	**0.791**	**0.748**	**0.279**	54.2
Omni. Ricoh360	360° × 180°	ODGS	21.89	0.813	0.786	0.246	42.8
		OP43DGS	23.07	0.833	0.809	0.221	71.1
		OmniGS	**25.34**	**0.872**	0.843	**0.195**	**18.1**
		Ours	24.70	0.864	**0.863**	0.212	29.0
Omni. OmniBlender	360° × 180°	ODGS	29.81	0.876	0.859	0.239	40.5
		OP43DGS	32.01	0.883	0.865	0.182	91.8
		OmniGS	**33.21**	**0.919**	0.891	**0.166**	**19.8**
		Ours	32.57	0.905	**0.900**	0.172	28.2
Omni. 360Roam	360° × 180°	ODGS	20.72	0.722	0.685	0.383	247.6
		OP43DGS	21.04	0.732	0.686	0.376	330.7
		OmniGS	21.48	0.741	0.710	0.369	122.6
		Ours	**21.82**	**0.747**	**0.724**	**0.353**	**102.1**

We compare against 3DGS and OP43DGS [19] on perspective images, Fisheye-GS [29] and OP43DGS on fisheye images, and ODGS [25], OP43DGS, and OmniGS [28] on omnidirectional images. We conduct three sets of experiments: multi-FoV evaluation, cross-camera validation, and an ablation study.

5.1 Implementation and Experiment Setup

Implementation. UniTriSplat is built upon standard 3DGS with a redesigned rasterization pipeline and the HEALPix-aware SSIM module. While baseline methods retain their default training parameters, we adapt the densification thresholds, learning rates, and decay schedules for UniTriSplat to accommodate the unique gradient distributions inherent in the spherical domain (Sec. 3.2). We evaluate UniTriSplat using benchmarks covering multiple input FoV regimes: Mip-NeRF 360 [2] for perspective; ScanNet++ [39] and FIORD [13] for fisheye; and 360Roam [16], OmniBlender [7], and Ricoh360 [7] for omnidirectional FoVs. All models are trained on a single NVIDIA RTX 4090D GPU. We report standard metrics including PSNR, SSIM, and LPIPS [40]. As traditional planar metrics can be biased against spherical representations due to resampling, we project baseline results onto the HEALPix grid to ensure a fair comparison via HEALPix-SSIM (HSSIM) (Sec. 4.1).

Multi-FoV evaluation. To assess UniTriSplat across diverse camera models, we evaluate it over a broad range of FoVs, including perspective, fisheye,

