MDPI

Article

Dual-Camera LiDAR Fusion for Occlusion-Robust 3D Detection in Urban Driving Simulation

Xingnan Zhou [1] and Ciprian Alecsandru [1,*]

[1] Department of Building, Civil and Environmental Engineering, Concordia University, Montreal, QC H3G 1M8, Canada

* Correspondence: ciprian.alecsandru@concordia.ca (C.A.)

Version February 10, 2026 submitted to Sustainability

Abstract: Three-dimensional object detection from LiDAR point clouds is a cornerstone of autonomous driving perception, yet single-sensor systems remain vulnerable to occlusion in complex urban environments such as roundabouts and dense intersections. This paper proposes an asymmetric dual-camera LiDAR fusion framework that combines a PointPillar-based 3D LiDAR detector with YOLOv8-based 2D detections from two complementary camera viewpoints: a drone (top-down, 40 m altitude) and a subject-vehicle forward camera. The fusion operates at the decision level (late fusion), where camera-confirmed LiDAR detections receive confidence boosts—$\times 1.15$ for single-camera confirmation and $\times 1.30$ for dual-camera agreement—while unconfirmed low-confidence detections within the drone's field of view are suppressed ($\times 0.75$). Crucially, the design is asymmetric: the drone camera applies both boost and suppress operations, whereas the limited-field-of-view forward camera applies boost-only to avoid penalizing legitimate detections outside its narrow coverage. Evaluated on a CARLA Town10HD dataset comprising 650 frames and 12,308 annotations across Car and Pedestrian classes, with five-seed repeated random sub-sampling validation (all at epoch 80), the drone-fused system improves mAP@0.5 by +1.48 percentage points (+5.8% relative; sign test $p = 0.031$) and the full dual-camera fusion achieves +2.88 pp (+11.2% relative; sign test $p = 0.031$), with all five seeds showing positive improvement in every configuration. Critically, domain-adapted YOLOv8 models fine-tuned on CARLA-rendered images are essential—the forward camera contributes a meaningful +0.62 pp mAP improvement (sign test $p = 0.031$) only after domain adaptation. While baseline performance is constrained by the small training set (520 frames per seed), the consistent directional improvements across all seeds and configurations demonstrate the potential of asymmetric multi-viewpoint fusion for occlusion-robust perception.

Keywords: 3D object detection; LiDAR-camera fusion; intelligent transport system; late fusion; drone-assisted perception; sustainable traffic management; PointPillars; YOLOv8; CARLA simulation; autonomous driving safety

1. Introduction

Reliable three-dimensional (3D) object detection is a fundamental requirement for safe autonomous driving. LiDAR-based detectors have become the dominant paradigm for 3D perception in self-driving systems, offering precise depth measurements and geometric representations of the driving environment [1,2]. However, single-viewpoint LiDAR systems suffer from well-documented limitations: occlusion by foreground objects, sparse point density at long range, and blind spots caused by the sensor's mounting position [3]. These limitations are particularly acute in complex urban geometries such as roundabouts, dense intersections, and narrow streets, where vehicles, pedestrians, and infrastructure elements frequently occlude one another from the ego vehicle's perspective.

Camera-LiDAR fusion has emerged as a promising strategy to mitigate single-sensor limitations by combining the geometric precision of LiDAR with the rich semantic and texture information provided by cameras [4,5]. Early fusion methods such as PointPainting [6] and Frustum PointNets [7] augment point clouds with camera features at the data level, while deep fusion approaches like BEVFusion [8,9] and TransFusion [10] learn joint representations in a unified bird's-eye view (BEV) space. Although these methods have achieved impressive results on benchmarks such as nuScenes [11] and KITTI [12], they predominantly rely on cameras co-located with the ego vehicle, inheriting the same viewpoint limitations that constrain the LiDAR sensor.

A fundamentally different approach to overcoming single-viewpoint occlusion is to introduce cameras at *complementary* vantage points. Drone-assisted perception [13,14] and vehicle-to-infrastructure (V2I) cooperative systems [15,16] have demonstrated that elevated or offset viewpoints can observe objects hidden from the street-level perspective. However, most cooperative perception research has focused on sharing intermediate features between multiple LiDAR-equipped agents [17–19], which requires expensive sensor suites on all cooperating platforms. In contrast, cameras are lightweight, inexpensive, and easily deployable on drones or infrastructure poles, making camera-based viewpoint augmentation a practical and cost-effective alternative to full multi-LiDAR cooperative perception.

This paper proposes an asymmetric dual-camera LiDAR fusion framework that addresses occlusion in urban driving by combining a vehicle-mounted LiDAR 3D detector with 2D object detections from two complementary cameras: (1) a drone camera providing a top-down perspective at 40 m altitude, and (2) the subject vehicle's forward-facing camera. The key insight motivating the asymmetric design is that these two cameras have fundamentally different coverage characteristics. The drone camera provides a wide, overhead field of view (FOV) that can observe nearly the entire scene, including objects fully occluded from street level. The forward camera, by contrast, covers only a narrow frontal sector but provides high-resolution appearance information for objects directly ahead of the vehicle. Treating these cameras identically in the fusion pipeline would be suboptimal: the drone's comprehensive coverage justifies both boosting confirmed detections *and* suppressing unconfirmed ones, while the forward camera's limited FOV means that the absence of a camera detection does not reliably indicate the absence of an object.

