





# Geometry-Preserving in 3D Gaussian Splatting for LiDAR-Camera Extrinsic Calibration

Kyoleen Kwak<sup>1</sup>, Daeho Kim<sup>1</sup>, Jeong Woon Lee<sup>1</sup>, and Hyoseok Hwang<sup>\*1</sup>

College of Software, Kyung Hee University, Republic of Korea  
{2007kk1, kdh2769, everyman123, hyoseok}@khu.ac.kr

**Abstract.** Accurate LiDAR-camera calibration is essential for robust multi-modal perception. Targetless approaches avoid manual setup but remain limited by the scarcity of discriminative cross-modal features. Recent methods address this by reconstructing the scene within a differentiable model, enabling extrinsic optimization through dense photometric supervision. Among these, 3D Gaussian Splatting (3DGS) has been widely adopted as a geometric proxy that bridges LiDAR and camera within a single differentiable framework. However, since 3DGS was originally designed for novel view synthesis, existing methods tend to prioritize rendering quality, causing the proxy geometry to drift from the true LiDAR structure. We propose a framework that preserves the metric geometry of the Gaussian proxy by aggregating multi-view LiDAR observations for dense depth supervision and blocking photometric gradients from updating the Gaussian spatial parameters. We validate our method on public driving datasets, where it consistently outperforms existing targetless methods in calibration accuracy. Code and dataset are available at <https://github.com/AIRLABkhu/GeoP-Calib>

**Keywords:** LiDAR-Camera Calibration · 3D Gaussian Splatting · Targetless Calibration

## 1 Introduction

Autonomous systems, ranging from self-driving vehicles to mobile robots, rely on multi-modal sensor fusion for robust environmental perception [2, 4, 26, 28]. Among various sensor combinations, the integration of cameras and LiDAR is widely adopted, as cameras provide dense semantic information while LiDAR captures precise 3D geometric structures [6, 27]. By combining these two modalities, perception systems can compensate for the limitations of each sensor, such as the scale ambiguity in monocular vision or the spatial sparsity of LiDAR data [23, 25]. However, the effectiveness of this sensor fusion depends on accurate spatial alignment between the sensors, which is known as LiDAR-Camera extrinsic calibration.

For such spatial alignment, target-based methods [15, 32, 36] have served as the standard approach. These methods rely on physical targets with known geometry, such as checkerboards, to establish precise correspondences between the

---

\* Corresponding author.

two sensors. While target-based methods achieve high accuracy under controlled conditions, their need for careful manual setup and prepared environments limits practical deployment and makes online recalibration difficult, especially for correcting extrinsic misalignment that accumulates from thermal and mechanical stress during long-term operation [19, 29, 39].

Motivated by this overhead, targetless calibration methods have been actively explored [17, 31, 34]. Rather than relying on artificial markers, these approaches leverage shared geometric primitives such as edges and structural lines that can be extracted from both LiDAR and camera data. However, the accuracy of conventional targetless methods is constrained by the scarcity of such discriminative features in typical driving scenes, especially in texture-poor or structurally repetitive environments [33]. To overcome this limitation, recent approaches have shifted from extracting individual features to representing the entire scene with a differentiable model, enabling extrinsic optimization through dense pixel-level supervision rather than sparse feature correspondences.

Neural Radiance Fields for View Synthesis (NeRF) [20] exemplifies this direction, but the computational burden of volumetric rendering often makes NeRF-based calibration [30, 38] impractical for real-time or large-scale use. Recently, 3D Gaussian Splatting (3DGS) [14] has emerged as an efficient scene representation that enables fast, differentiable rendering through explicit point-based primitives. Since each Gaussian can be initialized from a LiDAR point, inheriting its precise metric position, and then rendered onto the camera image plane through differentiable rasterization, 3DGS can effectively function as a geometric proxy that bridges the LiDAR coordinate system and the camera image domain.

This proxy property provides a clear advantage for LiDAR-Camera extrinsic calibration, and several recent methods have adopted 3DGS for this task [7, 12, 33, 37]. These methods share a similar pipeline in which Gaussians are initialized from LiDAR points, rendered into images through the Gaussian proxy, and jointly optimized with the extrinsic parameters by minimizing image-domain losses. Since 3DGS was originally designed for novel view synthesis, its optimization is inherently driven toward improving rendering quality, and existing calibration methods naturally inherit this tendency. However, this rendering-oriented optimization freely adjusts Gaussian positions and shapes to reduce photometric residuals, even at the cost of distorting the underlying geometric structure. Although existing methods constrain the proxy geometry using sparse depth supervision from a single LiDAR scan at each timestep, this coverage is insufficient to prevent the proxy’s geometric parameters from drifting away from the true LiDAR structure during joint optimization. As a result, calibration accuracy deteriorates despite improved or visually plausible rendering. We refer to this phenomenon as *Geometric Decay*, a gradual loss of metric faithfulness in the proxy geometry that degrades calibration accuracy and stability.

To address this problem, we propose **Geometry-Preserving Calibration** (GeoP-Calib), a framework that fully exploits the geometric proxy property of 3DGS for extrinsic calibration. First, we introduce Dense Depth Anchoring (DDA), which aggregates multi-view LiDAR observations to construct a dense depth

prior. DDA increases the spatial coverage of depth supervision, providing a rigid metric foundation that penalizes geometric drift in regions where a single scan offers no constraint. To handle occlusion artifacts that arise from multi-view accumulation, DDA further introduces a Volumetric Soft Mask (VSM) that uses the rendered depth of the Gaussian proxy as a continuous visibility prior, down-weighting occluded LiDAR projections and resolving spatial ambiguities. Second, we design Gradient Decoupling (GD), which blocks the backpropagation of photometric gradients to the Gaussian spatial parameters, specifically position and covariance. GD prevents texture-driven residuals from deforming the proxy geometry, effectively mitigating *Geometric Decay* and preserving the bridge role of the proxy between LiDAR and camera.

In summary, our key contributions are: (i) We identify and analyze a photometric-geometric conflict in 3DGS-based calibration, and empirically demonstrate that photometric supervision can induce *Geometric Decay* of the Gaussian proxy. (ii) We propose GeoP-Calib, a novel framework that integrates Dense Depth Anchoring and Gradient Decoupling to mitigate *Geometric Decay* of the Gaussian proxy. (iii) Extensive evaluations on KITTI odometry [5] and KITTI-360 [18] demonstrate consistent improvements in calibration accuracy, with particularly notable gains in translation estimation.

## 2 Related Works

### 2.1 Target-Based LiDAR-Camera Calibration

Target-based methods estimate extrinsics by observing artificial objects with known geometry, such as checkerboards or fiducial markers, and solving for the transformation via explicit 2D-3D correspondences. Zhang and Pless [32] established the foundational point-to-plane formulation for LiDAR-Camera pairs, and Zhou *et al.* [36] relaxed the target constraints by exploiting multimodal edge line and plane correspondences. Another line of research, Beltrán *et al.* [1] diversified target-based calibration with circular targets, and Kim *et al.* [15] further improved it by more accurate circle-center localization. Despite their precision, target-based methods require meticulous manual setup, limiting their scalability for deployments where sensor extrinsics drift over time.

