MCSplat: Multi-View Photometric and Geometric Consistent Feed-Forward Gaussian Splatting for Driving Scenes

Yifei Wang[1,2]*, Wei Tian[3]✉, Wenhua Bao[2], Jia Zhou[3], Hanshi Wang[4], Fudong Ge[4], Jianhua Wu[3], Shuai Yuan[5], Zhipeng Zhang[5], Xingliang Liu[2,†], Xianming Zeng[2,†], Jianyun Xu[2,†], Bingzhao Gao[1,3]✉

[1]Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University
[2]Zaofu Intelligent Technology Co., Ltd.
[3]College of Automotive and Energy Engineering, Tongji University
[4]Institute of Automation, Chinese Academy of Sciences, [5]SAI, Shanghai Jiao Tong University
https://yfwang0824.github.io/mcsplat-project/

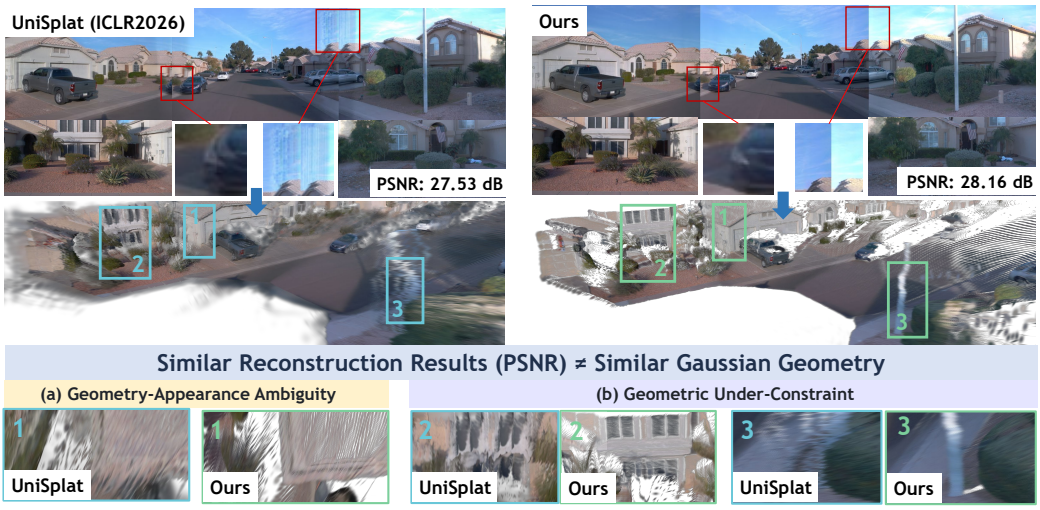


Figure 1: As shown by the comparison with UniSplat [1], comparable PSNR can still correspond to fundamentally different Gaussian representations. MCSplat addresses **(a) cross-view geometry-appearance ambiguity** and **(b) local geometric supervision unreliability** by combining multi-view photometric canonicalization with uncertainty-aware geometric self-supervision, producing cleaner Gaussian geometry and more consistent structures in cross-camera overlap regions.

Abstract

Driving-scene reconstruction via feed-forward Gaussian splatting from sparse multi-view images remains challenging due to two factors. First, cross-camera photometric inconsistency, caused by variations in exposure, ISP, and illumination, can be incorrectly absorbed into Gaussian geometry. Second, unreliable local geometric supervision arises from normal-based constraints near depth discontinuities. To address these issues, we present **MCSplat**, a multi-view consistent feed-forward Gaussian splatting framework for driving scenes. MCSplat constructs a canonical appearance space from depth-guided multi-camera correspondences to

*Work done during an internship at Zaofu Intelligent Technology Co., Ltd.
†Project leaders of Zaofu Intelligent Technology Co., Ltd.
✉Corresponding authors: Wei Tian (tian_wei@tongji.edu.cn) and Bingzhao Gao (gaobz@tongji.edu.cn).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

decouple geometry learning from camera-specific appearance. A multi-scale Bilateral Grid Transformer restores raw camera appearance from canonical renderings, enabling photometric supervision without contaminating Gaussian geometry. To improve local geometric reliability, we estimate Fisher-information-based uncertainty from photometric sensitivity and use it to adaptively weight surface-oriented self-supervision. Extensive experiments demonstrate improved reconstruction quality and generalization across driving datasets. Geometry-oriented evaluations further reveal enhanced cross-view depth consistency and local surface-normal agreement, while qualitative results show fewer geometric artifacts and more coherent structures across overlapping views.

1 Introduction

Neural rendering methods such as 3D Gaussian Splatting (3DGS) [2] have enabled high-quality 3D reconstruction and novel view synthesis, making urban driving scene reconstruction as a cornerstone of closed-loop autonomous driving simulation [3] and long-horizon planning [4]. Since per-scene optimization remains computationally expensive, recent generalizable splatting frameworks [5, 1, 6] directly predict Gaussian primitives from multi-frame images. Leveraging Geometry Foundation Models (GFMs) [7, 8, 9, 10], they use dense depth predictions as priors for Gaussian means, along with a Gaussian head to predict rotation, scale, opacity, color, and motion. Despite improved efficiency and generalization, they still inherit two key challenges from 3DGS optimization:

Cross-view geometry-appearance ambiguity: Ideally, multi-view supervision should attribute cross-view image differences to the underlying 3D structure rather than camera-dependent photometric effects. In real-world driving scenarios, however, photometric consistency is difficult to guarantee across views and time due to camera-dependent exposure, color response, automatic image signal processing (ISP), and changing illumination. As illustrated in Fig. 1 (a), inconsistent appearance in cross-camera overlap regions can cause the same structure, such as the roof, to be explained as separate surfaces, producing floating Gaussians and degraded geometry [11].

Local geometric supervision unreliability: Driving scenes provide dense observations along the ego trajectory but sparse off-trajectory views. Even with GFM-predicted depths as Gaussian means, rotation, scale, opacity, and color remain weakly constrained by RGB and depth losses, allowing multiple Gaussian configurations to fit training views while differing in off-trajectory renderings. As illustrated in Fig. 1 (b), this can lead to geometrically incorrect yet view-consistent Gaussian representations, making cross-lane novel view synthesis challenging. Geometric self-supervision can alleviate this ambiguity by aligning Gaussian orientations with local surface normals. However, surface normals are typically derived from local finite differences of depth or Gaussian centers, which become unreliable near depth discontinuities and may introduce inaccurate supervision [12].

Existing methods attempt to mitigate the geometry-appearance ambiguity using appearance embeddings, global color transforms, explicit ISP parameters, or bilateral-grid-based local color transforms [11, 13, 14, 15, 16, 17]. However, most of these methods rely on per-scene optimization, which incurs high computational cost and limits cross-scene generalization. Recent feed-forward bilateral-grid methods reduce the need for scene-specific optimization by assuming sufficient overlap between reference and target views [18], making them less suitable for surround-view driving systems where inter-camera overlap is inherently limited.

Beyond photometric variation, the under-constrained nature of geometric reconstruction is typically addressed by introducing additional supervisory signals. Existing methods employ semantic, flow, or normal priors to constrain geometry [19, 20, 21]. However, these priors rely on strong teacher models and are costly to scale to feed-forward reconstruction, as they require additional data preparation, storage, and training overhead. Self-supervised geometric constraints reduce the reliance on extra priors but remain unreliable near depth discontinuities or when depth estimates from foundation models are inaccurate [12].

In this work, we propose to enhance the GFM-based feed-forward Gaussian reconstruction for driving scenes by combining geometry–appearance decoupling with uncertainty-aware geometric self-supervision. To mitigate the geometry–appearance ambiguity, we provide appearance-consistent inputs to the feed-forward Gaussian model, thereby reducing the impact of cross-view and cross-time photometric discrepancies on Gaussian prediction. A lightweight transformer predicts multi-scale 3D

bilateral grids to map rendered images back to the original GT appearance, enabling photometric supervision while preserving view-specific imaging effects. To handle local geometric supervision unreliability, we use Fisher information with respect to the photometric loss to quantify geometric uncertainty and construct an uncertainty-aware self-supervised geometric loss.

Our contributions are summarized as follows. **First,** we propose **MCSplat**, a GFM-compatible geometry–appearance decoupling framework that builds a canonical appearance space and recovers camera-specific appearance with a multi-scale Bilateral Grid Transformer, reducing cross-view geometry–appearance ambiguity. **Second,** we introduce Fisher information-guided geometric self-supervision that estimates geometric uncertainty from photometric sensitivity and adaptively weights surface regularization, alleviating unreliable local supervision. **Third,** experiments on Waymo and nuScenes demonstrate improved reconstruction quality and geometric consistency, including under challenging cross-camera appearance variations.

2 Related Works

Neural Scene Reconstruction for Driving Scenes. NeRF [22] enables high-quality novel view synthesis but is inefficient for rendering, while 3DGS [2] uses explicit learnable Gaussians for real-time rendering, making it suitable for dynamic driving scenes. Many driving-scene methods adopt per-scene optimization with priors, including LiDAR [23], semantics [24], optical flow [20], human models [25], scene flow [26], and multi-scale voxels [21]. However, scene-specific retraining limits scalability and cross-scene generalization. Feed-forward methods instead directly regress Gaussian representations from images. STORM [27] predict pixel-aligned Gaussian motion. DrivingForward [28] and DrivingScene [29] infer Gaussian primitives from surround-view inputs. Recent GFM-based methods, including DGGT [5], StreetForward [6], UniSplat [1], and ReconDrive [30], leverage GFMs [7, 8, 9, 10] to constrain Gaussian positions and decode Gaussian attributes. Despite improved efficiency and generalization, these methods remain vulnerable to geometry–appearance ambiguity under cross-camera photometric inconsistency. Additionally, they are prone to geometrically under-constrained reconstruction due to unreliable geometric supervision.

Appearance Harmonization in Neural Scene Representations. Cross-camera photometric inconsistency is inevitable in real world and induces geometry–appearance ambiguity. NeRF in the Wild [11] and WildGaussians [13] model imaging variation with appearance branches or observation-dependent embeddings. STORM [31] learns coarse global affine transforms for feed-forward reconstruction. PPISP [15] explicitly parameterizes ISP factors such as exposure, vignetting, and camera response functions. However, these methods mainly rely on global or per-scene optimization, making them less suitable for handling local and cross-camera discrepancies. Thus, refined photometric operation has been employed in recent methods. Bilateral-grid [32] provides spatially varying photometric transforms. BilaRF [16] uses bilateral-space affine transforms for pixel-wise color adjustment. CHROMA [18] explores feed-forward bilateral-grid prediction for multi-view harmonization. However, these methods require substantial overlap between reference and target views, limiting their applicability to surround-view driving scenes with small inter-camera overlap.

Uncertainty Quantification in Neural Scene Reconstruction. Uncertainty estimation measures reconstruction reliability in challenging regions such as occlusion boundaries, weak textures, and dynamic objects [33, 34, 35]. Fisher information [36] provides a parameter-sensitivity-based uncertainty measure and has been used for active view selection, pruning, and robust reconstruction. FisherRF [33] uses expected information gain for active view selection. PUP-3DGS [37] adopts sensitivity scores for Gaussian pruning. OUGS [38] propagates Gaussian-parameter uncertainty to pixel space for object-aware view selection. However, uncertainty has primarily been explored in the context of per-scene optimization, with its role in feed-forward reconstruction remaining limited. In contrast, we compute geometry uncertainty from the photometric loss within a feed-forward framework, and leverage it for adaptive geometric self-supervision.