The fusion operates at the decision level (late fusion), modifying the confidence scores of the 3D LiDAR detector's output based on spatial agreement with 2D camera detections projected into the BEV plane. Camera-confirmed LiDAR detections receive a confidence boost of $\times 1.15$ for single-camera confirmation or $\times 1.30$ for dual-camera agreement. Unconfirmed low-confidence Car detections within the drone's FOV are suppressed by a factor of $\times 0.75$, while no suppression is applied for detections outside the drone's coverage or within the forward camera's limited FOV alone. This late-fusion approach has several practical advantages: it is modular (any 3D detector and any 2D detector can be used), computationally lightweight (no retraining required), and transparent (the confidence adjustments are interpretable).

From a sustainable transportation perspective, improved 3D perception directly supports safer autonomous driving—a critical enabler of sustainable urban mobility. Traffic accidents remain a leading cause of preventable death [20], and occlusion-related collisions at intersections and roundabouts disproportionately affect vulnerable road users such as pedestrians and cyclists. By leveraging cost-effective drone cameras rather than expensive multi-LiDAR infrastructure, the proposed approach offers a scalable pathway to enhanced perception that aligns with sustainable smart-city transportation strategies [21].

We evaluate the proposed framework using the CARLA driving simulator [22], which provides precise ground-truth annotations and full control over sensor placement. The dataset comprises 650 frames collected in Town10HD with 12,308 object annotations spanning Car and Pedestrian classes. To ensure statistical rigor, we conduct five-seed repeated random sub-sampling validation with all

models trained to epoch 80, and report both parametric (paired t-test) and non-parametric (sign test) significance measures. Our contributions are as follows:

1. We propose an *asymmetric* late-fusion framework that differentiates between cameras based on their coverage characteristics, applying boost-and-suppress operations for the wide-FOV drone camera and boost-only operations for the narrow-FOV forward camera.
2. We demonstrate that drone-camera fusion improves mAP@0.5 by +1.48 percentage points (+5.8% relative) over the LiDAR-only baseline (sign test $p = 0.031$), with the full dual-camera system achieving +2.88 pp (+11.2% relative). All three fusion configurations achieve 5/5 positive seeds ($p = 0.031$), with domain-adapted YOLOv8 detectors proving essential for effective fusion.
3. We conduct five-seed repeated random sub-sampling validation with both parametric (t-test) and non-parametric (sign test) significance measures, providing transparent reporting of statistical power and its limitations in small-sample detection experiments.
4. We provide the complete data-collection and fusion pipeline built on CARLA and OpenPCDet [23], with code to be released upon publication, enabling reproducible research on multi-viewpoint fusion for autonomous driving perception.

The remainder of this paper is organized as follows. Section 2 reviews related work on LiDAR-based 3D detection, camera-LiDAR fusion, and drone-assisted perception. Section 3 describes the proposed asymmetric fusion methodology. Section 4 details the experimental setup, including the CARLA data-collection pipeline and training configuration. Section 5 presents quantitative results with statistical analysis. Section 6 discusses findings, limitations, and practical implications. Section 7 concludes the paper.

2. Related Work

2.1. LiDAR-Based 3D Object Detection

LiDAR-based 3D object detection has progressed through several architectural paradigms. Point-based methods such as PointNet [24] and PointNet++ [25] operate directly on raw point clouds, learning per-point features through shared multi-layer perceptrons and set abstraction layers. PointRCNN [26] extended this paradigm to two-stage 3D detection by generating proposals from point-level features. While point-based methods preserve fine geometric detail, their computational cost scales unfavorably with point cloud density.

Voxel-based methods discretize the point cloud into a regular 3D grid and apply sparse 3D convolutions. VoxelNet [27] pioneered end-to-end voxel-based detection, and SECOND [28] introduced spatially sparse convolutions that dramatically reduced computation by only processing occupied voxels. CenterPoint [29] further advanced voxel-based detection with center-heatmap prediction and a two-stage refinement module. These methods achieve strong accuracy but require careful voxel resolution tuning to balance precision and efficiency.

Pillar-based methods, led by PointPillars [30], offer a compelling trade-off between speed and accuracy by collapsing the vertical dimension into a single "pillar" per horizontal grid cell. The resulting 2D pseudo-image can be processed by standard 2D convolutional backbones and detection heads, enabling real-time inference. PointPillars remains a widely used baseline in both academic benchmarks and practical deployments due to its simplicity, speed, and competitive accuracy. Wang et al. [31] further explored pillar-based architectures with improved feature encoding. Hybrid approaches such as PV-RCNN [32] combine voxel and point-based processing for state-of-the-art accuracy at the cost of increased complexity.

In this work, we adopt PointPillars as the LiDAR 3D detector due to its favorable speed–accuracy trade-off and its suitability as a baseline for evaluating fusion strategies. The modular nature of our late-fusion approach means that the 3D detector can be replaced with any alternative without modifying the fusion logic.

2.2. Camera-LiDAR Fusion for 3D Detection

Camera-LiDAR fusion methods can be broadly categorized by the stage at which sensor information is combined: early (data-level), deep (feature-level), and late (decision-level) fusion [4,5].

2.2.1. Early Fusion

Early fusion methods augment the LiDAR point cloud with camera-derived features before detection. PointPainting [6] projects LiDAR points onto the camera image and appends per-point semantic segmentation scores to the point features, enabling the 3D detector to leverage appearance information. MV3D [33] generates multi-view representations (BEV, front view, and camera image) and fuses them through region-based networks. While conceptually straightforward, early fusion methods are sensitive to calibration accuracy and cannot leverage camera information for regions not covered by the LiDAR.