### 2.2 Targetless LiDAR-Camera Calibration

Targetless methods have exploited naturally occurring scene structures, avoiding dedicated calibration objects. One line of work aligned geometric features across modalities, such as edges [13, 31], 3D-2D lines [34], and cross-modal structural correspondences [21]. Another direction optimized photometric consistency directly via mutual information or intensity-based objectives [10, 16, 22]. On the learning side, Iyer *et al.* [11] and Lv *et al.* [19] regressed extrinsics from raw LiDAR-image pairs using CNNs and cost volumes, while Zhang *et al.* [35] more recently combined foundation-model-based monocular depth with LiDAR intensity alignment. Despite their flexibility, these approaches remain limited by the

inherent sparsity of LiDAR point clouds, which restricts both the availability of discriminative features and the reliability of cross-modal comparison [33].

### 2.3 Scene Reconstruction-Based LiDAR-Camera Calibration

Scene reconstruction-based methods formulate calibration as a joint optimization of a differentiable scene representation and extrinsic parameters, where the photometric loss between rendered and observed images drives pose refinement. NeRF-based approaches [8, 38] train neural radiance fields while jointly optimizing extrinsics, but the volumetric rendering pipeline incurs substantial computational cost, often requiring several hours per calibration. To overcome this efficiency bottleneck, recent methods adopt 3DGS [14] as the underlying representation. Herau *et al.* [7] first applied 3DGS to LiDAR-Camera calibration by fixing Gaussians on LiDAR points and jointly optimizing the scene model and extrinsics. Zhou *et al.* [37] adopted 2D Gaussian Splatting [9] and refined extrinsics through reprojection and triangulation losses. Zhang *et al.* [33] decoupled scene modeling from extrinsic refinement via a hierarchical coarse-to-fine architecture. Jung *et al.* [12] introduced fixed anchor Gaussians and learnable auxiliary Gaussians with a camera rig optimization strategy for consistent multi-view calibration.

## 3 Problem Definition

### 3.1 3DGS Preliminaries

3DGS [14] represents a 3D scene using a set of anisotropic 3D Gaussians,  $\mathcal{G} = \{G_i\}_{i=1}^N$ . Each Gaussian  $G_i$  is parameterized by its mean  $\mu_i \in \mathbb{R}^3$ , covariance  $\Sigma_i \in \mathbb{R}^{3 \times 3}$ , opacity  $\alpha_i \in (0, 1)$ , and color coefficients  $c_i$ . The covariance matrix  $\Sigma_i$  is typically decomposed into a rotation matrix  $R_i$  and a scaling matrix  $S_i$  as  $\Sigma_i = R_i S_i S_i^\top R_i^\top$  to maintain its physical validity during optimization.

To render the scene from a specific viewpoint, the 3D Gaussians are projected onto the 2D image plane. Given World to Camera  $\mathbf{T}_{cw} \in SE(3)$  and intrinsics  $\mathbf{K}$ , each 3D Gaussian is projected to form a 2D Gaussian density  $G_i^{2D}(\mathbf{p})$  evaluated at pixel  $\mathbf{p} \in \mathbb{R}^2$ .

3DGS utilizes tile-based rasterization with alpha-compositing to render images in a differentiable manner. For each pixel  $\mathbf{p}$ , the rendered color  $\Phi_{color}$  and depth  $\Phi_{depth}$  are computed by blending the  $N$  Gaussians overlapping that pixel:

$$\Phi_{color}(\mathcal{G}, \mathbf{T}_{cw})(\mathbf{p}) = \sum_{i=1}^N T_i(\mathbf{p}) \alpha_i G_i^{2D}(\mathbf{p}) c_i, \quad (1)$$

$$\Phi_{depth}(\mathcal{G}, \mathbf{T}_{cw})(\mathbf{p}) = \sum_{i=1}^N T_i(\mathbf{p}) \alpha_i G_i^{2D}(\mathbf{p}) d_i, \quad (2)$$

where  $T_i(\mathbf{p}) = \prod_{j=1}^{i-1} (1 - \alpha_j G_j^{2D}(\mathbf{p}))$  represents the accumulated transmittance, and  $d_i$  denotes the depth of the  $i$ -th Gaussian in the camera coordinate system.

The entire rendering pipeline is fully differentiable with respect to both the scene parameters  $\mathcal{G}$  and the camera pose  $\mathbf{T}_{cw}$ , enabling gradient-based optimization for calibration tasks.

### 3.2 3DGS-based Calibration Preliminaries

The objective of 3DGS-based LiDAR-Camera calibration is to estimate the optimal scene representation and the extrinsic transformation by maximizing the joint posterior probability [33]. Given a set of camera images  $\mathcal{I} = \{I^t\}$ , a set of LiDAR point clouds  $\mathcal{P} = \{\mathcal{P}^t\}$ , and the world to LiDAR poses  $\mathbf{T}_{lw} = \{\mathbf{T}_{lw}^t\}$ , the optimization problem is formulated as:

$$\hat{\mathcal{G}}, \hat{\mathbf{T}}_{cl} = \arg \max_{\mathcal{G}, \mathbf{T}_{cl}} P(\mathcal{G}, \mathbf{T}_{cl} \mid \mathcal{I}, \mathcal{P}, \mathbf{T}_{lw}), \quad (3)$$

where  $\mathcal{G}$  denotes the set of 3D Gaussian parameters, and  $\mathbf{T}_{cl} \in SE(3)$  represents the extrinsic transformation from the LiDAR coordinate system ( $l$ ) to the camera coordinate system ( $c$ ), parameterized by a rotation matrix  $\mathbf{R} \in SO(3)$  and a translation vector  $\mathbf{t} \in \mathbb{R}^3$ .

To solve this MAP estimation, existing frameworks typically minimize a multi-modal objective function. The extrinsic parameters are optimized through a differentiable rendering process  $\Phi(\mathcal{G}, \mathbf{T}_{cw})$ , where the camera pose at time  $t$  is defined as a function of the calibration target:  $\mathbf{T}_{cw}^t = \mathbf{T}_{cl} \cdot \mathbf{T}_{lw}^t$ .

**Photometric Alignment.** The first objective ensures that the rendered appearance matches the observed images. The photometric alignment loss  $\mathcal{L}_{pho}$  is generally defined using a distance metric  $\mathcal{D}_{photo}$ :

$$\mathcal{L}_{pho} = \mathcal{D}_{photo}\left(I^t, \Phi_{color}(\mathcal{G}, \mathbf{T}_{cl}\mathbf{T}_{lw}^t)\right), \quad (4)$$

where  $\mathcal{D}_{photo}$  measures the visual discrepancy. In practice, this metric typically employs the pixel-wise  $L_1$  distance to ensure absolute color alignment and minimize rendering residuals across the observed views [7, 12, 33, 37].