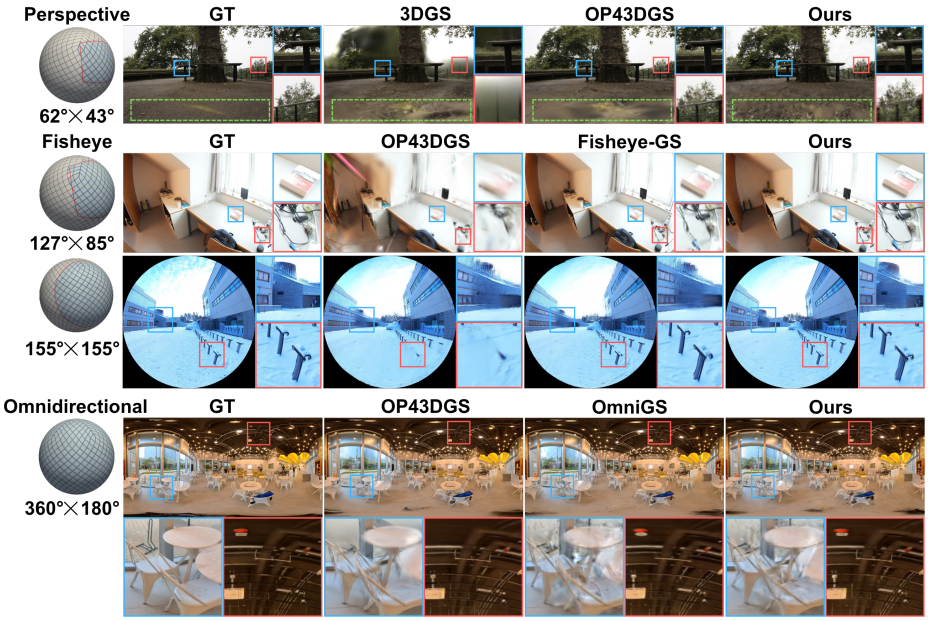

Fig. 6: Qualitative Results of Multi-FoV Evaluation. We compare UniTriSplat with camera-specific baselines across diverse camera models and FoVs. From top to bottom, the four rows show results on selected scenes from Mip-NeRF 360, ScanNet++, FIORD, and 360Roam, respectively. UniTriSplat maintains stable rendering quality, with better performance on fisheye and omnidirectional images. Additional results are provided in the supplementary material.

and omnidirectional imaging. Each camera setting is evaluated on representative benchmarks against the corresponding projection-specific 3DGS baselines. This evaluates whether unified spherical rasterization maintains reconstruction quality across different projection distortion and spherical coverage.

Cross-camera validation. To verify that the learned 3D Gaussians maintain consistency across diverse projections, we train on omnidirectional data and render perspective and fisheye views at multiple FoVs using our HEALPix rasterizer. We define the original camera orientation as the front direction. For Ricoh360, OmniBlender, and 360Roam, respectively, we render front-facing fisheye views with FoVs of 120°, 180°, and 240°, and perspective views with FoVs of 45°, 60°, and 90° facing backward, rightward, and leftward. Ground-truth views are generated through sphere-to-plane sampling.

5.2 Results, Evaluation and Discussion

Multi-FoV Evaluation. The quantitative results in Tab. 1 demonstrate that UniTriSplat maintains stable and competitive performance across multiple camera models and FoVs. On perspective inputs, UniTriSplat outperforms vanilla

Table 2: Quantitative Results of Cross-camera Validation. Given Gaussian scenes reconstructed from omnidirectional inputs, we evaluate perspective (Pp.) and fisheye (Fe.) rendering using camera-specific rasterizers and our unified rasterizer. Blue and light blue indicate the best and second-best results. Our method demonstrates consistent cross-camera generalization across diverse projection models.

Dataset	Method	Pp. from 3DGS			Pp. from Ours			Fe. from OP43DGS			Fe. from Ours		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Ricoh360	ODGS	12.27	0.502	0.704	17.59	0.602	0.511	19.32	0.647	0.318	22.52	0.707	0.294
	OP43DGS	12.71	0.487	0.787	18.81	0.693	0.440	**25.69**	**0.881**	**0.190**	19.60	0.639	0.302
	OmniGS	**22.67**	**0.804**	**0.165**	22.77	0.724	0.304	24.17	0.783	0.252	21.08	0.684	0.335
	Ours	16.30	0.521	0.681	**29.01**	**0.864**	**0.144**	17.95	0.581	0.408	**27.66**	**0.893**	**0.147**
OmniBlender	ODGS	15.30	0.516	0.609	20.40	0.623	0.379	21.70	0.724	0.205	22.98	0.811	0.315
	OP43DGS	15.63	0.560	0.566	20.76	0.646	0.363	29.51	0.880	0.137	20.01	0.631	0.326
	OmniGS	22.44	0.720	0.313	21.18	0.685	0.337	21.03	0.726	0.218	23.65	0.832	0.293
	Ours	**22.54**	**0.759**	**0.262**	**28.17**	**0.834**	**0.168**	**31.33**	**0.905**	**0.128**	**31.94**	**0.916**	**0.134**
360Roam	ODGS	18.26	0.709	0.391	23.65	0.759	0.333	20.28	0.650	0.426	20.74	0.683	0.404
	OP43DGS	18.50	0.720	0.390	23.61	0.766	0.326	21.44	0.701	0.368	21.09	0.689	0.385
	OmniGS	18.75	0.731	0.389	23.55	0.773	0.319	22.03	0.729	0.327	22.40	0.741	0.299
	Ours	**22.02**	**0.765**	**0.333**	**24.70**	**0.786**	**0.304**	**23.87**	**0.809**	**0.276**	**23.99**	**0.815**	**0.261**