3 Methodology

We propose **MCSplat**, a framework for feed-forward reconstruction of multi-view driving scenes by integrating geometry–appearance decoupling with uncertainty-aware geometric self-supervision,

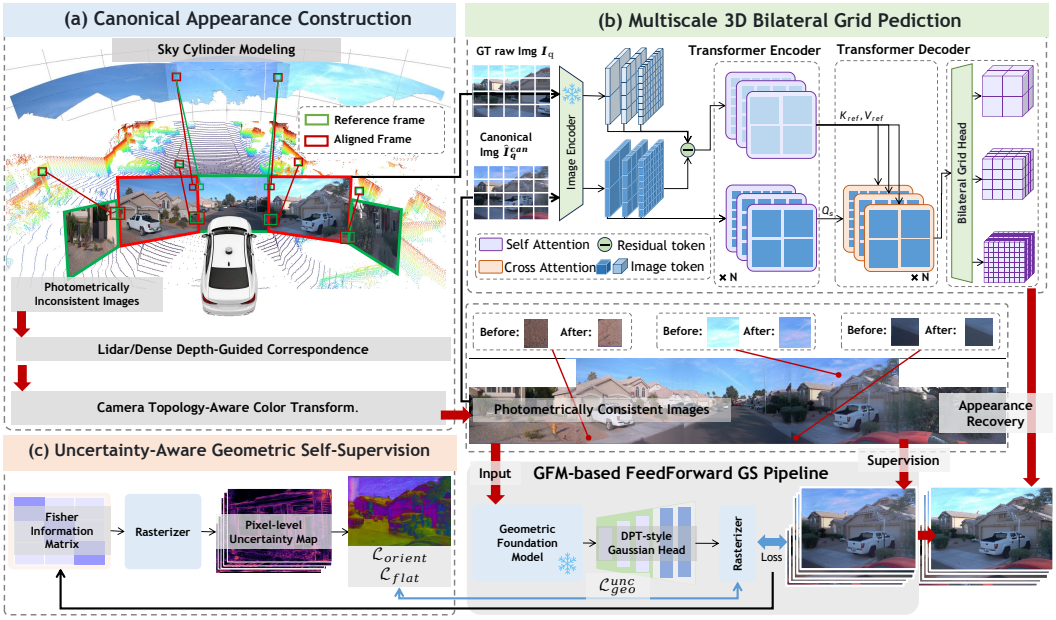


Figure 2: Overview of MCSplat. (a) Depth-guided canonical appearance construction aligns inconsistent multi-view images. (b) A multi-scale bilateral-grid transformer restores camera-specific appearance from canonical renderings. (c) Fisher information-based uncertainty guides geometric self-supervision in under-constrained regions.

shown in Fig. 2. Its goal is to predict reliable 3D Gaussian primitives from multi-view images, enabling novel-view synthesis for closed-loop autonomous driving simulation.

In our framework, given photometrically inconsistent multi-view images, we first construct a *canonical appearance space* using depth-guided correspondences, and learn Gaussian primitives within this normalized domain. To recover the original, camera-specific appearance, a lightweight transformer predicts multi-scale 3D bilateral grids and applies them to rendered images in a coarse-to-fine manner. We further estimate pixel-wise geometric uncertainty based on photometric-loss sensitivity, and use it to adaptively weight geometric self-supervision in weakly constrained regions.

3.1 Robust Canonical Appearance Construction

To reduce geometry–appearance ambiguity at the input stage, we construct a canonical appearance space using depth-guided correspondences. Given a multi-view camera set $\mathcal{V}$, reference cameras $\mathcal{V}_{\text{ref}} \subset \mathcal{V}$ are selected according to rig topology and inter-camera overlap. Reference views define the canonical appearance space with identity transforms, while non-reference views are aligned to adjacent reference anchors. For example, in the 5-camera Waymo layout shown in Fig. 2 (a), we use $\mathcal{V}_{\text{ref}} = \{\text{Front}, \text{SideLeft}, \text{SideRight}\}$. Consequently, the two remaining cameras, FrontLeft and FrontRight are each anchored by two neighboring references. For the nuScenes layout, we select $\mathcal{V}_{\text{ref}} = \{\text{Front}, \text{BackLeft}, \text{BackRight}\}$. The remaining cameras, FrontLeft, FrontRight, and Back, are aligned to their neighboring reference anchors.

For each pixel p_i in a reference camera view $r \in \mathcal{V}_{\text{ref}}$ with a valid depth $D_r(p_i)$, we back-project it to 3D coordinates and then reproject it into a new pixel $\tilde{p}_i$ in a target camera view t, where t is selected from the non-reference camera set $\mathcal{V} \setminus \mathcal{V}_{\text{ref}}$. For sky regions without reliable depth, we establish correspondences by intersecting each reference-camera ray with a distant ego-centered sky cylinder. The resulting 3D intersection point is then projected into the target camera, yielding a sky-region matched pixel.

Given correspondences between r and t, we estimate a per-channel diagonal affine transform $\mathbf{A}_{t\to r}$ (with parameters a_c, b_c for image channel I^c) from target to reference appearance:

$$I_r^c(p_i) \approx a_c I_t^c(\tilde{p}_i) + b_c, \qquad c \in \{R, G, B\}, \tag{1}$$

by solving the optimization problem with hyperparameters λ_a, λ_b:

$$\min_{\{a_c,b_c\}} \sum_i \rho(a_c I_t^c(\tilde{p}_i) + b_c - I_r^c(p_i)) + \lambda_a \sum_c (a_c - 1)^2 + \lambda_b \sum_c b_c^2. \tag{2}$$

Here, $\rho(\cdot)$ is a robust penalty, and the identity prior makes the transform fall back to no correction when correspondences are sparse or unreliable. We additionally use a shared-gain fallback for chromatically degenerate samples, a few IRLS reweighting steps, and residual-based inlier refitting when enough inliers are available.

For target cameras with two adjacent references, to avoid biased single-reference alignment and discontinuous hard switching, we interpolate target-to-reference transforms along the horizontal image axis. Given a horizontal image coordinate $u \in [0, W-1]$ and the ratio $\alpha(u) = u/(W-1)$, for a target camera t with adjacent references r_1 and r_2, the spatially varying affine field is

$$\mathbf{A}_t(u) = (1 - \alpha(u))\, \mathbf{A}_{t \to r_1} + \alpha(u)\, \mathbf{A}_{t \to r_2}. \tag{3}$$

Sky and non-sky regions share the same interpolation form but use separately estimated affine transforms, accounting for their distinct color statistics.

3.2 Multi-Scale Bilateral Grid Transformer for Appearance Recovery

Given the canonical render $\hat{I}_q^{\text{can}}$, we recover the original camera-specific appearance I_q without re-entangling photometric variation with geometry. For each query view, we predict multi-scale bilateral transforms and apply them to $\hat{I}_q^{\text{can}}$ in a coarse-to-fine manner, enabling compact modeling of global exposure shifts, color biases, and spatially varying local corrections.

3D bilateral-grid parameterization. At a given scale m, the appearance transform is represented by a 3D bilateral grid

$$\mathbf{B}_q^{(m)} \in \mathbb{R}^{12 \times L_m \times H_m \times W_m}, \tag{4}$$

where H_m, W_m are spatial resolutions and L_m is the guidance resolution. Each grid cell stores a flattened 3×4 affine color transform. For a pixel p, we slice the grid using its normalized coordinates $(x_p^{(m)}, y_p^{(m)})$ and grayscale guidance $\gamma_p = \text{gray}(\hat{I}_q^{\text{can}}(p))$:

$$\boldsymbol{\theta}_{q,p}^{(m)} = \mathcal{S}\Big(\mathbf{B}_q^{(m)}, x_p^{(m)}, y_p^{(m)}, \gamma_p\Big), \tag{5}$$

where $\boldsymbol{\theta}_{q,p}^{(m)}$ is a 3×4 affine color transform and reshaped into an affine pair $(\mathbf{A}_{q,p}^{(m)}, \mathbf{b}_{q,p}^{(m)})$. The multi-scale transforms are composed in homogeneous coordinates as

$$\mathbf{T}_{q,p} = \mathbf{T}_{q,p}^{(M)} \cdots \mathbf{T}_{q,p}^{(1)}, \qquad \mathbf{T}_{q,p}^{(m)} = \begin{bmatrix} \mathbf{A}_{q,p}^{(m)} & \mathbf{b}_{q,p}^{(m)} \\ \mathbf{0}^\top & 1 \end{bmatrix}, \qquad m \in \{1, \cdots, M\}, \tag{6}$$

and the recovered image is obtained by

$$[\hat{I}_q(p), 1]^\top = \mathbf{T}_{q,p} [\hat{I}_q^{\text{can}}(p), 1]^\top. \tag{7}$$

Here, grids from coarse scales capture global shifts while fine grids refine local photometric variations.

Residual-token bilateral grid transformer decoder. As illustrated in Fig. 2 (b), we predict bilateral grids using a lightweight transformer decoder built on multi-scale token interactions. For a view q, a frozen image encoder extracts patch tokens $P_q^{\text{can}}, P_q^{\text{raw}} \in \mathbb{R}^{P \times D}$ from the canonicalized and raw images, respectively. At each scale m, we pool them to the target grid resolution:

$$Q_q^{(m)} = \text{Pool}\big(P_q^{\text{can}}; H_m, W_m\big), \qquad R_q^{(m)} = \text{Pool}\big(P_q^{\text{raw}} - P_q^{\text{can}}; H_m, W_m\big). \tag{8}$$

Here, $Q_q^{(m)}$ serves as canonical-domain query tokens, while $R_q^{(m)}$ provides residual key–value tokens. Compared with raw tokens, residual tokens better isolate the photometric transform to be recovered and reduce content redundancy in cross-attention.

For each scale, with the learned 2D positional embedding $\mathbf{E}_m$, self-attention SA_m captures spatial dependencies in the canonical token grid, followed by cross-attention CA_m that injects residual appearance cues:

$$Q_q^{(m,1)} = \text{SA}_m\Big(Q_q^{(m)} + \mathbf{E}_m\Big), \qquad Q_q^{(m,2)} = \text{CA}_m\Big(Q_q^{(m,1)}, R_q^{(m)} + \mathbf{E}_m\Big). \tag{9}$$

The refined tokens are passed into a feedforward network and mapped to bilateral-grid coefficients through a scale-specific head:

$$\mathbf{B}_q^{(m)} = \mathrm{Head}_m\Big(\mathrm{FFN}_m\Big(Q_q^{(m,2)}\Big)\Big) \in \mathbb{R}^{12\times L_m\times H_m\times W_m}. \tag{10}$$

This design captures transforms from global shifts to fine local corrections, while residual cross-attention focuses the prediction on raw-to-canonical appearance differences.

3.3 Uncertainty-Aware Geometric Self-Supervision

Fisher information-based geometry uncertainty. For each Gaussian g_i, we estimate its geometric uncertainty by evaluating the sensitivity of the photometric loss to its center $\boldsymbol{\mu}_i \in \mathbb{R}^3$. Over the current training views $\mathcal{V}_{\mathrm{cur}}$, we adopt a smooth Charbonnier photometric loss:

$$\mathcal{L}_{\mathrm{photo}} = \sum_{q\in\mathcal{V}_{\mathrm{cur}}}\sum_{p}\sqrt{\Big(\hat{I}_q^{\mathrm{can}}(p) - I_q^{\mathrm{can}}(p)\Big)^2 + \epsilon^2}. \tag{11}$$

Here, p indexes valid pixels in view q. $\epsilon > 0$ is a small constant that ensures differentiability and numerical stability.