**Cross-view Geometric Consistency.** Recent works further regularize the extrinsics using reprojection consistency [33, 37]. Utilizing the rendered depth map  $D^t = \Phi_{depth}(\mathcal{G}, \mathbf{T}_{cl}\mathbf{T}_{lw}^t)$ , a pixel  $\mathbf{p}^t$  in the source frame  $t$  can be warped to the target frame  $t + s$  as follows:

$$\tilde{\mathbf{p}}^{t+s} = \pi\left(\mathbf{K}(\mathbf{T}_{cl}\mathbf{T}_{lw}^{t+s})(\mathbf{T}_{cl}\mathbf{T}_{lw}^t)^{-1}(D^t(\mathbf{p}^t)\mathbf{K}^{-1}\bar{\mathbf{p}}^t)\right), \quad (5)$$

where  $\mathbf{K}$  is the camera intrinsic matrix,  $\bar{\mathbf{p}}^t$  is the homogeneous coordinate of  $\mathbf{p}^t$ , and  $\pi(\cdot)$  denotes the perspective projection function that divides by the depth component. The reprojection loss  $\mathcal{L}_{rep}$  minimizes the intensity discrepancy between the warped and target pixels:

$$\mathcal{L}_{rep} = \mathcal{D}_{rep}\left(I^{t+s}(\tilde{\mathbf{p}}^{t+s}), I^t(\mathbf{p}^t)\right), \quad (6)$$

where  $\mathcal{D}_{rep}$  represents a photometric distance function.

**Optimization through Gaussian Proxy.** In 3DGS-based calibration, the extrinsic  $\mathbf{T}_{cl}$  is refined by minimizing image-domain objectives such as photometric alignment and cross-view reprojection consistency. We define the image-domain objective as

$$\mathcal{L}_{img} = \mathcal{L}_{pho} + \lambda_{rep}\mathcal{L}_{rep}. \quad (7)$$

Since both  $\mathcal{L}_{pho}$  and  $\mathcal{L}_{rep}$  are computed from rendered outputs, their gradients with respect to  $\mathbf{T}_{cl}$  flow through the differentiable renderer:

$$\frac{\partial \mathcal{L}_{img}}{\partial \mathbf{T}_{cl}} = \frac{\partial \mathcal{L}_{img}}{\partial \Phi(\mathcal{G}, \mathbf{T}_{cw})} \cdot \frac{\partial \Phi(\mathcal{G}, \mathbf{T}_{cw})}{\partial \mathbf{T}_{cl}}, \quad (8)$$

where  $\mathbf{T}_{cw} = \mathbf{T}_{cl}\mathbf{T}_{lw}$ . Importantly, the renderer Jacobian  $\partial\Phi(\mathcal{G}, \mathbf{T}_{cw})/\partial\mathbf{T}_{cl}$  depends on the current proxy state  $\mathcal{G}$ , implying that the reliability of the extrinsic update is tightly coupled to the geometric faithfulness of  $\mathcal{G}$  with respect to the underlying LiDAR structure  $\mathcal{P}$ .

**LiDAR Depth Anchoring.** To explicitly constrain the proxy geometry in metric space, prior 3DGS-based calibration pipelines commonly include a LiDAR-derived depth anchoring term [33, 37]. In the standard setting, this term is instantiated from a single LiDAR scan at time  $t$  by projecting LiDAR points onto the image plane and comparing the rendered depth with the LiDAR depth at valid projection pixels. Since the extrinsic translation is still uncertain during optimization, the depth rendering adopts a rotation-only extrinsic  $\bar{\mathbf{T}}_{cl} = [\mathbf{R}_{cl} \mid \mathbf{0}]$ , isolating the depth supervision from the translation uncertainty by keeping both measurements at the LiDAR origin. The depth anchoring loss is then formulated as:

$$\mathcal{L}_{depth}^{sparse} = \frac{1}{|\Omega^t|} \sum_{\mathbf{p} \in \Omega^t} \mathcal{D}_{depth}(D_{rend}(\mathbf{p}), D_l(\mathbf{p})), \quad (9)$$

where  $\Omega^t$  denotes the set of valid projected pixels,  $D_{rend} = \Phi_{depth}(\mathcal{G}, \bar{\mathbf{T}}_{cl}\mathbf{T}_{lw}^t)$  is the depth rendered from the virtual viewpoint at the LiDAR origin with the camera orientation, and  $D_l$  is the depth obtained from the projected LiDAR points. We denote  $\mathcal{L}_{depth} = \mathcal{L}_{depth}^{sparse}$  to represent the depth objective in existing baselines.

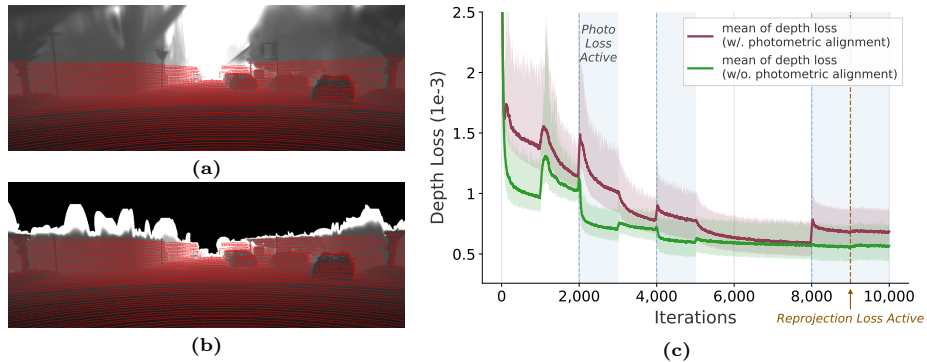
**Overall Objective.** The full objective minimized in typical 3DGS-based calibration pipelines is a weighted sum of the photometric alignment, cross-view consistency, and geometric constraints:

$$\mathcal{L}_{total} = \mathcal{L}_{pho} + \lambda_{rep}\mathcal{L}_{rep} + \lambda_{depth}\mathcal{L}_{depth}, \quad (10)$$

where  $\lambda_{rep}$  and  $\lambda_{depth}$  are balancing weights.

### 3.3 Limitations of Sparse LiDAR Supervision

As Eq. 9 supervises depth only at pixels where LiDAR points project, a single scan provides supervision over only a small fraction of the image, leaving most regions unconstrained in metric space. Notably, rendered depth is a view-dependent output produced at rendering time,  $D_{rend} = \Phi_{depth}(\mathcal{G}, \bar{\mathbf{T}}_{cl}\mathbf{T}_{lw}^t)$ . Thus,



**Fig. 1:** Illustration of Geometric Decay: (a) Rendered depth map of 3DGS optimized using Base pipeline with photometric alignment loss. (b) Rendered depth map of 3DGS optimized with only depth loss. (c) Depth loss on KITTI-360 Seq. 1. Red curve uses scheduled photometric alignment, while green curve never uses it; shaded bands mark photometric-active intervals, and the envelope shows the 100-iteration min–max range.

outside the supervised set  $\Omega^t$ , the proxy geometry can change without being directly penalized by the depth objective. As a result, many Gaussians can drift in unconstrained areas while still producing plausible renderings, which undermines the geometric faithfulness of the proxy and can mislead extrinsic updates.