2.2.2. Deep Fusion

Deep fusion methods learn joint representations from both modalities. BEVFusion [8,9] lifts camera features into 3D space using depth estimation (e.g., the Lift-Splat-Shoot paradigm [34]) and fuses them with LiDAR BEV features through concatenation or attention mechanisms. TransFusion [10] uses transformer-based cross-attention to fuse LiDAR and camera features at the object query level. DeepFusion [35] introduces cross-modal alignment through learned geometric transformations. These methods achieve state-of-the-art accuracy on benchmarks like nuScenes [11] but require end-to-end retraining and are computationally expensive.

2.2.3. Late Fusion

Late fusion methods combine the outputs (detections) of independent modality-specific detectors at the decision level. CLOCs [36] learns a fusion network that combines 2D camera detections with 3D LiDAR detections based on geometric consistency, improving recall without modifying the base detectors. AVOD [37] jointly generates 3D proposals from both modalities and fuses them at the region-of-interest level. Nobis et al. [38] demonstrated late fusion of radar and camera detectors using learned confidence recalibration. Late fusion has practical advantages: the individual detectors can be trained independently, the fusion module is lightweight, and the approach is inherently modular. Our work follows the late-fusion paradigm but introduces asymmetric confidence adjustment that accounts for the differing coverage characteristics of the two cameras.

2.3. Drone-Assisted and Cooperative Perception

The use of elevated viewpoints to overcome occlusion has gained increasing attention. Vehicle-to-everything (V2X) cooperative perception frameworks such as OPV2V [17], V2X-ViT [18], and CoBEVT [19] enable multiple LiDAR-equipped agents to share intermediate features for improved detection. DAIR-V2X [16] provides a benchmark for vehicle-infrastructure cooperative 3D detection. These approaches typically assume that all cooperating agents are equipped with LiDAR sensors and high-bandwidth communication links.

Drone-assisted perception offers a complementary paradigm where an unmanned aerial vehicle (UAV) provides an elevated camera viewpoint to augment the ego vehicle's perception [13,14]. The overhead perspective of a drone is particularly effective for resolving occlusions in dense traffic, as objects hidden behind foreground vehicles are often fully visible from above. However, drones typically carry only cameras (not LiDAR) due to payload constraints, necessitating a heterogeneous fusion approach that combines 3D LiDAR detections with 2D camera detections from the drone.

Our work is distinguished from prior cooperative perception research in two key aspects. First, we fuse *heterogeneous* sensor modalities (3D LiDAR from the ego vehicle + 2D cameras from two viewpoints), rather than sharing homogeneous LiDAR features. Second, our asymmetric fusion design

explicitly accounts for the different FOV characteristics of the drone and forward cameras, applying differentiated confidence adjustments rather than treating all camera sources uniformly.

3. Methodology

This section describes the overall system architecture, the individual detection components, and the asymmetric fusion algorithm.

3.1. System Overview

The proposed pipeline comprises three stages (Figure 1):

1. **3D LiDAR Detection.** A PointPillar network processes the ego vehicle's LiDAR point cloud to produce 3D bounding boxes with class labels and confidence scores in BEV coordinates.
2. **2D Camera Detection.** YOLOv8 independently processes images from the drone camera and the forward camera, producing 2D bounding boxes with class labels and confidence scores in each camera's image plane.
3. **Asymmetric Late Fusion.** The 3D LiDAR detections are refined by spatially matching them against the 2D camera detections (projected into BEV or world coordinates) and applying asymmetric confidence adjustments based on agreement, camera identity, and detection confidence.

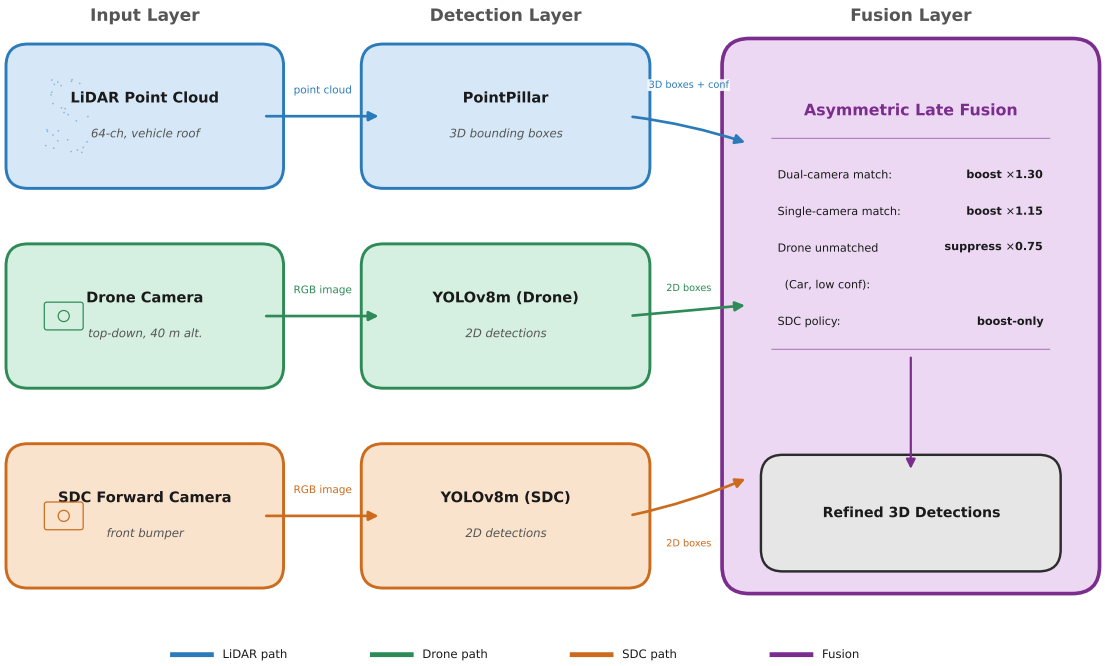

Figure 1. Overview of the proposed dual-camera LiDAR fusion pipeline. The PointPillar 3D detector and two independent YOLOv8 2D detectors produce detections from their respective sensors. The asymmetric late-fusion module refines the 3D detection confidence scores based on spatial agreement with camera detections, applying camera-specific boost and suppress operations.