We estimate a scalar translational Fisher-information score for each Gaussian using the squared photometric-loss gradient norm:

$$F_{i,q}^{xyz} = \left\| \frac{\partial \mathcal{L}_{\mathrm{photo},q}}{\partial \boldsymbol{\mu}_i} \right\|_2^2, \qquad \bar{F}_i^{xyz} = \frac{1}{|\mathcal{V}_{\mathrm{cur}}|}\sum_{q\in\mathcal{V}_{\mathrm{cur}}} F_{i,q}^{xyz}. \tag{12}$$

Intuitively, a large score indicates strong photometric-loss sensitivity to perturbations of the Gaussian center, which may arise in geometrically unstable regions.

Why Gaussian-center Fisher information? We use photometric-loss sensitivity to Gaussian centers (x, y, z) as a geometric uncertainty cue, as it more directly reflects positional constraints than color or scale sensitivity. As shown in Fig. 3, the center-based uncertainty map better captures object boundaries and fine structures (e.g., the white truck boundary and jeep grille), whereas color- and scale-based maps are less discriminative. This motivates its use for adaptive geometric supervision.

We normalize the center-based Fisher scores within each mini-batch and convert them into uncertainty weights:

$$\tilde{F}_i^{xyz} = \frac{\bar{F}_i^{xyz}}{\max_j \bar{F}_j^{xyz} + \varepsilon}, \qquad w_i = \exp\Big(-\gamma \tilde{F}_i^{xyz}\Big), \tag{13}$$

where $\gamma > 0$ controls the sharpness of the weighting. Since w_i decreases monotonically with Fisher information, Gaussians that are weakly constrained by photometric supervision receive larger geometric regularization weights, while confident Gaussians are regularized less aggressively. In practice, this score is computed online within each training step and detached before weighting the geometric regularizers; implementation details are provided in Sec. A.2.

Notably, this uncertainty measure is derived entirely from the model's own photometric sensitivity, without relying on external depth confidence or heuristic visibility priors, making it naturally compatible with feed-forward reconstruction.

Uncertainty-guided surface regularization. We apply the uncertainty weights to surface-oriented geometric self-supervision on non-sky Gaussians. Given the predicted Gaussian set, we use the pixel-aligned Gaussian centers $\{\boldsymbol{\mu}_i\}$ as a 3D point field and estimate local surface normals $\mathbf{n}_i^{\mathrm{surf}}$ by finite differences over neighboring centers.

For a Gaussian g_i, we define its normal $\mathbf{n}_i^{\mathrm{gs}}$ as the principal axis with the smallest scale $s_{i,k}$:

$$k_i^\star = \arg\min_{k\in\{1,2,3\}} s_{i,k}, \qquad \mathbf{n}_i^{\mathrm{gs}} = \mathbf{R}_i \mathbf{e}_{k_i^\star}, \tag{14}$$

where $\mathbf{R}_i$ is the rotation matrix and $\mathbf{e}_{k_i^\star}$ is the canonical basis vector of the thinnest axis. We then define the uncertainty-weighted orientation loss:

$$\ell_i^{\mathrm{orient}} = 1 - \left|\left\langle \mathbf{n}_i^{\mathrm{surf}}, \mathbf{n}_i^{\mathrm{gs}} \right\rangle\right|, \qquad \mathcal{L}_{\mathrm{orient}} = \frac{\sum_i w_i\, \ell_i^{\mathrm{orient}}}{\sum_i w_i}. \tag{15}$$

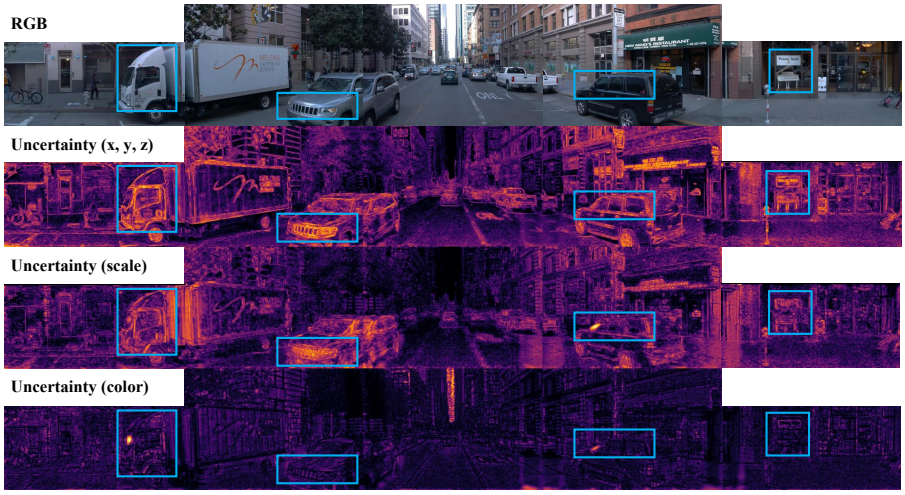


Figure 3: Visualization of photometric-loss uncertainty maps for Gaussian color, scale, and center parameters. The center-based map captures local structure around the white truck boundary and the jeep grille more distinctly than the scale map and the color map.

The above absolute inner product removes normal sign ambiguity and encourages the thinnest Gaussian axis to align with the local surface normal $\mathbf{n}_i^{\text{surf}}$.

To further encourage surface-like Gaussians, we penalize the thinnest scale $s_{i,k_i^\star}$ in loss term:

$$\ell_i^{\text{flat}} = s_{i,k_i^\star}, \qquad \mathcal{L}_{\text{flat}} = \frac{\sum_i w_i \, \ell_i^{\text{flat}}}{\sum_i w_i}. \tag{16}$$

The final uncertainty-aware geometric self-supervision loss is defined as

$$\mathcal{L}_{\text{geo}}^{\text{unc}} = \lambda_{\text{orient}} \mathcal{L}_{\text{orient}} + \lambda_{\text{flat}} \mathcal{L}_{\text{flat}}. \tag{17}$$

with hyperparameter weights $\lambda_{\text{orient}}, \lambda_{\text{flat}}$.

3.4 Training Objective

We optimize the model in two stages. **In Stage 1**, the GFM backbone is frozen, while the sparse 3D convolution module and Gaussian head are trained in the canonical appearance space. The objective is compactly written as

$$\mathcal{L}_{\text{stage1}} = \mathcal{L}_{\text{can}} + \mathcal{L}_{\text{aux}}, \tag{18}$$

where $\mathcal{L}_{\text{can}}$ supervises canonical rendering with MSE and LPIPS losses, and $\mathcal{L}_{\text{aux}}$ includes dynamic-region and scale-alignment losses. **In Stage 2**, most of the Gaussian head is frozen except for the Gaussian MLPs, while the bilateral-grid transformer is optimized for appearance recovery. The predicted bilateral grids map the canonical render to the GT image domain as in Eq. (7). The objective becomes

$$\mathcal{L}_{\text{stage2}} = \mathcal{L}_{\text{stage1}} + \mathcal{L}_{\text{img}} + \lambda_7 \mathcal{L}_{\text{tv}} + \mathcal{L}_{\text{geo}}^{\text{unc}}, \tag{19}$$

where $\mathcal{L}_{\text{img}}$ supervises appearance recovery, $\mathcal{L}_{\text{tv}}$ regularizes the predicted bilateral grids. The full objective is provided in Sec. A.3.

4 Experiments

4.1 Experimental Settings

Datasets and Metrics. We primarily evaluate on the Waymo Open Dataset [39], which contains 798 training and 202 validation scenes, each lasting about 20 seconds at 10 Hz with 5 vehicle-mounted cameras. Following UniSplat [1], we evaluate input-view reconstruction using images at timestamp t as targets and novel-view synthesis using images at $t + 1$. We report PSNR, SSIM [40],

Table 1: Quantitative comparison of image quality on the Waymo and nuScenes datasets. The best, second-best, and third-best results are highlighted. *: Reported by UniSplat. [3]: Evaluated on the front 3 views only.

Method	Views	Waymo						nuScenes		
		Reconstruction			Novel View Synthesis			Recon&NVS		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
EvolSplat* [44]	Front	23.35	0.70	0.29	–	–	–	–	–	–
UniSplat* [1]	Front	28.93	0.86	0.18	27.34	0.80	0.20	–	–	–
Ours	Front	**31.05**	**0.89**	**0.12**	**27.64**	**0.84**	**0.16**	–	–	–
PixelSplat* [45]	Multi	–	–	–	–	–	–	21.51	0.616	0.372
DriveRecon*[3] [46]	Multi	23.86	0.72	0.33	17.32	0.58	0.53	–	–	–
MVSplat* [47]	Multi	24.94	0.80	0.23	22.04	0.68	0.34	21.61	0.658	0.295
DepthSplat* [48]	Multi	25.38	0.76	0.26	23.86	0.70	0.31	–	–	–
Omni-Scene* [49]	Multi	–	–	–	–	–	–	24.27	0.736	**0.237**
UniSplat* [1]	Multi	29.58	0.86	0.17	25.98	0.76	0.24	25.37	0.765	0.246
Ours	Multi	**30.53**	**0.88**	**0.13**	**26.35**	**0.76**	**0.22**	**26.19**	**0.783**	0.248

and LPIPS [41]. **We also construct an appearance-challenging subset of 57 scenes** selected by input-only inter-camera appearance discrepancy scores from adjacent camera pairs in matched or overlapping regions. More details and results about the subset are provided in Sec. A.5. We also train and evaluate MCSplat independently on nuScenes, following the protocol described in Sec. 4.2.

Baseline. UniSplat [1] is a GFM-based feed-forward Gaussian reconstruction method for dynamic driving scenes. It constructs a 3D latent scaffold from multi-camera videos, performs spatio-temporal fusion in the scaffold space, and decodes dynamic-aware Gaussians using both point and voxel-based branches. We choose UniSplat as our baseline because, to the best of our knowledge, UniSplat is the state-of-the-art feed-forward method trained and evaluated under the full 5-camera setting.

Implementation Details. We follow the baseline architecture, including the GFM backbone, semantic feature extraction, and Gaussian head design. For Waymo, both input and rendering resolutions are 350×518, while nuScenes uses a rendering resolution of 400×224. Training is performed in two stages as described in Sec. 3.4. The Bilateral Grid Transformer uses a DINOv2 [42] ViT-Small encoder for appearance token extraction and predicts multi-scale bilateral grids for appearance recovery. We train the model using AdamW [43] on 16 NVIDIA H20 GPUs with a total batch size of 32. Additional details are provided in Sec. A.4.

4.2 Experimental Results

Image quality on Waymo. We compare MCSplat with MVSplat [47], DepthSplat [48], EvolSplat [44], DriveRecon [46], and UniSplat [1], using the Waymo results reported by UniSplat for comparison. As shown in Tab. 1, **under the front-camera setting**, MCSplat improves all reconstruction metrics over UniSplat (PSNR $+2.12$ dB and LPIPS -33.3%) as well as all NVS metrics. **Under the 5-camera setting**, MCSplat improves all reconstruction metrics. Qualitative comparisons in Figs. 4 and 5 illustrate fewer texture-induced geometric artifacts and improved cross-camera consistency; additional visualizations, including cross-lane NVS, are provided in Secs. A.6 to A.8.

Image quality on nuScenes. We additionally evaluate on nuScenes following the protocol used by UniSplat [1]: the middle frame of each temporal interval is used as input, and the first, middle, and last frames are evaluation targets. As shown in Tab. 1, MCSplat achieves 26.19 dB PSNR(+0.82 dB) and 0.783 SSIM(+0.018). This supports generalization beyond the Waymo 5-camera configuration.

Cross-View Geometry and Normal Consistency on Waymo. Image-quality metrics alone cannot directly assess Gaussian geometry. We therefore evaluate cross-view depth reprojection errors and dense pseudo-normal consistency in Tab. 2. Since LiDAR observations are too sparse for reliable dense normal estimation, we use MoGe-2 [50] predictions as pseudo-normal references. Depth reprojection errors are evaluated in valid overlapping regions of adjacent cameras. Compared with UniSplat, MCSplat reduces the mean depth reprojection error by 61.73% (2.378 to 0.910 m) and

Table 2: Geometry-oriented evaluation on Waymo. Normal consistency is measured against MoGe-2[50] pseudo-normals.