### 3.4 Analysis of Photometric-Geometric Conflict

In 3DGS-based LiDAR-Camera calibration, the extrinsic parameters are optimized through image-domain objectives in Eq. 4 and Eq. 6, and the resulting gradients backpropagate through the Gaussian proxy. Many pipelines intentionally limit Gaussian density, for instance, by tying the primitive budget to LiDAR voxelization or point sampling. [7, 12, 33, 37] This constraint is designed to preserve metric correspondence with the LiDAR initialization and prevent the model from introducing excessive geometric degrees of freedom to compensate for photometric errors. [7] Under this constrained proxy resolution, photometric observations containing fine textures cannot always be fully explained by appearance parameters alone. Consequently, the optimizer may reduce photometric error by modifying geometric parameters such as Gaussian means and covariances, trading metric geometric fidelity for visually plausible renderings. Combined with the insufficient LiDAR supervision discussed above, this texture-driven photometric-geometric conflict leads to *Geometric Decay*, where the Gaussian metric structure loses faithfulness to the underlying LiDAR points.

Fig. 1 illustrates this phenomenon. Fig. 1a shows rendered depth appearing outside LiDAR-supported regions, whereas Fig. 1b keeps rendered depth more localized to regions where LiDAR points exist. Fig. 1c plots the depth loss, defined as an  $L_1$  discrepancy between rendered depth and LiDAR depth in the inverse-depth domain. During the photometric-active intervals, the run with

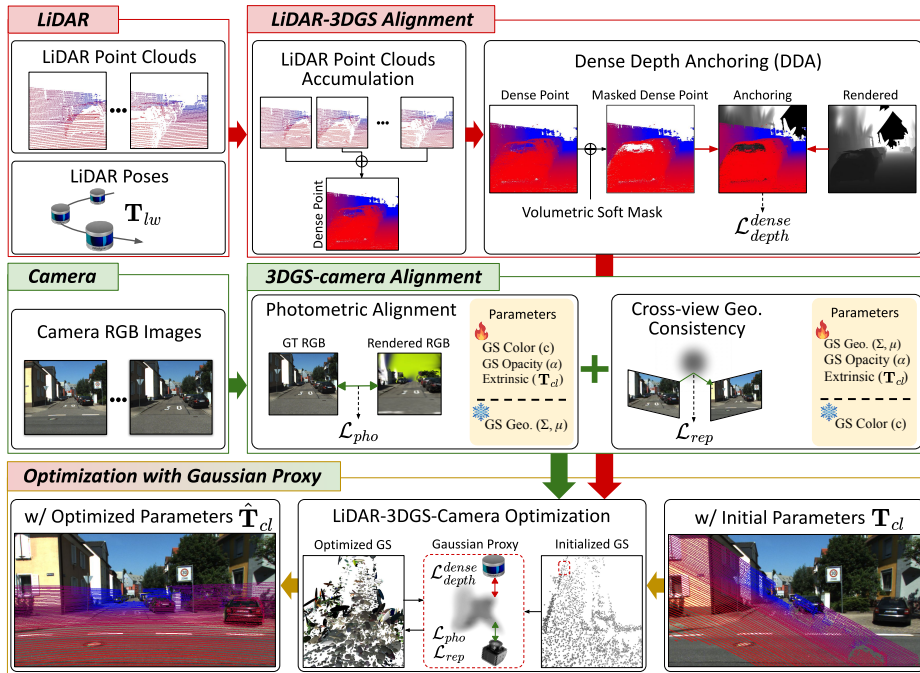


Fig. 2: Overview of the proposed geometry-preserving calibration integration.

photometric alignment shows a higher mean loss and larger variability than the run without it. The curves become close near 8000 iterations only while photometric alignment is inactive, and diverge again once it is reactivated. These results indicate that optimizing photometric consistency can degrade the geometric accuracy required for LiDAR-Camera calibration.

## 4 Geometry-Preserving LiDAR-Camera Calibration

We propose Geometry-Preserving LiDAR-Camera Calibration (GeoP-Calib), a framework that preserves LiDAR-consistent proxy geometry throughout joint optimization. GeoP-Calib addresses the conflict identified in Sec. 3 through two complementary components: (1) Dense Depth Anchoring (DDA), which establishes a rigid metric foundation by aggregating multi-view LiDAR observations, and (2) Gradient Decoupling (GD), which blocks the gradients of the photometric alignment loss  $\mathcal{L}_{pho}$  from updating the Gaussian spatial parameters, thereby mitigating *Geometric Decay*. These two components operate cooperatively while keeping their gradient effects decoupled. DDA governs the spatial parameters of the proxy, while GD allows photometric alignment to refine only appearance and sensor extrinsics without interfering with the established geometry.

#### 4.1 Dense Depth Anchoring with Volumetric Masking

To resolve the structural instability caused by the sparsity of single-scan LiDAR data, we propose **Dense Depth Anchoring** (DDA). This module provides a continuous geometric reference by accumulating multi-view LiDAR observations, ensuring that the Gaussian proxy remains rigidly aligned with the global scene structure.

**Geometric Foundation via Point Accumulation.** As identified in Sec. 3.3, the lack of spatial constraints in unsupervised void regions leads to Gaussian drift. We mitigate this by constructing a dense global point cloud  $\mathcal{P}_{global} = \bigcup_t (\mathbf{T}_{lw}^t)^{-1} \mathcal{P}^t$ . By projecting  $\mathcal{P}_{global}$  from the LiDAR viewpoint onto the image plane, we generate a dense depth prior  $D_l^{dense}$  that covers a significantly larger portion of the image space than a single scan. This dense anchor effectively locks the spatial parameters  $(\mu, \Sigma)$  of the Gaussians, preventing them from drifting even in regions where the current frame’s LiDAR data is absent.

**Volumetric Soft Mask for Occlusion Handling.** A fundamental challenge in dense point accumulation is the *occlusion ambiguity*, where points from distant viewpoints are incorrectly projected onto foreground objects. To effectively resolve this, we leverage the volumetric nature of 3DGS. Since multi-view reprojection evaluates structural consistency across different camera poses, it naturally encourages Gaussians to accurately fill valid spatial regions. This inherent spatial awareness makes the splatted Gaussians a useful geometric cue for reasoning about occlusions.

Building upon this insight, we propose utilizing the 3DGS representation itself as a continuous occlusion filter, introducing a **Volumetric Soft Mask** (VSM). Because the rendered depth  $D_{rend}$  represents the expected ray termination, it implicitly captures the visibility state of the scene. We formulate the visibility weight  $W_{vis}$  to adaptively filter the accumulated points as follows:

$$W_{vis}(\mathbf{p}) = \sigma(\beta \cdot (D_{rend}(\mathbf{p})(1 + \tau) - D_l^{dense}(\mathbf{p}))), \quad (11)$$

where  $\sigma(\cdot)$  is the sigmoid function,  $\tau$  is a depth-proportional tolerance margin, and  $\beta$  controls the sharpness of the transition. By smoothly penalizing points located significantly behind the rendered Gaussian surface ( $D_l^{dense} \gg D_{rend}$ ), this soft masking strategy effectively neutralizes invalid projections while maintaining a stable, differentiable loss landscape.