3.2. 3D LiDAR Detection: PointPillars

We employ the PointPillar architecture [30] for LiDAR-based 3D object detection. PointPillars discretizes the x-y plane of the point cloud into a grid of vertical pillars and encodes the points within each pillar using a simplified PointNet [24]. The encoded pillar features are scattered back to a 2D pseudo-image and processed by a 2D convolutional backbone with a feature pyramid network

(FPN) [39] to produce multi-scale feature maps. A single-shot detection (SSD) head generates oriented 3D bounding boxes $(x, y, z, l, w, h, \theta)$ with associated class probabilities and confidence scores.

In our configuration, the point cloud is discretized with a voxel (pillar) size of 0.16×0.16 m in the horizontal plane, covering a detection range of $[-70.4, 70.4]$ m in both x and y directions and $[-3.0, 10.0]$ m in z. The backbone uses three downsampling blocks with strides of $[1, 2, 4]$ and an upsampling neck with strides $[1, 2, 4]$ to produce a multi-scale BEV feature map. The detection head predicts boxes for two classes: `Car` and `Pedestrian`, using class-specific anchor boxes. We train the model using the OpenPCDet framework [23] with CARLA ground-truth labels.

3.3. 2D Camera Detection: YOLOv8

For 2D object detection from camera images, we employ YOLOv8 [40], a state-of-the-art real-time detector that builds on the YOLO family [41] with an anchor-free detection head, decoupled classification and regression branches, and a CSPDarknet backbone with path aggregation. YOLOv8 achieves an excellent balance between speed and accuracy on standard 2D benchmarks such as COCO [42].

Two independent YOLOv8 models are trained on CARLA-rendered images from the two camera viewpoints:

- **Drone camera.** Mounted at 40 m altitude directly above the ego vehicle, pointing downward (-90° pitch). This camera provides a near-orthographic top-down view of the scene, enabling detection of vehicles and pedestrians regardless of inter-object occlusion at street level. The image resolution is 1920×1280 pixels with a 110° FOV.
- **SDC forward camera.** Mounted on the front bumper of the subject driving car (SDC) at standard height ($\sim$1.6 m), facing forward. This camera provides high-resolution frontal coverage typical of production autonomous vehicles. The image resolution is 1920×1280 pixels with a 110° FOV.

Both YOLOv8 models are trained to detect the same two classes (`Car` and `Pedestrian`) as the LiDAR detector, using 2D bounding-box annotations generated by projecting CARLA ground-truth 3D boxes into each camera's image plane. Note that the YOLOv8 models are trained once on the full set of available camera images and shared across all five PointPillar seeds, as the camera detectors serve as fixed auxiliary inputs to the fusion pipeline.

3.4. Asymmetric Late Fusion

The core contribution of this paper is the asymmetric late-fusion algorithm that combines 3D LiDAR detections with 2D camera detections while respecting the different coverage characteristics of the drone and forward cameras. The fusion operates by adjusting the confidence score s of each LiDAR detection based on its spatial agreement with camera detections.

3.4.1. Spatial Matching

For each LiDAR 3D detection d_L, we project its 3D bounding box into each camera's image plane using the known LiDAR-to-camera extrinsic and intrinsic matrices (available from CARLA's ground-truth sensor calibration), producing a 2D bounding box in image coordinates. A camera detection d_C is considered a *match* to the projected LiDAR detection if:

1. The class labels agree (both `Car` or both `Pedestrian`).
2. The 2D intersection-over-union (IoU) between the camera detection box and the projected LiDAR box in image space exceeds a threshold $\tau_{\text{IoU}} = 0.3$.

The relatively low IoU threshold of 0.3 accounts for the geometric mismatch between YOLO's 2D bounding boxes (tight image-space rectangles) and the projected 3D LiDAR boxes (which may include empty space due to the box-to-image projection). When multiple matches exist, optimal assignment is computed using the Hungarian algorithm [43] to maximize total IoU.

3.4.2. FOV Determination

A critical step is determining whether a LiDAR detection falls within each camera's FOV. For the drone camera, which provides near-complete overhead coverage, we define the FOV as a circle of radius $R_{\text{drone}} = 50\,\text{m}$ centered on the ego vehicle. For the forward camera, we define the FOV as a sector of angle $\alpha_{\text{fwd}} = 110^\circ$ and range $R_{\text{fwd}} = 50\,\text{m}$, aligned with the ego vehicle's heading direction. A LiDAR detection is *in-FOV* for a camera if its BEV center falls within the camera's defined coverage region.