Method	Depth Reproj. (m)		Pseudo-Normal Consistency			
	Mean↓	RMSE↓	Mean (°)↓	$< 5°$ (%)↑	$< 11.25°$ (%)↑	$< 30°$ (%)↑
UniSplat	2.378	6.427	115.05	0.02	0.10	1.18
MCSplat	**0.910**	**3.920**	**92.82**	**4.24**	**10.08**	**17.38**

Table 3: Ablation on the Stage-1 training objective.

Stage-1 Obj.	PSNR↑	SSIM↑	LPIPS↓
Raw GT	29.53	0.85	0.19
Canonical	**30.53**	**0.88**	**0.13**

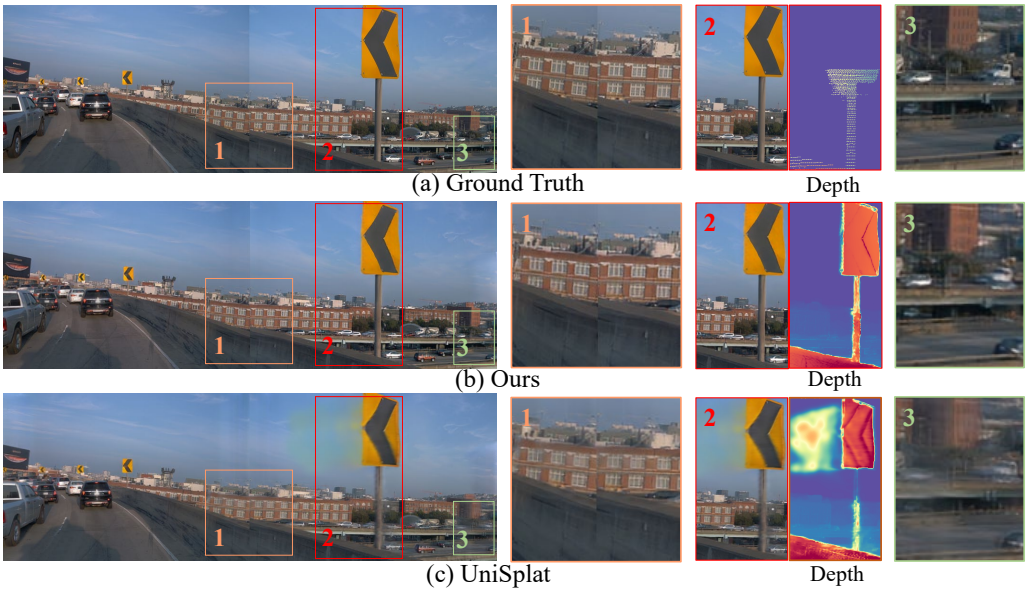


Figure 4: Qualitative comparisons on the Waymo dataset. Compared with existing methods, MCSplat produces finer and cross-camera-consistent results, avoiding texture-induced depth ambiguity within the yellow road sign.

RMSE by 39.01% (6.427 to 3.920 m). Meanwhile, the mean pseudo-normal angular error decreases from 115.05° to 92.82°, while the fraction of pixels with angular error below 30° increases from 1.18% to 17.38%. These results indicate improved cross-view geometric consistency and local surface orientation.

Waymo Multi-View Appearance-Inconsistent Subset. To assess robustness under strong cross-camera photometric variations, we additionally evaluate MCSplat on a challenging subset of 57 Waymo validation scenes, selected using model-independent cross-camera appearance discrepancy scores (see Sec. A.5). The detailed reconstruction results are reported in the appendix (Tab. 6).

4.3 Ablation Study

We evaluate the roles of Canonical Appearance Construction (CAC), uncertainty-aware geometric self-supervision (Geo), and the Bilateral Grid Transformer (BGT) on Waymo using input-view reconstruction and geometric consistency.

Canonical-Space Training in Stage 1. Tab. 3 compares raw-image versus canonical-image supervision for Stage 1 while keeping the remainder of the training protocol unchanged. Canonical supervision improves PSNR from 29.53 to 30.53 dB, SSIM from 0.85 to 0.88, and LPIPS from 0.19 to 0.13. Training Stage 1 directly on raw, camera-dependent images can introduce conflicting photometric signals in overlap regions and an inconsistent training target for subsequent canonical-space appearance recovery.

Component-Wise Ablation. Tab. 4 evaluates the contributions of CAC, Geo, and BGT to cross-view depth consistency, pseudo-normal consistency, and input-view reconstruction. CAC reduces

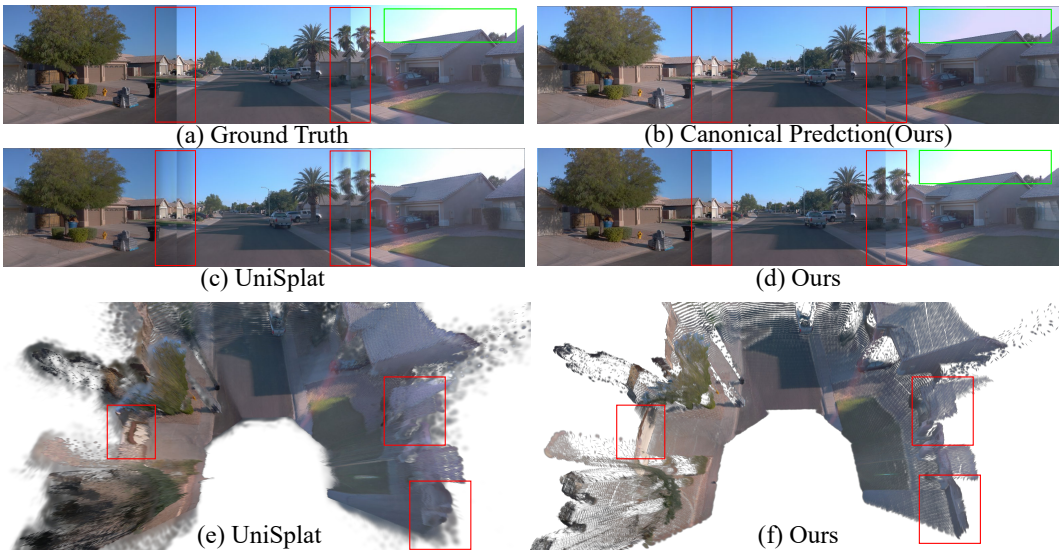


Figure 5: Qualitative results under cross-camera appearance inconsistency. **Zoom in for best view.** Compared with UniSplat (c), our canonical rendering removes camera-boundary artifacts (b), while achieving effective appearance recovery (d).

Table 4: Component ablation on the Waymo dataset. "Can." and "Raw" denote canonicalized and raw GT images used for evaluation, respectively. Depth reprojection error is evaluated in adjacent-camera overlap regions, while normal consistency is measured against MoGe-2 [50] pseudo-normals.

Method	Eval.	Depth Reproj. (m)		Reconstruction			Pseudo-Normal Consistency			
		Mean↓	RMSE↓	PSNR↑	SSIM↑	LPIPS↓	Mean (°)↓	$< 5°$ (%)↑	$< 11.25°$ (%)↑	$< 30°$ (%)↑
Baseline	Raw	2.378	6.427	29.58	0.86	0.17	115.05	0.02	0.10	1.18
Baseline + CAC	Can.	**0.967**	**4.051**	**31.06**	**0.88**	**0.13**	**97.25**	**0.17**	**0.74**	**4.08**
Baseline + CAC + BGT	Raw	0.967	4.051	30.76	**0.88**	**0.13**	97.25	0.17	0.74	4.08
Baseline + CAC + Geo	Can.	0.910	3.920	31.09	0.88	0.14	92.82	4.24	10.08	17.38
Baseline + CAC + Geo	Raw	0.910	3.920	29.35	0.86	0.14	92.82	4.24	10.08	17.38
MCSplat (CAC + Geo + BGT)	Raw	**0.910**	**3.920**	**30.53**	**0.88**	**0.13**	**92.82**	**4.24**	**10.08**	**17.38**

the mean depth reprojection error by 59.34% (2.378 to 0.967 m) and RMSE from 6.427 to 4.051 m, indicating improved shared geometry through consistent appearance supervision. It also reduces the mean pseudo-normal angular error from $115.05°$ to $97.25°$. Adding Geo further lowers the depth errors to 0.910 m(mean) and 3.920 m (RMSE), while increasing the fraction of pixels with normal error below $30°$ from 4.08% to 17.38%. BGT primarily restores camera-dependent appearance: with CAC and Geo fixed, it improves raw-target PSNR from 29.35 to 30.53 dB, SSIM from 0.86 to 0.88, and LPIPS from 0.14 to 0.13, without changing depth or normal consistency. Without Geo, CAC+BGT achieves a slightly higher raw-target PSNR (30.76 dB), suggesting a geometry–photometry trade-off. These results support the complementary roles of CAC (cross-view geometry), Geo (local surface geometry), and BGT (appearance recovery).

5 Conclusion

We presented **MCSplat**, a multi-view consistent feed-forward Gaussian splatting framework for driving-scene reconstruction. MCSplat decouples geometry learning from camera-specific appearance by constructing a canonical appearance space, recovering raw camera appearance with a multi-scale Bilateral Grid Transformer, and applying Fisher-information guided geometric self-supervision to uncertain Gaussians. Experiments on Waymo and nuScenes demonstrate improved reconstruction quality and cross-dataset generalization, while geometry-oriented evaluations show enhanced cross-view depth consistency and local surface-normal agreement. Future work will explore stronger temporal appearance modeling and explicit modeling of ISP parameters.

Acknowledgments and Disclosure of Funding

This work was supported by the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (Grant No. JYB2025XDXM123), the Science and Technology Commission of Shanghai Municipality (Grant No. 25ZR1401356), the Shanghai Municipal Science and Shanghai Automotive Industry Science and Technology Development Foundation (Grant No. 2601), the National Key R&D Program of China (Grant No. 2024YFE0115400), the National Natural Science Foundation of China (Grant No. 62273256), and the Jiangxi Provincial Major Science and Technology Special Project (Grant No. 20262ABE010004). We are particularly grateful to Zaofu Intelligent Technology Co., Ltd. for its valuable support.

References

[1] Chen Shi, Shaoshuai Shi, Xiaoyang Lyu, Chunyang Liu, Kehua Sheng, Bo Zhang, and Li Jiang. Unisplat: Unified spatio-temporal fusion via 3d latent scaffolds for dynamic driving scene reconstruction. In *The Fourteenth International Conference on Learning Representations*, 2026.

[2] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139–1, 2023.

[3] Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and Raquel Urtasun. Unisim: A neural closed-loop sensor simulator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1389–1399, 2023.

[4] Hao Gao, Shaoyu Chen, Bo Jiang, Bencheng Liao, Yiang Shi, Xiaoyang Guo, Yuechuan Pu, Haoran Yin, Xiangyu Li, Xinbang Zhang, et al. Rad: Training an end-to-end driving policy via large-scale 3dgs-based reinforcement learning. *arXiv preprint arXiv:2502.13144*, 2025.

[5] Xiaoxue Chen, Ziyi Xiong, Yuantao Chen, Gen Li, Nan Wang, Hongcheng Luo, Long Chen, Haiyang Sun, Bing Wang, Guang Chen, et al. Dggt: Feedforward 4d reconstruction of dynamic driving scenes using unposed images. *arXiv preprint arXiv:2512.03004*, 2025.