**Dense Depth Consistency Loss.** Using the accumulated points projected onto the current view, we explicitly enforce consistency between the rendered depth and the LiDAR-implied depth at the corresponding pixel locations. Let  $\Omega^{dense}$  denote the set of valid projected LiDAR points, and let  $M(\mathbf{p}) \in \{0, 1\}$  be a binary validity mask indicating whether the rendered depth is defined at pixel  $\mathbf{p}$ . We compute the depth residual in the inverse-depth domain to provide a more stable parameterization across a wide depth range [3]:

$$\Delta d^{inv}(\mathbf{p}) = |d_{rend}^{inv}(\mathbf{p}) - d_l^{inv}(\mathbf{p})|, \quad d_l^{inv}(\mathbf{p}) = \frac{1}{D_l^{dense}(\mathbf{p}) + \epsilon}, \quad (12)$$

where  $d_{rend}^{inv}(\mathbf{p})$  denotes the rendered inverse depth at  $\mathbf{p}$ , obtained using the same rotation-only extrinsic  $\bar{\mathbf{T}}_{cl}$  as in Eq. 9. The final dense depth anchoring loss combines the validity mask and the soft visibility weight:

$$\mathcal{L}_{depth}^{dense} = \frac{\sum_{\mathbf{p} \in \Omega^{dense}} M(\mathbf{p}) W_{vis}(\mathbf{p}) \Delta d_{rend}^{inv}(\mathbf{p})}{\sum_{\mathbf{p} \in \Omega^{dense}} M(\mathbf{p}) W_{vis}(\mathbf{p}) + \epsilon}. \quad (13)$$

## 4.2 Gradient Decoupling for Structural Protection

To mitigate the *Geometric Decay* identified in Sec. 3.4, we introduce **Gradient Decoupling** (GD). GD blocks the backpropagation of the photometric alignment loss to the Gaussian spatial parameters, preventing texture-driven photometric residuals from directly deforming the established metric structure.

**Implementation via Gradient Blocking.** In a standard joint optimization, the photometric alignment loss  $\mathcal{L}_{pho}$  backpropagates through the entire set of Gaussian parameters  $\mathcal{G}$ , comprising  $\{\mu, \Sigma, \alpha, c\}$ . To protect the structural integrity, we apply a stop-gradient operator  $\text{sg}[\cdot]$  to the spatial parameters  $(\mu, \Sigma)$  specifically within the color rendering process. The modified photometric objective is defined as:

$$\mathcal{L}_{pho}^{decoupled} = \mathcal{D}_{photo} \left( I^t, \Phi_{color}(\text{sg}[\mu, \Sigma], \alpha, c, \mathbf{T}_{cl} \cdot \mathbf{T}_{lw}^t) \right). \quad (14)$$

By explicitly decoupling these gradients, we ensure that the refinement of scene appearance and sensor extrinsics proceeds without inducing non-physical deformations in the underlying geometry.

**Selective Gradient Blocking.** As summarized in Fig. 2, we apply GD only to  $\mathcal{L}_{pho}$ , while allowing the reprojection loss  $\mathcal{L}_{rep}$  to update  $(\mu, \Sigma)$ . Although  $\mathcal{L}_{rep}$  also uses a photometric distance, it compares ground-truth image intensities through depth-induced reprojection correspondences without involving the learned Gaussian color attributes. As a result, its gradients to  $(\mu, \Sigma)$  are mediated by the reprojection geometry and primarily encourage view-consistent structure rather than per-view texture fitting. This is empirically supported by Fig. 1c. The depth loss sharply increases once the photometric alignment loss is activated, yet the subsequent activation of the reprojection loss does not induce a comparable degradation. This design leverages image-domain supervision while mitigating *Geometric Decay* and preserving LiDAR-anchored structure during extrinsic refinement.

Combining both components, GeoP-Calib modifies the baseline objective (Eq. 10) as:

$$\mathcal{L}_{total}^{GeoP} = \mathcal{L}_{pho}^{decoupled} + \lambda_{rep} \mathcal{L}_{rep} + \lambda_{depth}^{sparse} \mathcal{L}_{depth}^{sparse} + \lambda_{depth}^{dense} \mathcal{L}_{depth}^{dense}. \quad (15)$$

## 5 Experiment

### 5.1 Experiment Setup

**Dataset.** We evaluated GeoP-Calib on KITTI-360 [18] and KITTI odometry [5] (denoted as KITTI hereafter). For KITTI-360, we used the same five sequences

**Table 1:** Quantitative comparison of extrinsic calibration accuracy on KITTI and KITTI-360 datasets. Best results are in bold, and second-best results are underlined. Values in parentheses denote the standard deviation.