3DGS [23] and remains competitive with OP43DGS [19]. While spherical-to-planar resampling slightly reduces the sharpness, UniTriSplat achieves the best HSSIM, indicating improved structural consistency. In addition, UniTriSplat performs particularly well on severely distorted fisheye inputs. On the ultra-wide-FoV FIORD [13] dataset, UniTriSplat improves PSNR by approximately 3 dB over the baselines, indicating that spherical-grid representations are inherently well suited to spherical imaging geometry. On ScanNet++ [39], UniTriSplat also consistently improves all reconstruction metrics over Fisheye-GS [29], while the larger gain on FIORD suggests that spherical rasterization is effective under more severe distortion. On omnidirectional benchmarks, UniTriSplat achieves state-of-the-art performance on 360Roam [16] while remaining competitive on Ricoh360 [7] and OmniBlender [7]. On the evaluated omnidirectional datasets, UniTriSplat maintains competitive training efficiency and records the shortest training time on 360Roam. Further efficiency analysis across input resolutions is provided in the supplementary material. Although OP43DGS supports multiple camera models, it is less competitive on the evaluated wide-FoV fisheye and omnidirectional datasets. This suggests that its camera-specific tangent-plane formulation is less robust under severe projection distortion. Moreover, UniTriSplat consistently achieves the best HSSIM across all datasets, demonstrating effective optimization over spherical geometry.

The qualitative results in Fig. 6 show that, by restricting HEALPix rasterization to the target FoV, UniTriSplat supports training and rendering across varying FoVs. On fisheye data, UniTriSplat produces fewer artifacts than the baselines, indicating that uniform spherical tessellation facilitates more accurate optimization under spherical projection. A similar effect is observed for omnidirectional images, where UniTriSplat better preserves polar details, such as the ceiling structures, which are degraded by the baselines. Although resampling from the sphere to the perspective plane slightly reduces sharpness, UniTriSplat