[6] Zhongrui Yu, Zhao Wang, Yijia Xie, Yida Wang, Xueyang Zhang, Yifei Zhan, and Kun Zhan. Streetforward: Perceiving dynamic street with feedforward causal attention. *arXiv preprint arXiv:2603.19552*, 2026.

[7] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025.

[8] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

[9] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, et al. Mapanything: Universal feed-forward metric 3d reconstruction. *arXiv preprint arXiv:2509.13414*, 2025.

[10] Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

[11] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7210–7219, 2021.

[12] Mehdi Hosseinzadeh, Shin-Fang Chng, Yi Xu, Simon Lucey, Ian Reid, and Ravi Garg. G3splat: Geometrically consistent generalizable gaussian splatting. *arXiv preprint arXiv:2512.17547*, 2025.

[13] Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024.

[14] Ziteng Cui, Xuangeng Chu, and Tatsuya Harada. Luminance-gs: Adapting 3d gaussian splatting to challenging lighting conditions with view-adaptive curve adjustment. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26472–26482, 2025.

[15] Isaac Deutsch, Nicolas Moënne-Loccoz, Gavriel State, and Zan Gojcic. Ppisp: Physically-plausible compensation and control of photometric variations in radiance field reconstruction. *arXiv preprint arXiv:2601.18336*, 2026.

[16] Yuehao Wang, Chaoyi Wang, Bingchen Gong, and Tianfan Xue. Bilateral guided radiance field processing. *ACM Transactions on Graphics (TOG)*, 43(4):1–13, 2024.

[17] Nan Wang, Yuantao Chen, Lixing Xiao, Weiqing Xiao, Bohan Li, Zhaoxi Chen, Chongjie Ye, Shaocong Xu, Saining Zhang, Ziyang Yan, et al. Unifying appearance codes and bilateral grids for driving scene gaussian splatting. *arXiv preprint arXiv:2506.05280*, 2025.

[18] Jisu Shin, Richard Shaw, Seunghyun Shin, Zhensong Zhang, Hae-Gon Jeon, and Eduardo Pérez-Pellitero. CHROMA: Consistent harmonization of multi-view appearance via bilateral grid prediction. In *The Fourteenth International Conference on Learning Representations*, 2026.

[19] Chensheng Peng, Chengwei Zhang, Yixiao Wang, Chenfeng Xu, Yichen Xie, Wenzhao Zheng, Kurt Keutzer, Masayoshi Tomizuka, and Wei Zhan. Desire-gs: 4d street gaussians for static-dynamic decomposition and surface reconstruction for urban driving scenes. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6782–6791, 2025.

[20] Hongyu Zhou, Jiahao Shao, Lu Xu, Dongfeng Bai, Weichao Qiu, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugs: Holistic urban 3d scene understanding via gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21336–21345, 2024.

[21] Cong Wang, Xianda Guo, Wenbo Xu, Wei Tian, Ruiqi Song, Chenming Zhang, Lingxi Li, and Long Chen. Drivesplat: Decoupled driving scene reconstruction with geometry-enhanced partitioned neural gaussians. *arXiv preprint arXiv:2508.15376*, 2025.

[22] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[23] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21634–21643, 2024.

[24] Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In *European Conference on Computer Vision*, pages 156–173. Springer, 2024.

[25] Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo De Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, et al. Omnire: Omni urban scene reconstruction. *arXiv preprint arXiv:2408.16760*, 2024.

[26] Jiawei Xu, Kai Deng, Zexin Fan, Shenlong Wang, Jin Xie, and Jian Yang. Ad-gs: Object-aware b-spline gaussian splatting for self-supervised autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24770–24779, 2025.

[27] Jiawei Yang, Jiahui Huang, Yuxiao Chen, Yan Wang, Boyi Li, Yurong You, Apoorva Sharma, Maximilian Igl, Peter Karkus, Danfei Xu, et al. Storm: Spatio-temporal reconstruction model for large-scale outdoor scenes. *arXiv preprint arXiv:2501.00602*, 2024.

[28] Qijian Tian, Xin Tan, Yuan Xie, and Lizhuang Ma. Drivingforward: Feed-forward 3d gaussian splatting for driving scene reconstruction from flexible surround-view input. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7374–7382, 2025.

[29] Qirui Hou, Wenzhang Sun, Chang Zeng, Chunfeng Wang, Hao Li, and Jianxun Cui. Drivingscene: A multi-task online feed-forward 3d gaussian splatting method for dynamic driving scenes. *arXiv preprint arXiv:2510.24734*, 2025.

[30] Haibao Yu, Kuntao Xiao, Jiahang Wang, Ruiyang Hao, Yuxin Huang, Guoran Hu, Haifang Qin, Bowen Jing, Yuntian Bo, and Ping Luo. Recondrive: Fast feed-forward 4d gaussian splatting for autonomous driving scene reconstruction. *arXiv preprint arXiv:2603.07552*, 2026.

[31] Jiawei Yang, Jiahui Huang, Boris Ivanovic, Yuxiao Chen, Yan Wang, Boyi Li, Yurong You, Apoorva Sharma, Maximilian Igl, Peter Karkus, Danfei Xu, Yue Wang, and Marco Pavone. STORM: Spatio-temporal reconstruction model for large-scale outdoor scenes. In *The Thirteenth International Conference on Learning Representations*, 2025.

[32] Jiawen Chen, Sylvain Paris, and Frédo Durand. Real-time edge-aware image processing with the bilateral grid. *ACM Transactions on Graphics (TOG)*, 26(3):103–es, 2007.

[33] Wen Jiang, Boshu Lei, and Kostas Daniilidis. Fisherrf: Active view selection and uncertainty quantification for radiance fields using fisher information. *arXiv preprint arXiv:2311.17874*, 2023.

[34] Mijeong Kim, Jongwoo Lim, and Bohyung Han. 4d gaussian splatting in the wild with uncertainty-aware regularization. *Advances in Neural Information Processing Systems*, 37:129209–129226, 2024.

[35] Weiquan Wang, Feifei Shao, Lin Li, Zhen Wang, Jun Xiao, and Long Chen. Uncertainty-aware 4d gaussian splatting for monocular occluded human rendering. *arXiv preprint arXiv:2602.06343*, 2026.

[36] Ronald A Fisher. On the mathematical foundations of theoretical statistics. *Philosophical transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character*, 222(594-604):309–368, 1922.

[37] Alex Hanson, Allen Tu, Vasu Singla, Mayuka Jayawardhana, Matthias Zwicker, and Tom Goldstein. Pup 3d-gs: Principled uncertainty pruning for 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5949–5958, 2025.

[38] Haiyi Li, Qi Chen, Denis Kalkofen, and Hsiang-Ting Chen. Ougs: Active view selection via object-aware uncertainty estimation in 3dgs. *arXiv preprint arXiv:2511.09397*, 2025.

[39] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020.

[40] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[41] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[42] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[43] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[44] Sheng Miao, Jiaxin Huang, Dongfeng Bai, Xu Yan, Hongyu Zhou, Yue Wang, Bingbing Liu, Andreas Geiger, and Yiyi Liao. Evolsplat: Efficient volume-based gaussian splatting for urban view synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11286–11296, 2025.

[45] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19457–19467. IEEE, 2024.

[46] Hao Lu, Tianshuo Xu, Wenzhao Zheng, Yunpeng Zhang, Wei Zhan, Dalong Du, Masayoshi Tomizuka, Kurt Keutzer, and Yingcong Chen. Drivingrecon: Large 4d gaussian reconstruction model for autonomous driving. *arXiv preprint arXiv:2412.09043*, 2024.

[47] Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European conference on computer vision*, pages 370–386. Springer, 2024.

[48] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16453–16463, 2025.

[49] Dongxu Wei, Zhiqi Li, and Peidong Liu. Omni-scene: Omni-gaussian representation for ego-centric sparse-view scene reconstruction. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22317–22327. IEEE, 2025.

[50] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details, 2025.

[51] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025.

A Technical appendices and supplementary material

This supplementary material provides additional technical details, implementation clarifications, and qualitative results that complement the main paper.

Sec. A.1 discusses the compatibility of MCSplat with existing GFM-based feed-forward driving Gaussian frameworks.

Sec. A.2 explains how Fisher information is computed and used for within-step geometric reweighting.

Sec. A.3 provides detailed definitions of the training loss terms.

Sec. A.4 provides implementation details, including model configuration, training stages, optimization settings, and hyperparameters.

Sec. A.5 describes the construction and quantitative evaluation of the appearance-challenging validation subset for cross-camera photometric inconsistency.

Sec. A.6 presents additional 2D rendering results on the Waymo validation set.

Sec. A.7 presents additional 3D Gaussian visualizations under novel views.

Sec. A.8 presents qualitative cross-lane novel-view synthesis under laterally shifted camera trajectories.

A.1 Compatibility with GFM-Based Feed-Forward Driving Gaussian Frameworks

Existing GFM-based feed-forward driving Gaussian frameworks[5, 1, 6, 30] usually follow a common pipeline: multi-view images are first encoded by a pretrained geometry foundation model, then decoded by a dense prediction or DPT-style Gaussian head to produce image-aligned Gaussian primitives, which are finally supervised through differentiable rendering and reconstruction losses. Although different methods may vary in temporal fusion, Gaussian parameterization, or auxiliary dynamic modules, their core Gaussian prediction process can be abstracted as an image-to-Gaussian feed-forward mapping followed by rasterization, as shown in Fig. 6(a).

MCSplat is designed to be compatible with this general pipeline. As shown in Fig. 6(b), we keep the GFM backbone and Gaussian prediction branch unchanged, and introduce three modular additions: canonical appearance construction provides photometrically consistent inputs and supervision, the multi-scale Bilateral Grid Transformer maps canonical renderings back to raw camera appearance, and Fisher-information-based uncertainty reweights geometric self-supervision. Therefore, MCSplat can enhance existing GFM-based feed-forward Gaussian frameworks by decoupling geometry learning from camera-specific appearance variation while preserving their original feed-forward reconstruction structure. The ablation results in Tab. 4 further validate the effectiveness of the proposed modules.

A.2 Within-Step Fisher Information-Guided Reweighting

A practical question is how Fisher information computed at the current training step can be used to weight geometric regularizers in the same step. Our implementation adopts a single-step online reweighting strategy rather than a recursive optimization. At iteration t, a standard forward pass first produces the current Gaussian set $\mathcal{G}^{(t)}$, rendered images $\hat{I}^{(t)}$, and raw per-Gaussian geometric

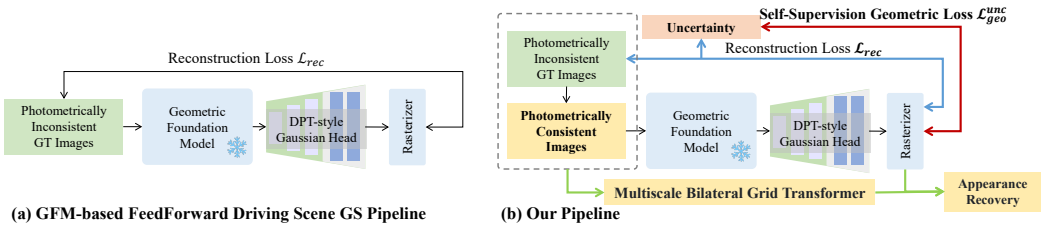


Figure 6: Intuitive pipeline comparison between our method and existing GFM-based feed-forward driving Gaussian frameworks.

penalties $\ell_{i,\text{orient}}^{(t)}$ and $\ell_{i,\text{flat}}^{(t)}$. Before the main backward pass, we query the gradient of the photometric loss with respect to Gaussian centers on the same forward graph to obtain Fisher information.