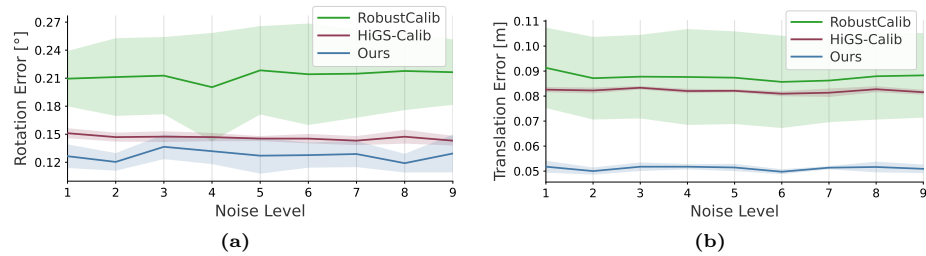
Seq.	GST [16]		Claim [35]		RobustCalib [37]		HiGS-Calib [33]		GeoP-Calib		
	$E_r(^{\circ})$	$E_t(\text{m})$	$E_r(^{\circ})$	$E_t(\text{m})$	$E_r(^{\circ})$	$E_t(\text{m})$	$E_r(^{\circ})$	$E_t(\text{m})$	$E_r(^{\circ})$	$E_t(\text{m})$	
KITTI-360	seq1	0.351 (0.080)	0.326 (0.018)	0.663 (0.409)	0.142 (0.148)	0.202 (0.021)	0.059 (0.006)	<b>0.111</b> (0.016)	<u>0.036</u> (0.006)	<u>0.124</u> (0.046)	<b>0.031</b> (0.006)
	seq2	4.631 (2.543)	0.394 (0.254)	0.614 (0.219)	<b>0.056</b> (0.030)	0.272 (0.046)	0.107 (0.019)	<u>0.177</u> (0.031)	0.087 (0.004)	<b>0.145</b> (0.020)	<u>0.059</u> (0.007)
	seq3	0.991 (0.199)	0.335 (0.212)	0.232 (0.073)	0.073 (0.041)	0.307 (0.047)	0.076 (0.002)	<u>0.103</u> (0.096)	0.177 (0.106)	<b>0.074</b> (0.030)	<b>0.059</b> (0.002)
	seq4	1.011 (0.176)	0.316 (0.210)	0.809 (0.641)	0.296 (0.253)	0.246 (0.068)	0.133 (0.005)	<u>0.167</u> (0.161)	<u>0.118</u> (0.111)	<b>0.110</b> (0.042)	<b>0.071</b> (0.001)
	seq5	0.751 (0.423)	0.206 (0.071)	0.284 (0.110)	<b>0.079</b> (0.039)	0.248 (0.003)	0.109 (0.012)	<b>0.141</b> (0.010)	0.098 (0.001)	<u>0.153</u> (0.030)	<u>0.093</u> (0.003)
	Avg.	1.547 (1.570)	0.315 (0.099)	0.520 (0.303)	0.129 (0.126)	0.255 (0.055)	<u>0.097</u> (0.028)	<u>0.140</u> (0.059)	0.103 (0.055)	<b>0.121</b> (0.041)	<b>0.063</b> (0.021)
KITTI	seq1	2.626 (1.151)	0.251 (0.082)	0.247 (0.125)	0.044 (0.016)	0.305 (0.051)	0.050 (0.006)	<u>0.112</u> (0.008)	<u>0.042</u> (0.002)	<b>0.105</b> (0.009)	<b>0.034</b> (0.001)
	seq2	1.310 (0.294)	0.232 (0.058)	0.371 (0.107)	0.092 (0.030)	0.482 (0.417)	0.074 (0.014)	<u>0.201</u> (0.003)	<u>0.056</u> (0.001)	<b>0.160</b> (0.010)	<b>0.054</b> (0.001)
	seq3	1.106 (0.678)	0.167 (0.050)	<b>0.219</b> (0.066)	<u>0.052</u> (0.017)	0.300 (0.018)	0.083 (0.008)	0.239 (0.016)	<u>0.052</u> (0.005)	<u>0.224</u> (0.018)	<b>0.023</b> (0.001)
	seq4	0.512 (0.186)	0.413 (0.159)	0.354 (0.078)	0.067 (0.018)	0.298 (0.102)	0.084 (0.009)	<u>0.269</u> (0.011)	<u>0.062</u> (0.001)	<b>0.250</b> (0.024)	<b>0.052</b> (0.001)
	seq5	2.801 (1.475)	0.327 (0.178)	0.411 (0.056)	<b>0.051</b> (0.013)	0.375 (0.150)	0.119 (0.012)	<u>0.209</u> (0.007)	0.065 (0.003)	<b>0.201</b> (0.016)	<u>0.056</u> (0.002)
	Avg.	1.671 (0.892)	0.278 (0.085)	0.352 (0.074)	0.082 (0.022)	0.352 (0.071)	0.082 (0.022)	<u>0.206</u> (0.053)	<u>0.055</u> (0.008)	<b>0.188</b> (0.051)	<b>0.044</b> (0.013)

as in [7, 33]. For KITTI, we used subsets of sequences 5, 6, 7, 9, and 10 following [37]. On KITTI-360, calibration was run on the two forward-facing cameras and averaged. On KITTI, we used only cam2.

**Baselines and Protocols.** We compared against GST [16] as a representative traditional optimization-based method, CLAIM [35] as a foundation-model-based method, and RobustCalib [37] and HiGS-Calib [33] as recent 3DGS-based methods. All methods shared a unified initialization where the camera pose was aligned with the forward-facing LiDAR frame, except GST, whose SuperGlue [24] based correspondence initialization could fail under sparse LiDAR projections and lead to invalid or unstable optimization. Therefore, we initialized GST by perturbing the ground-truth extrinsics with a  $1^{\circ}$  rotation and 0.2m translation to ensure that it entered a valid optimization regime. For reliable evaluation, all results are reported as the mean and standard deviation over five random seeds; further details are provided in the supplementary material.

**Table 2:** Component analysis for GeoP-Calib on KITTI-360. Note that VSM is a sub-component of DDA and is not defined without it.

GD	DDA	VSM	$E_r(^{\circ})$	$E_t(\text{m})$	$T(\text{s})$
			0.122 (0.031)	0.077 (0.022)	875 (16)
	✓	✓	0.117 (0.044)	0.068 (0.021)	923 (16)
✓			0.127 (0.042)	0.067 (0.022)	866 (21)
✓	✓		0.124 (0.045)	0.066 (0.024)	—
✓	✓	✓	0.121 (0.041)	0.063 (0.021)	922 (21)

**Fig. 3:** Calibration error under varying noise levels: (a) rotation and (b) translation.

**Evaluation Metrics.** We quantitatively evaluated calibration accuracy using the translation error  $E_t$  and the geodesic rotation error  $E_r$ :

$$E_t = \|\hat{\mathbf{t}} - \mathbf{t}^*\|_2, \quad E_r = \arccos\left(\frac{\text{Tr}\left((\mathbf{R}^*)^\top \hat{\mathbf{R}}\right) - 1}{2}\right), \quad (16)$$

where  $(\mathbf{R}^*, \mathbf{t}^*)$  and  $(\hat{\mathbf{R}}, \hat{\mathbf{t}})$  denote the ground-truth and estimated extrinsics, respectively. We report  $E_r$  in degrees by converting radians to degrees.

## 5.2 Implementation Details

We built our method on top of the HiGS code base [33]. Notably, GD is applied throughout all stages to prevent photometric residuals from distorting the scene geometry. We adopt a two-stage optimization schedule: a geometric warm-up stage using sparse depth supervision to establish stable coarse geometry, followed by a refinement stage that activates the dense depth anchoring loss  $\mathcal{L}_{depth}^{dense}$ . For detailed hyperparameter settings, we refer the reader to the supplementary material.

## 5.3 Calibration Performance on Driving Datasets

We evaluated rotation and translation errors on KITTI-360 and KITTI. As shown in Table 1, GeoP-Calib achieved the lowest rotation error on 7 out of 10 sequences and the lowest translation error on 7 out of 10 sequences across

KITTI-360 and KITTI. GST did not perform reliably in these driving scenarios, showing large errors and high variance. CLAIM, which leverages foundation models, attained relatively accurate translation in several sequences. However, its rotation accuracy was consistently inferior to the 3DGS-based approaches, limiting its overall calibration quality. Compared to prior 3DGS-based methods, GeoP-Calib improved translation over HiGS-Calib on every sequence, demonstrating that preserving the geometric integrity of the 3DGS proxy provides a stronger signal for metric alignment. Specifically, GeoP-Calib improves over the second-best result by  $0.019^\circ$  in rotation and 0.034 m in translation on KITTI-360, and by  $0.018^\circ$  in rotation and 0.011 m in translation on KITTI.

#### 5.4 Ablation Study

Table 2 reports the component-wise ablation of GeoP-Calib on KITTI-360. Starting from Base, adding DDA reduced the translation error from 0.077 m to 0.068 m and slightly improved the rotation error from  $0.122^\circ$  to  $0.117^\circ$ . Adding GD alone also reduced the translation error to 0.067 m, while the rotation error slightly increased to  $0.127^\circ$ . When both DDA and GD were enabled, GeoP-Calib achieved the best overall performance, reducing translation to 0.063 m and rotation to  $0.121^\circ$ . Compared to Base, this corresponds to a 0.014 m improvement in translation while maintaining comparable rotation accuracy.