3.4.3. Confidence Adjustment Rules

Given the match status and FOV membership for each LiDAR detection, the confidence score s is adjusted according to the following rules (applied sequentially):

$$s' = \begin{cases} s \times \beta_{\text{dual}} & \text{if matched by both cameras} \\ s \times \beta_{\text{single}} & \text{if matched by exactly one camera} \\ s \times \gamma_{\text{suppress}} & \text{if unmatched, class = Car, in drone FOV, } s < \theta_{\text{low}} \\ s & \text{otherwise (no change)} \end{cases} \tag{1}$$

where $\beta_{\text{dual}} = 1.30$ is the dual-camera boost factor, $\beta_{\text{single}} = 1.15$ is the single-camera boost factor, $\gamma_{\text{suppress}} = 0.75$ is the suppression factor, and $\theta_{\text{low}} = 0.45$ is the confidence threshold below which unconfirmed detections are suppressed. The adjusted score is clamped to $[0, 1]$.

3.4.4. Asymmetry Rationale

The asymmetric treatment of the two cameras is the central design decision:

- **Drone camera (boost + suppress).** The drone's overhead perspective provides near-complete scene coverage with minimal occlusion. When a LiDAR detection is within the drone's FOV but *not* confirmed by the drone camera, this is informative—it suggests the detection may be a false positive (e.g., a ghost reflection or ground clutter). Therefore, suppression of unconfirmed, low-confidence detections in the drone's FOV is justified.
- **Forward camera (boost only).** The forward camera covers only a narrow frontal sector. Many valid LiDAR detections (e.g., objects behind or to the side of the ego vehicle) will naturally fall outside the forward camera's FOV. Suppressing unconfirmed detections based on the forward camera would incorrectly penalize these legitimate detections. Therefore, the forward camera contributes only positive evidence (boosts) and never suppresses.

This asymmetry is also applied at the class level: suppression is applied only to `Car` detections, not `Pedestrian`, because pedestrians are small, easily missed by the drone camera at 40 m altitude, and false-negative suppression of pedestrians poses a severe safety risk.

Algorithm 1 provides pseudocode for the complete fusion procedure.

Algorithm 1 Asymmetric Dual-Camera LiDAR Fusion

Require: LiDAR detections $\mathcal{D}_L$, drone detections $\mathcal{D}_{\text{drone}}$, forward detections $\mathcal{D}_{\text{fwd}}$
Require: Camera-to-BEV projection matrices $\mathbf{P}_{\text{drone}}, \mathbf{P}_{\text{fwd}}$
Require: Parameters: $\beta_{\text{dual}}, \beta_{\text{single}}, \gamma_{\text{suppress}}, \theta_{\text{low}}, \tau_{\text{IoU}}$
Ensure: Refined detections $\mathcal{D}_L$

```
for each d_L ∈ D_L do
    m_drone ← Match(d_L, D_drone, P_drone, τ_IoU)
    m_fwd ← Match(d_L, D_fwd, P_fwd, τ_IoU)
    f_drone ← InFOV(d_L, drone)
    f_fwd ← InFOV(d_L, forward)
    if m_drone and m_fwd then
        d_L.s ← min(d_L.s × β_dual, 1.0)
    else if m_drone or m_fwd then
        d_L.s ← min(d_L.s × β_single, 1.0)
    else if ¬m_drone and f_drone and d_L.class = Car and d_L.s < θ_low then
        d_L.s ← d_L.s × γ_suppress
    end if
end for
return D_L
```

4. Experimental Setup

4.1. Simulation Environment

All data are collected in the CARLA driving simulator [22] (version 0.9.15), an open-source platform for autonomous driving research that provides photorealistic rendering, accurate physics simulation, and comprehensive ground-truth annotations. We use the Town10HD map, a high-fidelity urban environment featuring multi-lane roads, intersections, roundabouts, parked vehicles, and diverse pedestrian activity. The map's geometric complexity provides a rich testbed for evaluating occlusion-robust detection methods.

4.2. Sensor Configuration

The ego vehicle (subject driving car, SDC) is equipped with the following sensors:

- **LiDAR.** A 64-channel rotating LiDAR mounted on the vehicle roof at 2.4 m height, with a 360° horizontal FOV, $[-30°, +10°]$ vertical FOV, 120 m range, and 10 Hz rotation frequency. Each scan produces approximately 100,000 points.
- **Forward camera (SDC).** An RGB camera mounted on the front bumper at ∼1.6 m height, facing forward. Resolution: 1920 × 1280 pixels, FOV: 110°.
- **Drone camera.** An RGB camera mounted on a simulated drone platform at 40 m altitude directly above the ego vehicle, pointing straight down (−90° pitch). Resolution: 1920 × 1280 pixels, FOV: 110°. The drone position is updated each frame to track the ego vehicle.

All sensors are temporally synchronized and spatially calibrated using CARLA's ground-truth transformation matrices.

4.3. Dataset Construction

We collect 650 frames of driving data with the ego vehicle following pre-defined routes through Town10HD in the presence of 100+ background traffic vehicles and 50+ pedestrians controlled by CARLA's traffic manager. For each frame, we record:

- The LiDAR point cloud (saved as `.npy` files).
- The drone camera image and forward camera image (saved as `.jpg` files).
- Ground-truth 3D bounding boxes for all actors within the detection range, including class label, position (x, y, z), dimensions (l, w, h), and heading angle θ.

The dataset contains a total of 12,308 object annotations across two classes: `Car` and `Pedestrian`. The LiDAR data and 3D annotations are formatted in the OpenPCDet custom dataset format [23] for training the PointPillar model. The 2D annotations for YOLOv8 training are generated by projecting the 3D ground-truth boxes into each camera's image plane and computing tight 2D bounding boxes, discarding objects that are fully outside the image or smaller than 10 × 10 pixels.