Specifically, we use the translational Fisher information as the uncertainty cue:

$$F_i^{(t),xyz} = \frac{1}{|\mathcal{V}_{\text{cur}}|} \sum_{q \in \mathcal{V}_{\text{cur}}} \left\| \frac{\partial \mathcal{L}_{\text{photo},q}^{(t)}}{\partial \boldsymbol{\mu}_i^{(t)}} \right\|_2^2, \tag{20}$$

where $\boldsymbol{\mu}_i^{(t)}$ is the center of Gaussian g_i at the current step. We then convert the Fisher information into an uncertainty weight:

$$w_i^{(t)} = \text{stopgrad}\left[\exp\left(-\gamma \frac{F_i^{(t),xyz}}{\max_j F_j^{(t),xyz} + \varepsilon} \right) \right], \tag{21}$$

where $\text{stopgrad}(\cdot)$ denotes the stop-gradient operator. This detachment is essential: the Fisher information serves only as a current-step importance weight and does not introduce higher-order gradients through the computation.

The orientation and flatness losses are then obtained by weighting the raw per-Gaussian penalties from the same forward pass:

$$\mathcal{L}_{\text{orient}}^{(t)} = \frac{\sum_i w_i^{(t)} \, \ell_{i,\text{orient}}^{(t)}}{\sum_i w_i^{(t)} + \varepsilon}, \qquad \mathcal{L}_{\text{flat}}^{(t)} = \frac{\sum_i w_i^{(t)} \, \ell_{i,\text{flat}}^{(t)}}{\sum_i w_i^{(t)} + \varepsilon}. \tag{22}$$

Thus, the current-step Fisher information does not recursively define the geometric loss. Instead, it provides an online estimation of geometric uncertainty that reweights the geometric regularizers before the final loss is backpropagated:

$$\mathcal{L}^{(t)} = \mathcal{L}_{\text{base}}^{(t)} + \lambda_{\text{orient}} \mathcal{L}_{\text{orient}}^{(t)} + \lambda_{\text{flat}} \mathcal{L}_{\text{flat}}^{(t)}, \tag{23}$$

where $\mathcal{L}_{\text{base}}^{(t)}$ denotes the photometric and auxiliary losses used in the corresponding training stage. This design keeps the optimization first-order while allowing geometric self-supervision to adapt to the local uncertainty of the current reconstruction state.

A.3 Full Stage-wise Training Objective

Here we present the extended training objectives used in the two-stage optimization. We denote the current-frame and future-frame view sets as $\mathcal{V}_{\text{cur}}$ and $\mathcal{V}_{\text{fut}}$, respectively.

Stage 1. In the first stage, the GFM backbone is frozen, while the sparse 3D convolution modules and Gaussian head are trained in the canonical appearance space. The bilateral-grid branch is bypassed. The objective is defined as

$$\mathcal{L}_{\text{stage1}} = \sum_{v \in \mathcal{V}_{\text{cur}}} \left(\lambda_1 \mathcal{L}_{\text{mse}}^{v,\text{can}} + \lambda_2 \mathcal{L}_{\text{lpips}}^{v,\text{can}} + \lambda_3 \mathcal{L}_{\text{dyn}}^{v} + \lambda_4 \mathcal{L}_{\text{scale}}^{v} \right) \tag{24}$$

$$+ \sum_{v \in \mathcal{V}_{\text{fut}}} \left(\lambda_1 \mathcal{L}_{\text{mse}}^{v,\text{can}} \odot B^v + \lambda_2 \mathcal{L}_{\text{lpips}}^{v,\text{can}} \right). \tag{25}$$

Here, $\odot$ denotes element-wise multiplication, and the background mask B^v excludes dynamic regions in future-frame supervision to avoid optimization instability.

Stage 2. In the second stage, most of the Gaussian head is frozen except for the Gaussian MLPs, while the bilateral-grid transformer is optimized for appearance recovery. The predicted bilateral grids map the canonical render to the raw image domain as in Eq. (7). The objective is

$$\mathcal{L}_{\text{stage2}} = \mathcal{L}_{\text{stage1}} + \sum_{v \in \mathcal{V}_{\text{cur}} \cup \mathcal{V}_{\text{fut}}} \left(\lambda_5 \mathcal{L}_{\text{mse}}^{v,\text{img}} + \lambda_6 \mathcal{L}_{\text{lpips}}^{v,\text{img}} \right) + \lambda_7 \mathcal{L}_{\text{tv}} + \mathcal{L}_{\text{geo}}^{\text{unc}}. \tag{26}$$

The image-domain losses $\mathcal{L}_{\text{mse}}^{v,\text{img}}$ and $\mathcal{L}_{\text{lpips}}^{v,\text{img}}$ supervise the recovered raw-domain image. The total-variation loss $\mathcal{L}_{\text{tv}}$ regularizes the predicted bilateral grids, and $\mathcal{L}_{\text{geo}}^{\text{unc}}$ regularizes Gaussian geometry using uncertainty-aware orientation and flatness constraints. **We detail these loss terms below.**

Details about each loss term. Let $\hat{I}^v_{\text{can}}$ and I^v_{can} denote the canonical prediction and canonical target, and let $\hat{I}^v_{\text{img}} = \mathbf{T}_{q,p}(\hat{I}^v_{\text{can}})$ be the raw-domain image recovered by the predicted bilateral grids $\mathbf{B}^{(m)}_q$, with target I^v_{img}. Let $B^v(p) \in \{0,1\}$ be the exclusion mask (used only for future-view MSE) for pixel p in the future-view $\mathcal{P}^v$. The individual loss terms can be calculated as

$$\mathcal{L}^{v,\text{can}}_{\text{mse}} = \sum_{p\in\mathcal{P}^v} \left\| \hat{I}^v_{\text{can}}(p) - I^v_{\text{can}}(p) \right\|_2^2, \tag{27}$$

$$\mathcal{L}^{v,\text{can}}_{\text{mse}} \odot B^v = \sum_{p\in\mathcal{P}^v} B^v(p) \left\| \hat{I}^v_{\text{can}}(p) - I^v_{\text{can}}(p) \right\|_2^2, \tag{28}$$

$$\mathcal{L}^{v,\text{can}}_{\text{lpips}} = \text{LPIPS}\Big(\hat{I}^v_{\text{can}}, I^v_{\text{can}}\Big), \tag{29}$$

$$\mathcal{L}^{v}_{\text{dyn}} = -\sum_{p\in\mathcal{P}^v} \Big[\big(1 - B^v(p)\big) \log \hat{D}^v(p) + B^v(p) \log\big(1 - \hat{D}^v(p)\big)\Big], \tag{30}$$

$$\mathcal{L}^{v,\text{img}}_{\text{mse}} = \sum_{p\in\mathcal{P}^v} \left\| \hat{I}^v_{\text{img}}(p) - I^v_{\text{img}}(p) \right\|_2^2, \tag{31}$$

$$\mathcal{L}^{v,\text{img}}_{\text{lpips}} = \text{LPIPS}\Big(\hat{I}^v_{\text{img}}, I^v_{\text{img}}\Big). \tag{32}$$

Since the point clouds estimated by geometry foundation models are often not metrically accurate, we first use the ROE solver from MoGe [51] to compute the optimal scale and shift parameters (s^*_v, t^*_v) between the estimated point cloud and the LiDAR points:

$$(s^*_v, t^*_v) = \arg\min_{s,t} \sum_{p\in\mathcal{P}^v} \frac{1}{Z^v(p)} \Big(s\,\hat{Z}^v_{\text{local}}(p) + t - Z^v(p)\Big)^2. \tag{33}$$

Here, $\hat{Z}^v_{\text{local}}$ denotes the local depth predicted by the Gaussian head before applying the learned scale and shift, and Z^v denotes the corresponding depth supervision. A small MLP predicts scale $\hat{s}^v$ and shift $\hat{t}^v$ from the pooled GFM features:

$$(\hat{s}^v, \hat{t}^v) = \text{MLP}(\text{AvgPool}(\mathbf{F}^{\text{geo}}_v)). \tag{34}$$

The scale-shift loss $\mathcal{L}^v_{\text{scale}}$ then supervises the predicted alignment parameters to match this closed-form optimum:

$$\mathcal{L}^v_{\text{scale}} = \text{SmoothL1}(\log \hat{s}^v, \log s^*_v) + \text{SmoothL1}\big(\hat{t}^v, t^*_v\big). \tag{35}$$

For bilateral-grid smoothness loss $\mathcal{L}_{\text{tv}}$, we regularize the predicted multi-scale bilateral grids with a total-variation penalty. Let $\mathbf{B}^{(m)}_q$ denote the predicted bilateral grid at scale m, where $m = 1, \ldots, M$, and q indexes the corresponding view or camera-conditioned grid. Each grid stores affine color-transform coefficients over a 3D lattice consisting of the guidance/intensity dimension and two spatial dimensions. We penalize finite differences along all three lattice axes:

$$\mathcal{L}_{\text{tv}} = \sum_{m=1}^{M} \frac{1}{N_m} \Bigg(\frac{\Big\| \mathbf{B}^{(m)}_q{}_{:,:,1:,:,:} - \mathbf{B}^{(m)}_q{}_{:,:,:-1,:,:} \Big\|_2^2}{|\nabla_l \mathbf{B}^{(m)}_q|} + \frac{\Big\| \mathbf{B}^{(m)}_q{}_{:,:,:,1:,:} - \mathbf{B}^{(m)}_q{}_{:,:,:,:-1,:} \Big\|_2^2}{|\nabla_h \mathbf{B}^{(m)}_q|} + \frac{\Big\| \mathbf{B}^{(m)}_q{}_{:,:,:,:,1:} - \mathbf{B}^{(m)}_q{}_{:,:,:,:,:-1} \Big\|_2^2}{|\nabla_w \mathbf{B}^{(m)}_q|} \Bigg). \tag{36}$$

Here, ∇_l, ∇_h, and ∇_w denote finite differences along the guidance, height, and width dimensions of the bilateral grid, respectively; $|\nabla_l \mathbf{B}^{(m)}_q|$, $|\nabla_h \mathbf{B}^{(m)}_q|$, and $|\nabla_w \mathbf{B}^{(m)}_q|$ are the numbers of elements in the corresponding difference tensors, and N_m is the number of predicted grids at scale m. This loss encourages neighboring lattice cells to predict similar affine color transforms, suppressing high-frequency artifacts in the recovered raw-domain image while still allowing smooth spatial- and intensity-dependent appearance correction.

For uncertainty-guided geometric self-supervision loss $\mathcal{L}^{\text{unc}}_{\text{geo}}$, please refer to Sec. 3.3.

A.4 Implementation Details

We provide additional implementation details omitted from the main text. We follow the baseline architecture in the use of the GFM backbone, semantic feature extraction, and Gaussian head design. The input and rendering resolution is fixed to 350×518.

Training is performed in two stages. In Stage 1, the GFM backbone is frozen, and only the 3D sparse convolution modules and the Gaussian head are trained. Different from the baseline, both input and supervision use the canonicalized images from Sec. 3.1. In Stage 2, all components are frozen except for the Gaussian MLPs and the Bilateral Grid Transformer. This stage jointly learns bilateral-grid appearance recovery and applies geometric regularization to the trainable Gaussian outputs.

The Bilateral Grid Transformer adopts a DINOv2 [42] ViT-Small image encoder to extract 384-dimensional patch tokens for appearance conditioning. Both the Transformer encoder and decoder contain 3 layers. The multi-scale bilateral grids use three resolutions $[2, 2, 1]$, $[4, 4, 2]$, and $[8, 8, 4]$, with guidance downsampling factors $[4, 4, 2]$, respectively.