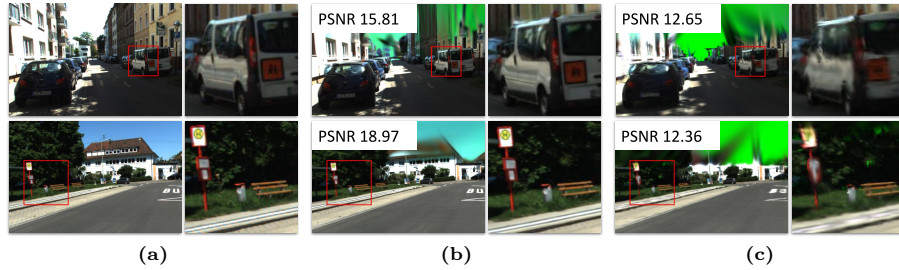
The limited rotation gain from GD alone stems from the differing geometric sensitivity of the two error types: translation induces a pixel offset inversely proportional to depth, whereas rotation produces a comparable angular offset regardless of distance. Since GD suppresses texture-driven photometric residuals that would pull Gaussians into LiDAR-absent regions, it limits far-range texture cues that could otherwise refine rotation.

We additionally ablated the VSM used for dense depth anchoring. With the same DDA+GD setting, enabling VSM improved rotation from  $0.124^\circ$  to  $0.121^\circ$  and translation from 0.066 m to 0.063 m. These results indicate that applying VSM for occlusion handling during dense depth anchoring provides a consistent benefit.

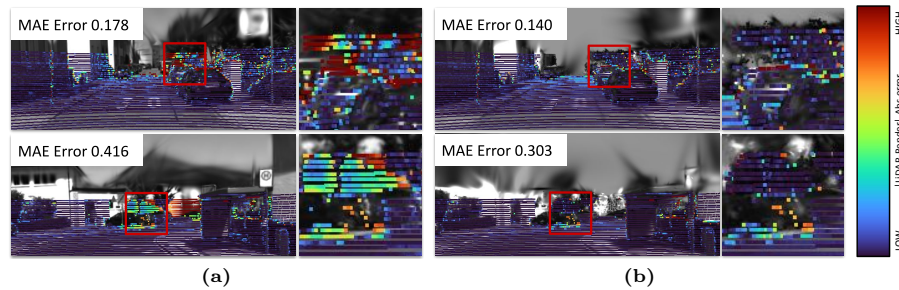
In addition, on our RTX 4070 Ti and Ryzen 3600 setup, adding DDA increased the runtime by about 50s compared to Base, resulting in an average per-sequence runtime of approximately 15.4 minutes.

#### 5.5 Extended Analysis on Calibration Properties

**Robustness to Pose Initialization.** We evaluated robustness to initialization on KITTI-360 Seq. 2 by injecting noise into the initial extrinsics. Specifically, we defined nine noise levels, where the rotation noise increased from  $1^\circ$  to  $9^\circ$ , and the translation noise increased from 0.1 m to 0.9 m. For each noise level, we ran calibration from the perturbed initialization and reported the resulting rotation and translation errors. Fig. 3 shows that GeoP-Calib converged robustly even under high initialization noise levels.



**Fig. 4:** Qualitative comparison of rendered RGB images on KITTI-360: (a) GT, (b) Base and (c) GeoP-Calib (Ours). PSNR is computed over the entire image to assess photometric consistency. The Gaussian proxy is discarded after the calibration process, and only the estimated extrinsics are retained.



**Fig. 5:** Rendered depth error maps on KITTI-360, measured as the Mean Absolute Error (MAE) between rendered depth and LiDAR depth in the LiDAR-translation and camera-rotation setting: (a) Base and (b) GeoP-Calib (Ours). Colors indicate error magnitude, where blue denotes lower error and red denotes higher error.

**Rendering Quality Does Not Imply Calibration Quality.** We conducted this analysis on KITTI-360 Seq. 2 and Seq. 4. As shown in Fig. 4, GeoP-Calib yielded lower PSNR than the Base pipeline, indicating reduced photometric consistency with the ground-truth RGB. Nevertheless, GeoP-Calib achieved higher LiDAR-Camera calibration accuracy on the same sequences. These results demonstrate that photometric fidelity is not a sufficient condition for accurate calibration. Rather, since extrinsic refinement is mediated through the Gaussian proxy, geometric faithfulness between the LiDAR structure and the 3DGS representation should be established as a prerequisite before improvements in photometric reconstruction can meaningfully translate into calibration gains.

**Analysis for Geometric Accuracy.** We evaluated how well GeoP-Calib preserves metric geometry compared to Base using the depth MAE between rendered depth and LiDAR depth on KITTI-360 after convergence. GeoP-Calib reduced the average MAE from 0.294 to 0.244, an absolute reduction of 0.05 or approximately 17%. As shown in Fig. 5, Base exhibited localized regions where the depth error sharply increased, which is consistent with *Geometric De-*

*cay* that distorts the proxy geometry under photometric optimization, whereas GeoP-Calib yields a more spatially consistent error distribution.

## 6 Conclusion

We present GeoP-Calib, a geometry-preserving framework for targetless LiDAR-Camera calibration with 3DGS. By combining Dense Depth Anchoring and Gradient Decoupling, GeoP-Calib mitigates texture-driven *Geometric Decay* while preserving LiDAR-consistent proxy geometry. Experiments on KITTI and KITTI-360 showed consistent calibration improvements, particularly in translation.

**Limitations and Future Works.** While translation accuracy improved notably, rotation gains remained modest. GeoP-Calib relies on LiDAR odometry to construct the Gaussian proxy, so degraded LiDAR registration quality may affect proxy consistency. Gradient Decoupling effectively defends against *Geometric Decay*, but may limit the use of photometric texture cues. Future work will explore how to better exploit texture cues while preserving metric faithfulness, and whether the diagnosis of *Geometric Decay* generalizes to other differentiable scene proxies for calibration.

## Acknowledgments

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) under Grant No. RS-2025-00564137, in part by Convergence security core talent training business support program under Grant IITP-2023-RS-2023-00266615, and in part by the Technology Innovation Program (RS-2025-25453780, Development of a National Humanoid AI Robot Foundation Model for Multi-Task Applications) funded by the Ministry of Trade Industry & Resources (MOTIR, Korea).