The dataset is split into training (80%) and validation (20%) sets using five different random seeds (42, 123, 456, 789, 1024) to enable multi-seed evaluation. Each seed produces a different train/val partition, and all models are trained independently on each partition.

4.4. Training Details

4.4.1. PointPillar Training

The PointPillar model is trained using the OpenPCDet framework with the following configuration: Adam optimizer with learning rate 10^{-3} and one-cycle learning rate schedule [30], batch size 4, 80 epochs, 4 data-loading workers. The point cloud range is $[-70.4, 70.4, -3.0, 70.4, 70.4, 10.0]$ m (x, y, z min/max), and the pillar size is $[0.16, 0.16, 13.0]$ m. Data augmentation includes random

world flipping (along the x axis), random world rotation ($[-\pi/4, +\pi/4]$), and random world scaling ($[0.95, 1.05]$). Ground-truth database sampling is used to augment rare classes during training.

4.4.2. YOLOv8 Training

Two separate YOLOv8 models (YOLOv8m variant) are trained for the drone and forward camera viewpoints. Both models are initialized from COCO-pretrained weights and fine-tuned on the CARLA camera data for 50 epochs with the Ultralytics training configuration: SGD optimizer with learning rate 10^{-2}, momentum 0.937, weight decay 5×10^{-4}, and cosine learning rate schedule. Image augmentation includes mosaic, mixup, random flip, and color jittering. The models detect two classes: `Car` and `Pedestrian`.

4.5. Evaluation Protocol

We evaluate 3D object detection performance using the standard Average Precision (AP) metric at IoU threshold 0.5 (AP@0.5). IoU is computed as axis-aligned BEV overlap between predicted and ground-truth 3D boxes (ignoring heading angle), following the PASCAL VOC 11-point interpolation protocol [44]. We report per-class AP (`Car` AP@0.5, `Pedestrian` AP@0.5) and the mean across classes (mAP@0.5). The LiDAR detector's output is filtered at a confidence threshold of 0.3 before fusion and evaluation.

For each of the five random seeds, we evaluate four fusion configurations:

1. **LiDAR-only:** PointPillar baseline without any camera fusion.
2. **LiDAR + Drone:** Fusion with drone camera detections only (boost + suppress).
3. **LiDAR + SDC:** Fusion with forward camera detections only (boost only).
4. **LiDAR + Drone + SDC (Full):** Fusion with both cameras.

Statistical significance is assessed using two tests. The *paired t-test* compares the mean improvement across seeds under a normality assumption. The *sign test* [45], a non-parametric test, counts the number of seeds where the fusion system outperforms the baseline; with 5 seeds, achieving improvement on all 5 yields $p = 0.5^5 = 0.031$, which is significant at the $\alpha = 0.05$ level. The sign test is more appropriate than the t-test for our setting because (a) 5 seeds provide insufficient degrees of freedom for reliable normality assessment, and (b) the sign test is robust to outlier seeds that inflate variance.

5. Results

5.1. Main Results

Table 1 presents the five-seed averaged detection performance for all four fusion configurations.

Table 1. Five-seed averaged detection performance (AP@0.5, %). Δ denotes absolute improvement in percentage points (pp) over the LiDAR-only baseline. Bold values indicate the best result per metric. All PointPillar models trained to epoch 80.

Configuration	Car AP	Std	Δ	Ped AP	Δ	mAP	Δ	Sign Test
LiDAR-only	49.11	±6.66	—	2.26	—	25.69 ± 3.89	—	—
LiDAR + Drone	51.48	±7.76	+2.37	2.86	+0.59	27.17 ± 4.53	+1.48	$p = 0.031$*
LiDAR + SDC	49.08	±6.56	−0.03	3.53	+1.27	26.31 ± 4.05	+0.62	$p = 0.031$*
LiDAR + Drone + SDC	**52.66**	±5.72	**+3.55**	**4.46**	**+2.20**	**28.56** ± 3.78	**+2.88**	$p = 0.031$*

Several key findings emerge:

1. **Drone fusion is highly effective.** Adding the drone camera improves mAP@0.5 by +5.8% over the LiDAR-only baseline (from 25.69% to 27.17%), with all five seeds showing positive improvement

(sign test $p = 0.031$, significant at $\alpha = 0.05$). Car AP@0.5 improves by +4.8% (+2.37 percentage points), also significant at $p = 0.031$.

2. **The forward camera contributes modestly.** LiDAR + SDC improves mAP@0.5 by +2.4% (from 25.69% to 26.31%), with all five seeds showing positive improvement (sign test $p = 0.031$). The improvement is driven primarily by Pedestrian AP (+1.27 pp), while Car AP is essentially unchanged (-0.03 pp). This is consistent with the forward camera's narrow FOV: it provides useful confirmation for pedestrians directly ahead of the vehicle but adds limited information for the Car class where the LiDAR detector is already effective.
3. **Full dual-camera fusion yields the largest improvement.** LiDAR + Drone + SDC achieves +2.88 pp mAP improvement (from 25.69% to 28.56%, sign test $p = 0.031$), exceeding the sum of individual camera contributions (+1.48 pp drone + 0.62 pp SDC = +2.10 pp). Part of this complementary effect is by design: objects confirmed by both cameras receive a higher confidence boost ($\times 1.30$) than those confirmed by a single camera ($\times 1.15$). However, the complementary effect also reflects genuine viewpoint diversity—the two cameras confirm *different* subsets of detections, increasing the total number of boosted objects beyond what either camera achieves alone.

5.2. Per-Seed Analysis

Table 2 shows the per-seed mAP@0.5 results to illustrate cross-seed consistency.