The model is optimized using AdamW [43] on 16 NVIDIA H20 GPUs with a total batch size of 32. For the training objective parameters introduced in Sec. 3.4 and above, we set $\lambda_1 = \lambda_5 = 5.0$, $\lambda_2 = \lambda_6 = 0.05$, $\lambda_3 = 0.05$, $\lambda_4 = 0.1$, $\lambda_7 = 1.0$, $\lambda_{\text{orient}} = 0.05$, and $\lambda_{\text{flat}} = 0.5$.

A.5 Validation Subset for Cross-Camera Photometric Inconsistency

Appearance-Inconsistent Subset Selection. To identify Waymo validation scenes with large cross-camera appearance variations, we compute an overlap-based appearance discrepancy score for each scene. We select 57 scenes using these input-derived, model-independent scores to form a challenging validation subset. Instead of comparing full images directly, we only evaluate pixels that have valid geometric correspondences across cameras. The score is computed separately for ground and sky regions.

For ground regions, we follow the same LiDAR-affine correspondence construction pipeline in Sec. 3.1. Given a valid LiDAR-depth pixel in a reference camera, we back-project it to 3D and then project it into the target camera. A match is retained only if it lies inside the target image, passes valid-mask filtering, and satisfies the LiDAR depth consistency check when target depth is available. The target RGB value is obtained by bilinear sampling.

For sky regions, where LiDAR depth is unavailable, we use the same proxy sky-point construction as in Sec. 3.1: each sky pixel is lifted along its camera ray to a fixed sky depth and then projected into the target camera. A correspondence is kept only when both the reference pixel and the projected target pixel are classified as sky.

For each valid correspondence i, let the RGB values in the reference and target cameras be $\mathbf{x}_i = (R_i, G_i, B_i)$ and $\mathbf{y}_i = (R'_i, G'_i, B'_i)$, respectively. We first compute luminance using BT.601 weights:

$$L(\mathbf{x}_i) = 0.299R_i + 0.587G_i + 0.114B_i,$$
$$L(\mathbf{y}_i) = 0.299R'_i + 0.587G'_i + 0.114B'_i.$$

The per-correspondence appearance difference is defined as the absolute log-luminance difference:

$$d_i = \left|\log_2\left(\max(L(\mathbf{x}_i), \epsilon)\right) - \log_2\left(\max(L(\mathbf{y}_i), \epsilon)\right)\right|,$$

where $\epsilon = 10^{-6}$ is used for numerical stability. Equivalently, this measures the absolute logarithmic brightness ratio between two matched observations.

For each scene s and region type $r \in \{\text{ground}, \text{sky}\}$, we average over all valid correspondences $\mathcal{M}_{s,r}$:

$$S_{s,r} = \frac{1}{N_{s,r}} \sum_{i \in \mathcal{M}_{s,r}} d_i, \qquad N_{s,r} = |\mathcal{M}_{s,r}|.$$

The final scene-level score is computed by weighting the ground and sky scores by their numbers of valid correspondences:

$$S_s = \frac{N_{s,\text{ground}} S_{s,\text{ground}} + N_{s,\text{sky}} S_{s,\text{sky}}}{N_{s,\text{ground}} + N_{s,\text{sky}}}.$$

The final selected Waymo Multi-View Appearance-Inconsistent Subset is listed in Tab. 5.

Table 5: Waymo Multi-View Appearance-Inconsistent Subset.

scene_id	seg_name	scene_id	seg_name	scene_id	seg_name
008	seg_107139	054	seg_143831	120	seg_283472
009	seg_108375	059	seg_146633	122	seg_303925
012	seg_110487	060	seg_146873	129	seg_366934
013	seg_110533	065	seg_150215	133	seg_419577
014	seg_113566	067	seg_150569	134	seg_424653
016	seg_114061	072	seg_154962	137	seg_442641
023	seg_121347	076	seg_159595	156	seg_599003
025	seg_123062	078	seg_162133	157	seg_600109
026	seg_123746	082	seg_169798	160	seg_618300
029	seg_128204	084	seg_171355	163	seg_662188
033	seg_131780	088	seg_173440	164	seg_663760
037	seg_133368	098	seg_180241	168	seg_716314
040	seg_134699	099	seg_180457	175	seg_798862
041	seg_135733	109	seg_210580	180	seg_833180
043	seg_139416	110	seg_230820	184	seg_884527
046	seg_140812	113	seg_250679	190	seg_902487
047	seg_141077	115	seg_260994	195	seg_924365
048	seg_141279	118	seg_272435	198	seg_944394
050	seg_142445	119	seg_273637	199	seg_947242

Reconstruction Results on the Appearance-Inconsistent Subset. We compare MCSplat with UniSplat [1] on the 57-scene subset. As shown in Tab. 6, MCSplat improves all three reconstruction metrics in both view configurations. **In the front-camera setting,** PSNR increases from 29.18 to 32.07 dB (+2.89 dB) and LPIPS decreases from 0.15 to 0.12 (-20%). **In the 5 camera setting**, PSNR increases from 29.28 to 30.88 dB (+1.60 dB) and LPIPS decreases from 0.18 to 0.14(-22.2%). The consistent gains on this subset indicate improved robustness to pronounced cross-camera appearance differences.

Table 6: Quantitative reconstruction results on the Waymo Appearance-Inconsistent Subset.

Method	Views	PSNR↑	SSIM↑	LPIPS↓
UniSplat	Front	29.18	0.89	0.15
Ours	Front	**32.07**	**0.90**	**0.12**
UniSplat	Multi	29.28	0.86	0.18
Ours	Multi	**30.88**	**0.88**	**0.14**

A.6 Additional Qualitative Results in Image Space

Additional qualitative results on the Waymo validation set are shown in Figs. 7 to 9. In the challenging rainy driving scene in Fig. 7, our method substantially reduces artifacts in the sky compared with the baseline, demonstrating strong appearance recovery capability. Moreover, the introduced uncertainty-guided geometric regularization improves depth estimation around object boundaries, such as shrubs in Fig. 9, transparent materials such as vehicle windows in Fig. 7, pedestrians in Fig. 8, and texture-rich regions such as the poster in Fig. 8, which is particularly important for cross-lane novel-view synthesis.

A.7 Additional Qualitative Results in 3D Gaussian Space

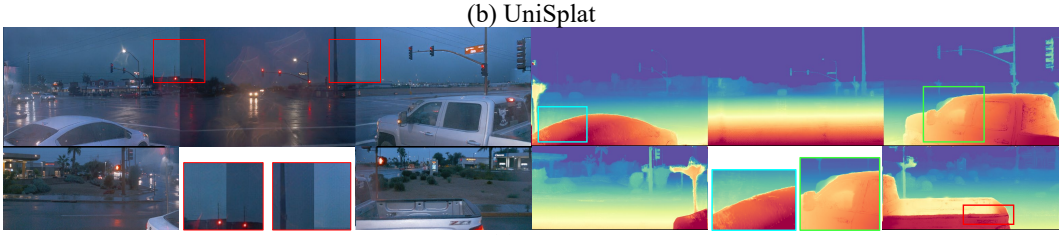
In Figs. 10 to 12, we show qualitative comparison of single-frame Gaussians under a novel view in various scenes. These samples with highlighted regions show that our method recovers the scene more faithfully than the baseline.



(a) Ground Truth

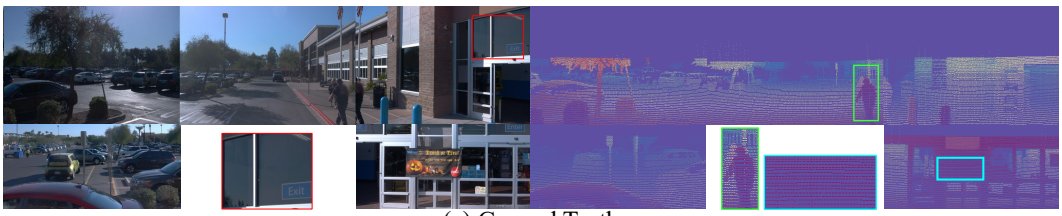


(b) UniSplat



(c) Ours

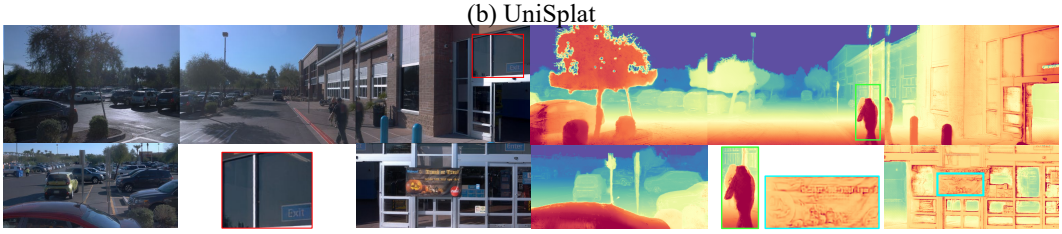
Figure 7: Additional qualitative comparisons under severe cross-camera exposure inconsistency. Our method reduces exposure-induced cross-camera artifacts and alleviates texture-induced depth errors. Around the pickup truck, it avoids confusing the window depth with the background depth, yielding a more plausible depth estimate.



(a) Ground Truth

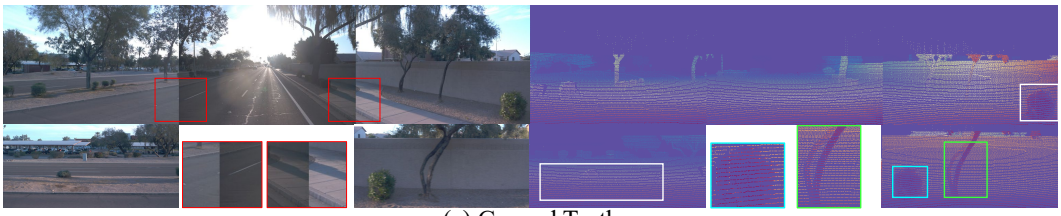


(b) UniSplat

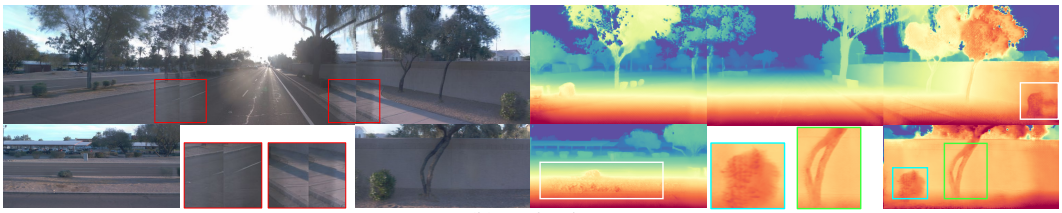


(c) Ours

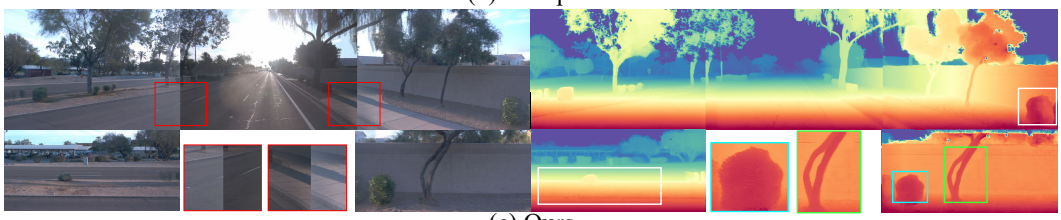
Figure 8: Additional qualitative comparisons. Our method effectively removes glass-surface artifacts in camera-boundary regions. Compared with LiDAR depth, it also yields improved depth estimation in pedestrian regions. On the planar poster surface, our method avoids texture-induced depth ambiguity, producing a more reasonable depth prediction.