## References

1. Beltrán, J., Guindel, C., De La Escalera, A., García, F.: Automatic extrinsic calibration method for lidar and camera sensor setups. *IEEE Transactions on Intelligent Transportation Systems* **23**(10), 17677–17689 (2022)
2. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 11621–11631 (2020)
3. Civera, J., Davison, A.J., Montiel, J.M.: Inverse depth parametrization for monocular slam. *IEEE transactions on robotics* **24**(5), 932–945 (2008)
4. Fan, Z., Zhang, L., Wang, X., Shen, Y., Deng, F.: Lidar, imu, and camera fusion for simultaneous localization and mapping: A systematic review. *Artificial Intelligence Review* **58**(6), 174 (2025)
5. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2012)

6. Guan, H., Song, C., Zhang, Z.: Lidar-camera cooperative semantic segmentation. *Machine Intelligence Research* **22**(5), 956–968 (2025)
7. Herau, Q., Bennehar, M., Moreau, A., Piasco, N., Roldão, L., Tsishkou, D., Migniot, C., Vasseur, P., Demonceaux, C.: 3dgs-calib: 3d gaussian splatting for multimodal spatiotemporal calibration. In: *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 8315–8321. IEEE (2024)
8. Herau, Q., Piasco, N., Bennehar, M., Roldao, L., Tsishkou, D., Migniot, C., Vasseur, P., Demonceaux, C.: Soac: Spatio-temporal overlap-aware multi-sensor calibration using neural radiance fields. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 15131–15140 (2024)
9. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2d gaussian splatting for geometrically accurate radiance fields. In: *ACM SIGGRAPH 2024 conference papers*. pp. 1–11 (2024)
10. Irie, K., Sugiyama, M., Tomono, M.: Target-less camera-lidar extrinsic calibration using a bagged dependence estimator. In: *2016 IEEE International Conference on Automation Science and Engineering (CASE)*. pp. 1340–1347. IEEE (2016)
11. Iyer, G., Murthy, J.K., Krishna, K.M., et al.: Calibnet: self-supervised extrinsic calibration using 3d spatial transformer networks. *arXiv preprint arXiv:1803.08181* (2018)
12. Jung, H., Kim, N., Kim, J., Park, J.: Targetless lidar-camera calibration with neural gaussian splatting (2026)
13. Kang, J., Doh, N.L.: Automatic targetless camera–lidar calibration by aligning edge with gaussian mixture model. *Journal of Field Robotics* **37**(1), 158–179 (2020)
14. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., et al.: 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.* **42**(4), 139–1 (2023)
15. Kim, D., Shin, S., Hwang, H.: Camera-lidar extrinsic calibration using constrained optimization with circle placement. *IEEE Robotics and Automation Letters* **10**(2), 883–890 (2024)
16. Koide, K., Oishi, S., Yokozuka, M., Banno, A.: General, single-shot, target-less, and automatic lidar-camera extrinsic calibration toolbox. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. pp. 11301–11307. IEEE (2023)
17. Levinson, J., Thrun, S.: Automatic online calibration of cameras and lasers. In: *Robotics: science and systems*. vol. 2. Berlin, Germany (2013)
18. Liao, Y., Xie, J., Geiger, A.: KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *arXiv preprint arXiv:2109.13410* (2021)
19. Lv, X., Wang, B., Dou, Z., Ye, D., Wang, S.: Lccnet: Lidar and camera self-calibration using cost volume network. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 2894–2901 (2021)
20. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* **65**(1), 99–106 (2021)
21. Ou, N., Cai, H., Wang, J.: Targetless lidar-camera calibration via cross-modality structure consistency. *IEEE Transactions on Intelligent Vehicles* **9**(1), 2636–2648 (2023)
22. Pandey, G., McBride, J., Savarese, S., Eustice, R.: Automatic targetless extrinsic calibration of a 3d lidar and camera by maximizing mutual information. In: *Proceedings of the AAAI conference on artificial intelligence*. vol. 26, pp. 2053–2059 (2012)
23. Qiu, J., Cui, Z., Zhang, Y., Zhang, X., Liu, S., Zeng, B., Pollefeys, M.: Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse

- lidar data and single color image. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3313–3322 (2019)
24. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperGlue: Learning feature matching with graph neural networks. In: CVPR (2020)
  25. Song, Z., Yang, L., Xu, S., Liu, L., Xu, D., Jia, C., Jia, F., Wang, L.: Graphbev: Towards robust bev feature alignment for multi-modal 3d object detection. In: European Conference on Computer Vision (ECCV). pp. 347–366. Springer (2024)
  26. Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., et al.: Stanley: The robot that won the darpa grand challenge. *Journal of field Robotics* **23**(9), 661–692 (2006)
  27. Vora, S., Lang, A.H., Helou, B., Beijbom, O.: Pointpainting: Sequential fusion for 3d object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4604–4612 (2020)
  28. Wang, Y., Mao, Q., Zhu, H., Deng, J., Zhang, Y., Ji, J., Li, H., Zhang, Y.: Multi-modal 3d object detection in autonomous driving: a survey. *International Journal of Computer Vision* **131**(8), 2122–2152 (2023)
  29. Yan, G., Liu, Z., Wang, C., Shi, C., Wei, P., Cai, X., Ma, T., Liu, Z., Zhong, Z., Liu, Y., et al.: OpenCalib: A multi-sensor calibration toolbox for autonomous driving. *Software Impacts* **14**, 100393 (2022)
  30. Yang, Z., Chen, G., Zhang, H., Ta, K., Bărsan, I.A., Murphy, D., Manivasagam, S., Urtasun, R.: Unical: Unified neural sensor calibration. In: European Conference on Computer Vision (ECCV). pp. 327–345. Springer (2024)
  31. Yuan, C., Liu, X., Hong, X., Zhang, F.: Pixel-level extrinsic self calibration of high resolution lidar and camera in targetless environments. *IEEE Robotics and Automation Letters* **6**(4), 7517–7524 (2021)
  32. Zhang, Q., Pless, R.: Extrinsic calibration of a camera and laser range finder (improves camera calibration). In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566). vol. 3, pp. 2301–2306. IEEE (2004)
  33. Zhang, T., Zhang, L., Wang, H.: Higs-calib: A hierarchical 3d gaussian splatting based targetless local-consistent lidar-camera calibration method. *IEEE Transactions on Circuits and Systems for Video Technology* (2025)
  34. Zhang, X., Zhu, S., Guo, S., Li, J., Liu, H.: Line-based automatic extrinsic calibration of lidar and camera. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 9347–9353. IEEE (2021)
  35. Zhang, Z., Liu, Y., Zhang, M., Tan, F., Ding, Y.: Claim: Camera-lidar alignment with intensity and monodepth. In: 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 17921–17926. IEEE (2025)
  36. Zhou, L., Li, Z., Kaess, M.: Automatic extrinsic calibration of a camera and a 3d lidar using line and plane correspondences. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5562–5569. IEEE (2018)
  37. Zhou, S., Xie, S., Ishikawa, R., Oishi, T.: Robust lidar-camera calibration with 2d gaussian splatting. *IEEE Robotics and Automation Letters* (2025)
  38. Zhou, S., Xie, S., Ishikawa, R., Sakurada, K., Onishi, M., Oishi, T.: Inf: Implicit neural fusion for lidar and camera. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 10918–10925. IEEE (2023)
  39. Zhu, Y., Li, C., Zhang, Y.: Online camera-lidar calibration with sensor semantic information. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 4970–4976. IEEE (2020)