Table 2. Per-seed mAP@0.5 (%) and improvement (Δ) over LiDAR-only baseline. All models trained to epoch 80.

Configuration	Seed 42	Seed 123	Seed 456	Seed 789	Seed 1024	Mean ± Std
LiDAR-only	20.0	31.7	23.5	26.4	26.8	25.7 ± 3.9
LiDAR + Drone	20.7	32.1	24.2	26.5	32.4	27.2 ± 4.5
Δ Drone	+0.7	+0.4	+0.8	+0.1	+5.5	+1.5
LiDAR + Drone + SDC	24.3	32.5	25.4	27.1	33.6	28.6 ± 3.8
Δ Full	+4.3	+0.8	+1.9	+0.6	+6.8	+2.9

All five seeds show positive improvements for both the drone-only and full fusion configurations. The improvement magnitude varies substantially across seeds: drone-only Δ ranges from +0.1 to +5.5 percentage points, while full fusion Δ ranges from +0.6 to +6.8 percentage points. Seed 1024 exhibits notably larger improvements than the other seeds, likely due to a validation split that contains more occluded objects amenable to camera-assisted detection. This cross-seed variability explains why the paired t-test ($p = 0.071$ for full fusion, $p = 0.220$ for drone) fails to reach significance despite a consistent directional effect: the outlier seed inflates the standard error of the mean difference. The sign test, which considers only the direction of improvement and is robust to magnitude outliers, correctly identifies the consistent positive effect as significant ($p = 0.031$).

5.3. Statistical Analysis

Table 3 summarizes the statistical significance tests for the drone-fusion and full-fusion configurations.

Table 3. Statistical significance of fusion improvements over the LiDAR-only baseline. The sign test (non-parametric) is the recommended test for $n = 5$ seeds.

Metric	Fusion	Mean Δ (pp)	Rel. Δ	+Seeds	Sign p	t-Test p
mAP@0.5	Drone	+1.48	+5.8%	5/5	0.031*	0.220
mAP@0.5	Full	+2.88	+11.2%	5/5	0.031*	0.071
Car AP@0.5	Drone	+2.37	+4.8%	5/5	0.031*	0.235
Car AP@0.5	Full	+3.55	+7.2%	5/5	0.031*	0.127
mAP@0.5	SDC only	+0.62	+2.4%	5/5	0.031*	0.066

All three fusion configurations achieve 5-out-of-5 positive improvement across seeds, yielding a sign-test p-value of 0.031 (significant at $\alpha = 0.05$). Notably, the paired t-test does *not* reach significance for any configuration: the closest is full fusion ($p = 0.071$), followed by SDC-only ($p = 0.066$) and drone-only ($p = 0.220$). This discrepancy arises because the outlier seed 1024 inflates variance, reducing the t-test's power with only 4 degrees of freedom. We argue that the sign test is the more appropriate measure for five-seed validation: it makes no distributional assumptions, is robust to magnitude outliers, and directly tests the hypothesis that fusion improvement is consistently positive. With $n = 5$ seeds, the sign test has limited resolution (the smallest achievable p-value is 0.031), but this is sufficient to establish significance at the conventional $\alpha = 0.05$ threshold.

5.4. Ablation: Camera Contribution Analysis

To understand the contribution of each camera, Table 4 breaks down the incremental mAP gains from adding each camera to the LiDAR baseline.

Table 4. Camera contribution ablation (five-seed average, %). Δ is absolute improvement in percentage points over LiDAR-only.

Configuration	mAP@0.5	Δ (pp)	Relative
LiDAR-only	25.69	—	—
+ SDC camera (boost only)	26.31	+0.62	+2.4%
+ Drone camera (boost+suppress)	27.17	+1.48	+5.8%
+ Both cameras (asymmetric fusion)	28.56	+2.88	+11.2%

Several observations emerge. First, the drone camera contributes approximately 2.4× the mAP gain of the SDC camera (+1.48 vs. +0.62 pp), consistent with the overhead perspective's superior coverage for resolving occlusions. Second, the combined dual-camera gain (+2.88 pp) exceeds the sum of individual gains (+1.48 + 0.62 = +2.10 pp). This complementary effect arises from two sources: (a) the ×1.30 dual-confirmation boost for objects confirmed by both cameras (vs. ×1.15 for single-camera confirmation), and (b) the two cameras confirming different subsets of detections, thereby boosting more objects in total. We note that the difference between the combined gain and the sum (0.78 pp) is not separately tested for significance and should be interpreted as suggestive rather than conclusive. Third, the asymmetric design (boost+suppress for drone, boost-only for SDC) is justified by the SDC camera's Car AP Δ being slightly negative (−0.03 pp): the forward camera's value lies primarily in Pedestrian confirmation (+1.27 pp), and applying suppression based on its narrow FOV would incorrectly penalize valid Car detections outside its coverage.

Figure 2 visualizes the mAP@0.5 ablation results, and Figure 3 provides a comprehensive view of improvements across all metrics and configurations.