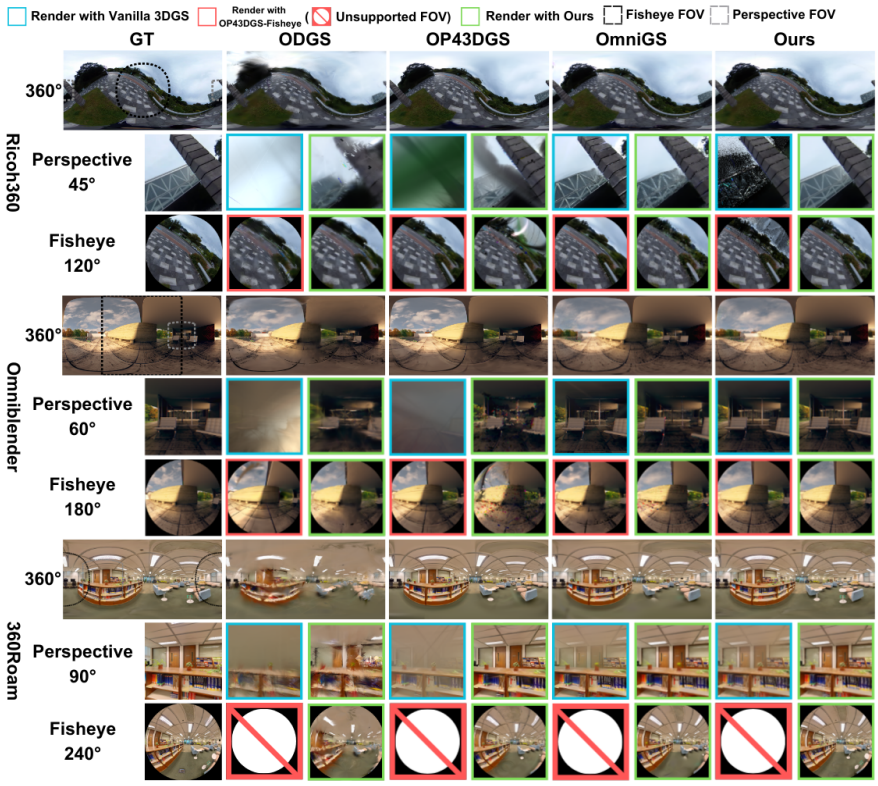

Fig. 7: The Qualitative Results of Cross-camera Validation. We evaluate renderings across different FoVs against GT images across three scenes (top to bottom: Pillar, Chair, Center). Our unified method maintains superior rendering quality and consistent geometric fidelity across heterogeneous cameras.

produces cleaner boundaries at narrow FoVs, with fewer spike-shaped artifacts, and maintains consistent fidelity across the image.

Cross-camera Validation. Table 2 reveals a strong interaction between scene optimization and the rasterizer used for image synthesis. Camera-specific rasterizers generally perform best when applied to Gaussian representations optimized with their native projection model, but their performance can degrade substantially when transferred across camera models. In contrast, UniTriSplat maintains more consistent synthesis quality across perspective and fisheye projections, particularly for scenes optimized using its unified spherical rasterizer. These results suggest that HEALPix-based optimization reduces projection-specific coupling between the learned Gaussians and the camera model.

UniTriSplat generally achieves higher perspective-rendering quality than the vanilla 3DGS across the evaluated scenes. In fisheye synthesis, our rasterizer achieves competitive results across the evaluated scenes, except when OP43DGS renders the Gaussian representation optimized with its own rasterization pipeline.

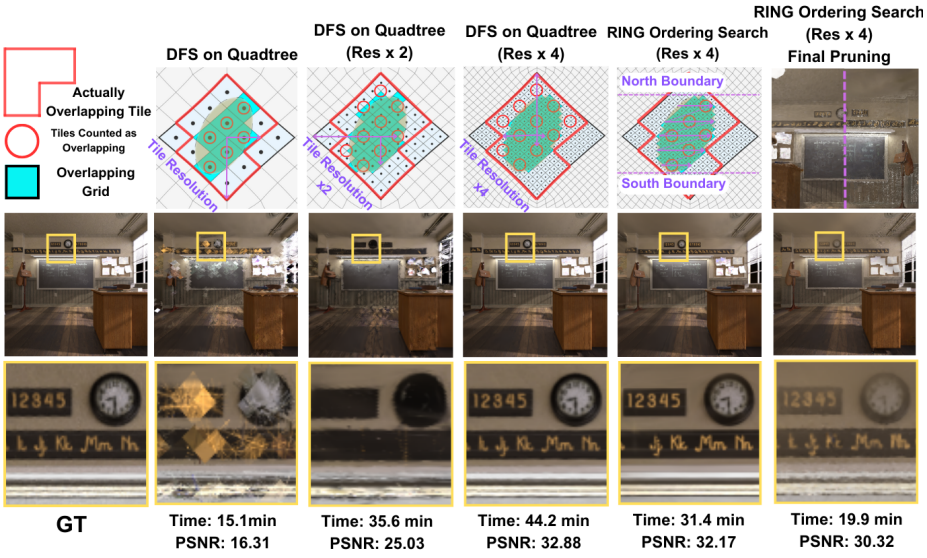

Fig. 8: The illustration of Tile Query Methods and the Qualitative Results of the Ablation Study. The 'classroom' scene is reported here. DFS stands for depth-first search [9]. Comparison of NESTED quadtree traversal and RING sequential scan demonstrates quality-efficiency trade-offs across different query resolutions.

Notably, while OP43DGS is constrained by specific camera models and FoV limitations such as a sub-180° range or the requirement for symmetric horizontal and vertical FoVs, UniTriSplat is camera-agnostic and handles arbitrary configurations. These results validate the effectiveness of our unified rasterization principle, demonstrating robust cross-camera generalization.