(a) Ground Truth

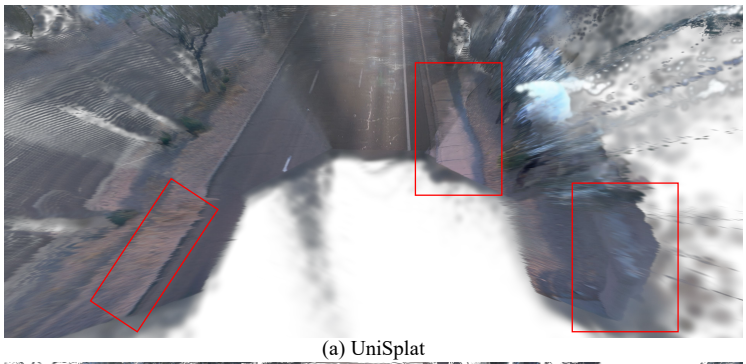


(b) UniSplat

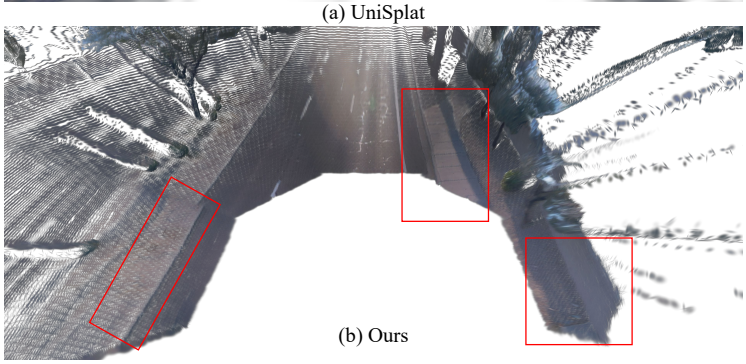


(c) Ours

Figure 9: Additional qualitative comparisons on the Waymo dataset. Our method reduces cross-camera stitching artifacts and produces more reasonable depth estimates, especially around road boundaries and tree or shrub boundaries.

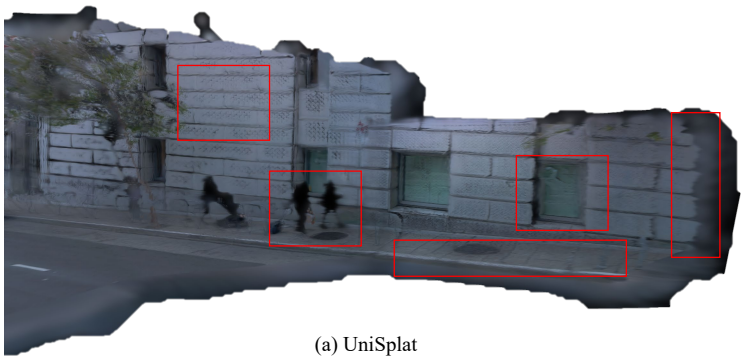


(a) UniSplat

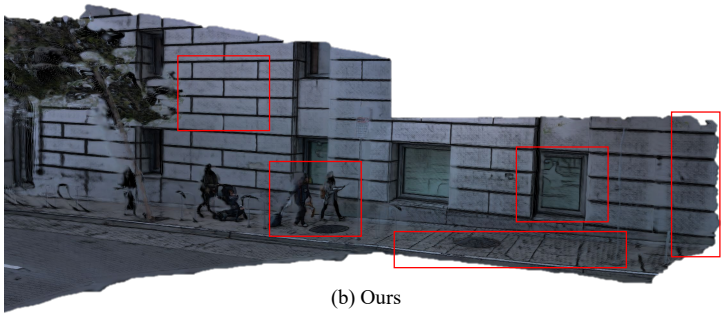


(b) Ours

Figure 10: Qualitative comparison of single-frame Gaussians under a novel view. Compared with the baseline, our method reconstructs cleaner road boundaries and sharper scene geometry. The highlighted regions show that the baseline suffers from geometry–appearance ambiguity caused by multi-view photometric inconsistency, leading to broken road edges, blurred boundary structures, and incomplete ground reconstruction.

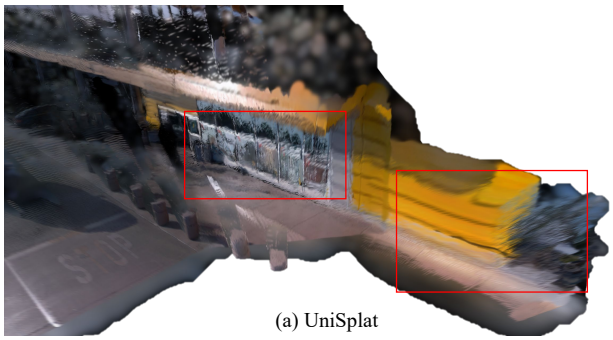


(a) UniSplat

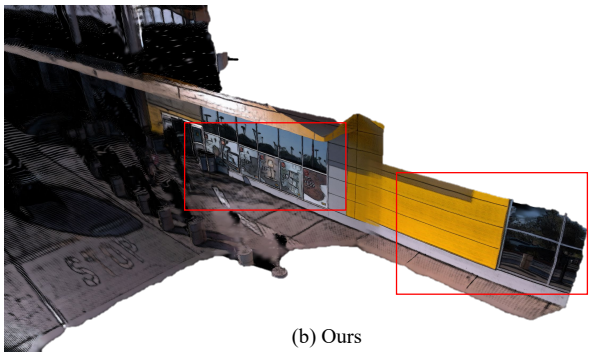


(b) Ours

Figure 11: Qualitative comparison of single-frame Gaussians under a novel view in urban scenes. The highlighted regions show that our method recovers dynamic pedestrians more faithfully than the baseline, while preserving clearer wall textures and more complete ground reconstruction.



(a) UniSplat



(b) Ours

Figure 12: Qualitative comparison of single-frame Gaussians under a novel view. UniSplat tends to confuse texture-rich regions with geometric variations, leading to blurred and distorted wall surfaces. In contrast, our method preserves sharper texture details and more consistent wall geometry in the highlighted regions.

A.8 Cross-Lane Novel-View Synthesis

We evaluate off-trajectory rendering by shifting the camera 1.8 m and 2.0 m leftward (Figs. 13 and 14). Compared with UniSplat, MCSplat preserves clearer vehicle contours and fewer geometric distortions, particularly around the foreground car and curbside motorcycles at 1.8 m, and the pickup trucks and red car at 2.0 m. These results suggest improved geometric consistency under lateral viewpoint shifts, although some artifacts remain in occluded or sparsely observed regions.

B Further Discussion

B.1 Social Impact

Social Impact. MCSplat aims to improve feed-forward 3D reconstruction and novel-view synthesis for autonomous driving scenes. Its potential positive impact lies in enabling more reliable closed-loop simulation, safer evaluation of driving policies, and more scalable reconstruction of complex urban environments. By reducing geometry–appearance ambiguity and improving novel-view geometric consistency, the method may help build more faithful simulation assets for testing autonomous systems before real-world deployment.

However, the method should not be viewed as a substitute for real-world safety validation. Reconstruction errors may still occur in rare weather, unusual lighting, long-tail objects, or sensor configurations outside the training distribution. If such reconstructed scenes are used uncritically for safety evaluation, they may lead to overconfidence in downstream autonomous driving systems. In addition, high-fidelity scene reconstruction techniques may raise privacy concerns if applied to data containing identifiable people, vehicles, or locations. Our experiments are conducted on established autonomous driving datasets, and the method is intended for research on reconstruction and simulation rather than surveillance or identification. Responsible deployment should include dataset governance, privacy-preserving data handling, and careful validation across diverse driving conditions.

B.2 Limitations

Although MCSplat demonstrates strong cross-camera appearance consistency and camera-specific appearance recovery, its performance can still be affected by the limited appearance diversity of the Waymo dataset. In a few scenes, the recovered raw-domain appearance may be less accurate than expected and exhibit color shifts. A promising direction is to improve appearance coverage through stronger data augmentation and larger, more diverse training datasets.



(a) Original trajectory



(b) Shifted Left by 1.8 m-UniSplat



(c) Shifted Left by 1.8 m-Ours

Figure 13: Qualitative cross-lane novel-view synthesis under a 1.8 m leftward camera-trajectory shift. (a) Original trajectory; (b) shifted view rendered by UniSplat; (c) shifted view rendered by MCSplat. The highlighted regions show improved preservation of the left-foreground car and the parked motorcycles along the right sidewalk.



(a) Original trajectory



(b) Shifted Left by 2.0 m-UniSplat



(c) Shifted Left by 2.0 m-Ours

Figure 14: Qualitative cross-lane novel-view synthesis under a 2.0 m leftward camera-trajectory shift. (a) Original trajectory; (b) shifted view rendered by UniSplat; (c) shifted view rendered by MCSplat. Compared with UniSplat, MCSplat better preserves the highlighted pickup-truck rear structures and the red foreground car, although off-trajectory artifacts remain.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the two main challenges addressed by the paper—cross-camera geometry–appearance ambiguity and unreliable local geometric supervision—and describe the corresponding proposed components, including canonical appearance construction, bilateral-grid appearance recovery, and Fisher-guided geometric self-supervision. The claims are supported by the experiments on the Waymo validation set and the appearance-challenging subset, and the scope is limited to feed-forward Gaussian splatting for multi-view driving-scene reconstruction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations of the work have been discussed in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: N/A. The paper does not present theoretical results or formal proofs. The main claims are supported by empirical experiments, including quantitative comparisons, ablation studies, and qualitative results on the Waymo dataset.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes. The paper describes the dataset split, evaluation protocol, metrics, baseline comparison setting, implementation details, training stages, optimization settings, loss terms, and ablation configurations needed to reproduce the main experimental results. Additional details, including the construction of the appearance-challenging validation subset and loss definitions, are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper uses the public Waymo Open Dataset, and the experimental protocol, training settings, loss terms, and evaluation details are described in the main text and supplementary material. The source code is not released at submission time to preserve anonymity, but it will be made publicly available upon publication with instructions for reproducing the main results.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes. The paper specifies the Waymo training/validation splits, evaluation protocol, image resolution, training stages, optimizer, batch size, hardware setup, loss weights, and key hyperparameters. Additional implementation details and loss definitions are provided in the supplementary material to support understanding and reproduction of the results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The experiments are benchmark evaluations with fixed test sets and deterministic metrics, so statistical variation is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the main compute resources used for training, including the GPU type and memory, number of GPUs, total batch size, and training configuration. These details provide sufficient information to estimate the resources needed to reproduce the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have reviewed the NeurIPS Code of Ethics and confirm that the research described in this paper complies with the Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses potential positive impacts, including improved closed-loop simulation and safer evaluation of autonomous driving systems, as well as potential negative impacts, such as overreliance on imperfect reconstructed scenes and privacy concerns in high-fidelity scene reconstruction.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release high-risk models such as language models, image generators, or scraped datasets. The work uses established autonomous driving datasets for reconstruction research, and the code will be released for reproducibility after publication.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper properly credits the datasets, baseline methods, pretrained models, and evaluation metrics used in this work through citations. We use the Waymo Open Dataset and existing model components according to their intended research use, and will respect the corresponding licenses and terms of use when releasing code and instructions.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces code for MCSplat, which will be released upon publication with documentation and instructions for reproducing the main experimental results. The paper does not introduce a new dataset; the appearance-challenging subset is constructed from the Waymo validation set and its selection procedure is documented in the appendix.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects, so no participant instructions, screenshots, or compensation details are applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects or study participants, so there are no participant risks or IRB approval requirements applicable to this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were not used as part of the core methodology or experiments. Any LLM usage was limited to writing assistance and did not impact the scientific content.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.