Qualitative results in Fig. 7 further substantiate these observations. Perspective crops synthesized by UniTriSplat exhibit higher fidelity with fewer geometric artifacts compared to vanilla 3DGS. A remaining challenge is observed in the Ricoh360 outdoor dataset, where OP43DGS rendering of our scenes shows artifacts in the sky regions, indicating a potential area for future optimization in handling unbounded backgrounds. Collectively, these results demonstrate that our unified spherical rasterization effectively decouples scene representation from camera-specific geometry, ensuring robust cross-camera generalization.

5.3 Ablation Study

We conduct ablation experiments on OmniBlender to analyze the contribution of each component in UniTriSplat. Table 3 summarizes the results, where R denotes RING Sequential Scan, D denotes NESTED Quadtree Traversal (Sec. 3.3), and $\times 1$, $\times 2$, $\times 4$ indicate search depths at $1\times$, $2\times$, and $4\times$ the original tile resolution, respectively. The first four rows of the left column reveal that quadtree search on denser HEALPix grids improves reconstruction quality at the expense of time

Table 3: Ablation study on the OmniBlender dataset. We evaluate the impact of tile query strategies and optimization components. **R** denotes RING Sequential Scan, **D** denotes NESTED Quadtree Traversal (Sec. 3.3), and $\times n$ indicates the search depth relative to the tile resolution. The highlighted configuration **R** $(\times 4)$ is selected as our default for its optimal performance-efficiency trade-off.

Tile Query	PSNR↑	SSIM↑	Time (min)	Optimization	PSNR↑	SSIM↑	Time (min)
R (×4)	32.57	0.905	28.2	Full Pipeline (Ref.)	32.57	0.905	28.2
D (×4)	33.64	0.916	49.1	w/o HSSIM	26.42	0.721	21.7
D (×2)	26.49	0.778	37.6	w/o Density Control	24.83	0.732	33.0
D (×1)	20.36	0.631	18.8	w/ Final Pruning	28.27	0.847	20.2

efficiency. RING Sequential Scan on denser grids achieves competitive metrics while maintaining high efficiency, and is therefore adopted as the default tile query module in UniTriSplat. As illustrated in Fig. 8, low-density deep search produces severe aliasing artifacts. Furthermore, removing either HSSIM or Density Control degrades performance across all reconstruction metrics, confirming the necessity of both modules. Final pruning substantially reduces the training time by removing redundant Gaussians. However, the resulting reduction in model capacity limits the achievable reconstruction quality, leading to lower metrics. We therefore exclude final pruning from our default configuration.

6 Conclusion

UniTriSplat introduces a unified 3DGS framework that reformulates rasterization on the equal-area HEALPix spherical tessellation, effectively mitigating the geometric distortions inherent in traditional planar projections. By decoupling spherical rasterization from camera-specific projection implementations, our approach enables model-agnostic synthesis for perspective, fisheye, and omnidirectional cameras within a single, consistent architecture. Leveraging a CUDA-accelerated rasterizer and a HEALPix-aware SSIM loss, UniTriSplat maintains stable reconstruction quality across diverse camera models and FoVs while enabling consistent cross-camera synthesis. Limitations include aliasing when resampling from HEALPix to planar images at lower resolutions, and minor projection errors from the EWA [43] splatting approximation. Future work will focus on refining rendering fidelity by addressing resampling aliasing and exploring advanced hierarchical acceleration structures tailored for the HEALPix topology.

Acknowledgements

This work was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKUST 16202323), an internal grant from HKUST (R9429), the Frontier Technology Research for Joint Institutes with Industry (FTRIS) Project at HKUST (Project No. FTRIS-25-056), and an internal grant from Beijing Institute of Technology.

References

1. Bai, J., Huang, L., Guo, J., Gong, W., Li, Y., Guo, Y.: 360-gs: Layout-guided panoramic gaussian splatting for indoor roaming. In: 2025 International Conference on 3D Vision (3DV). pp. 1042–1053. IEEE (2025)
2. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5470–5479 (2022)
3. Carlsson, O., Gerken, J.E., Linander, H., Spieß, H., Ohlsson, F., Petersson, C., Persson, D.: Heal-swin: A vision transformer on the sphere. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6067–6077 (2024)
4. Chen, S.E.: Quicktime vr: An image-based approach to virtual environment navigation. In: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques. pp. 29–38 (1995)
5. Cheng, X., Subramaniam, A., Wu, S., Brenowitz, N.: cuhpx: Gpu-accelerated differentiable spherical harmonic transforms on healpix grids. arXiv preprint arXiv:2510.01785 (2025)
6. Cho, S., Jung, R., Kwon, J.: Spherical transformer. arXiv preprint arXiv:2202.04942 (2022)
7. Choi, C., Kim, S.M., Kim, Y.M.: Balanced spherical grid for egocentric view synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16590–16599 (2023)
8. Coors, B., Condurache, A.P., Geiger, A.: Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In: Proceedings of the European conference on computer vision (ECCV). pp. 518–533 (2018)
9. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to algorithms. MIT press (2022)
10. Deng, Y., Xian, W., Yang, G., Guibas, L., Wetzstein, G., Marschner, S., Debevec, P.: Self-calibrating gaussian splatting for large field-of-view reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 25124–25133 (2025)
11. Gorski, K.M., Hivon, E., Banday, A.J., Wandelt, B.D., Hansen, F.K., Reinecke, M., Bartelmann, M.: Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. The Astrophysical Journal **622**(2), 759–771 (2005)
12. Greene, N.: Environment mapping and other applications of world projections. IEEE computer graphics and Applications **6**(11), 21–29 (1986)
13. Gunes, U., Turkulainen, M., Ren, X., Solin, A., Kannala, J., Rahtu, E.: Fiord: A fisheye indoor-outdoor dataset with lidar ground truth for 3d scene reconstruction and benchmarking. In: Scandinavian Conference on Image Analysis. pp. 3–17. Springer (2025)
14. Guo, X., Xu, Y., Huang, H., Yeung, S.K.: 360dvo: Deep visual odometry for monocular 360-degree camera. IEEE Robotics and Automation Letters **11**(3), 3079–3086 (2026)
15. Huang, H., Chen, Y., Li, L., Cheng, H., Braud, T., Zhao, Y., Yeung, S.K.: Sc-omnigs: Self-calibrating omnidirectional gaussian splatting. In: The Thirteenth International Conference on Learning Representations (2025)

16. Huang, H., Chen, Y., Zhang, T., Yeung, S.K.: 360roam: Real-time indoor roaming using geometry-aware 360° radiance fields. arXiv preprint arXiv:2208.02705 (2022)
17. Huang, H., Liu, C., Zhu, Y., Cheng, H., Braud, T., Yeung, S.K.: 360loc: A dataset and benchmark for omnidirectional visual localization with cross-device queries. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22314–22324 (2024)
18. Huang, H., Xu, Y., Chen, Y., Yeung, S.K.: 360vot: A new benchmark dataset for omnidirectional visual object tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20566–20576 (2023)
19. Huang, L., Bai, J., Guo, J., Li, Y., Guo, Y.: On the error analysis of 3d gaussian splatting and an optimal projection strategy. In: European conference on computer vision. pp. 247–263. Springer (2024)
20. Ito, S., Takama, N., Ito, K., Chen, H.T., Aoki, T.: Erpgs: Equirectangular image rendering enhanced with 3d gaussian regularization. In: 2025 IEEE International Conference on Image Processing (ICIP). pp. 2850–2855. IEEE (2025)
21. Jiang, Y., Yu, C., Xie, T., Li, X., Feng, Y., Wang, H., Li, M., Lau, H., Gao, F., Yang, Y., Jiang, C.: Vr-gs: A physical dynamics-aware interactive gaussian splatting system in virtual reality. arXiv preprint arXiv:2401.16663 (2024)
22. Kageyama, A., Sato, T.: "yin-yang grid": An overset grid in spherical geometry. Geochemistry, Geophysics, Geosystems **5**(9) (2004)
23. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., et al.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
24. Krachmalnicoff, N., Tomasi, M.: Convolutional neural networks on the healpix sphere: a pixel-based algorithm and its application to cmb data analysis. Astronomy & Astrophysics **628**, A129 (2019)
25. Lee, S., Chung, J., Huh, J., Lee, K.M.: Odgs: 3d scene reconstruction from omnidirectional images with 3d gaussian splattings. Advances in Neural Information Processing Systems **37**, 57050–57075 (2024)
26. Lee, S., Chung, J., Kim, K., Huh, J., Lee, G., Lee, M., Lee, K.M.: Omnisplat: Taming feed-forward 3d gaussian splatting for omnidirectional images with editable capabilities. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16356–16365 (2025)
27. Li, J., Hahlbohm, F., Scholz, T., Eisemann, M., Tauscher, J.P., Magnor, M.: Spags: Fast and accurate 3d gaussian splatting for spherical panoramas. In: Computer Graphics Forum. vol. 44, p. e70171. Wiley Online Library (2025)
28. Li, L., Huang, H., Yeung, S.K., Cheng, H.: Omnigs: Fast radiance field reconstruction using omnidirectional gaussian splatting. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 2260–2268. IEEE (2025)
29. Liao, Z., Chen, S., Fu, R., Wang, Y., Su, Z., Luo, H., Ma, L., Xu, L., Dai, B., Li, H., et al.: Fisheye-gs: Lightweight and extensible gaussian splatting module for fisheye cameras. arXiv preprint arXiv:2409.04751 (2024)
30. Ren, S., Wen, T., Fang, Y., Lu, B.: Fastgs: Training 3d gaussian splatting in 100 seconds. arXiv preprint arXiv:2511.04283 (2025)
31. Sadourny, R., Arakawa, A., Mintz, Y.: Integration of the nondivergent barotropic vorticity equation with an icosahedral-hexagonal grid for the sphere. Monthly Weather Review **96**(6), 351–356 (1968)
32. Shin, C., Cho, W.O., Kim, S.J.: Seam360gs: Seamless 360deg gaussian splatting from real-world omnidirectional images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 28970–28979 (2025)

33. Snyder, J.P.: Flattening the earth: two thousand years of map projections. University of Chicago Press (1997)
34. Tu, X., Radl, L., Steiner, M., Steinberger, M., Kerbl, B., De la Torre, F.: Vrsplat: Fast and robust gaussian splatting for virtual reality. Proceedings of the ACM on Computer Graphics and Interactive Techniques **8**(1), 1–22 (2025)
35. Van Brummelen, G.: Heavenly mathematics: The forgotten art of spherical trigonometry. Princeton University Press (2012)
36. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
37. Wu, Q., Esturo, J.M., Mirzaei, A., Moenne-Loccoz, N., Gojcic, Z.: 3dgut: Enabling distorted cameras and secondary rays in gaussian splatting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26036–26046 (2025)
38. Yan, Y., Lin, H., Zhou, C., Wang, W., Sun, H., Zhan, K., Lang, X., Zhou, X., Peng, S.: Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In: ECCV (2024)
39. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023)
40. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: 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 (2018)
41. Zhao, Q., Zhu, C., Dai, F., Ma, Y., Jin, G., Zhang, Y.: Distortion-aware cnns for spherical images. In: IJCAI. pp. 1198–1204 (2018)
42. Zhou, H., Shao, J., Xu, L., Bai, D., Qiu, W., Liu, B., Wang, Y., Geiger, A., Liao, Y.: Hugs: Holistic urban 3d scene understanding via gaussian splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 21336–21345 (June 2024)
43. Zwicker, M., Pfister, H., Van Baar, J., Gross, M.: Ewa volume splatting. In: Proceedings Visualization, 2001. VIS'01. pp. 29–538. IEEE (2001)