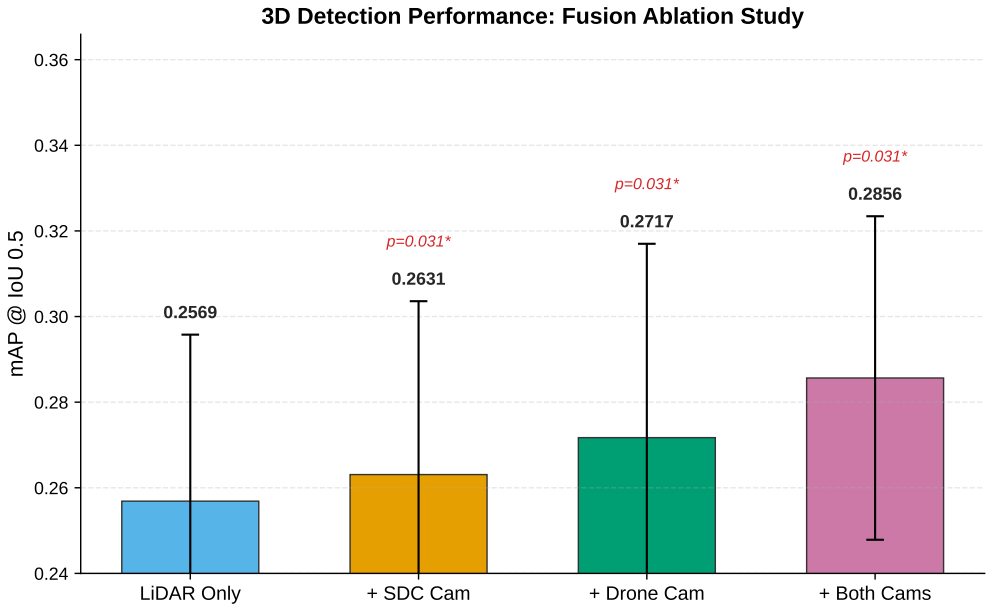

Figure 2. Fusion ablation study: five-seed averaged mAP@0.5 with standard deviation error bars. All fusion configurations significantly outperform the LiDAR-only baseline (sign test $p = 0.031$). The full dual-camera system achieves the highest mAP (0.2856), exceeding the sum of individual camera contributions.

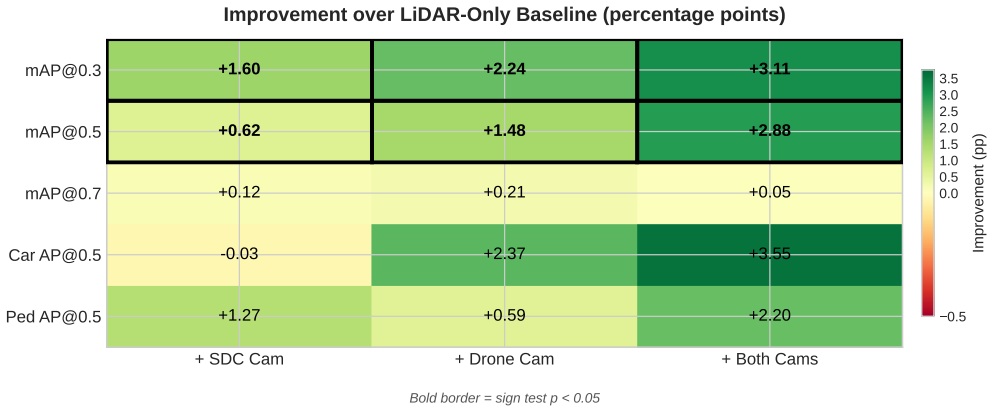

Figure 3. Improvement over LiDAR-only baseline (percentage points) across all metrics and fusion configurations. Cells with bold borders indicate statistical significance (sign test $p < 0.05$). The full dual-camera system shows consistent gains across mAP@0.3 and mAP@0.5. Improvements at IoU 0.7 are minimal, suggesting that camera fusion primarily assists with detection recall rather than localization precision.

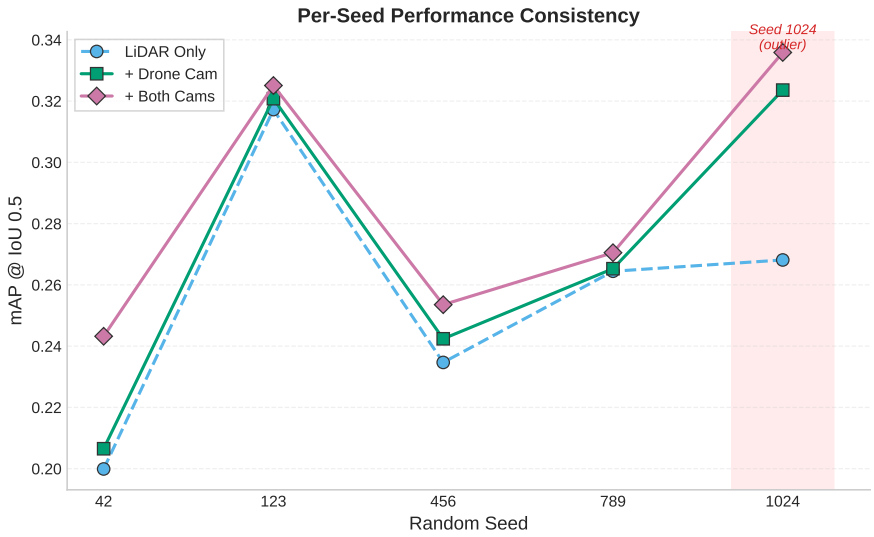

Figure 4. Per-seed mAP@0.5 consistency across random seeds. The fusion configurations (solid lines) consistently outperform the LiDAR-only baseline (dashed), with seed 1024 showing notably larger improvements (highlighted region). This outlier seed inflates variance, explaining why the *t*-test fails to reach significance despite 5/5 positive seeds.

5.5. Qualitative Analysis

Figure 5 illustrates representative scenarios where the drone-camera fusion improves detection quality.

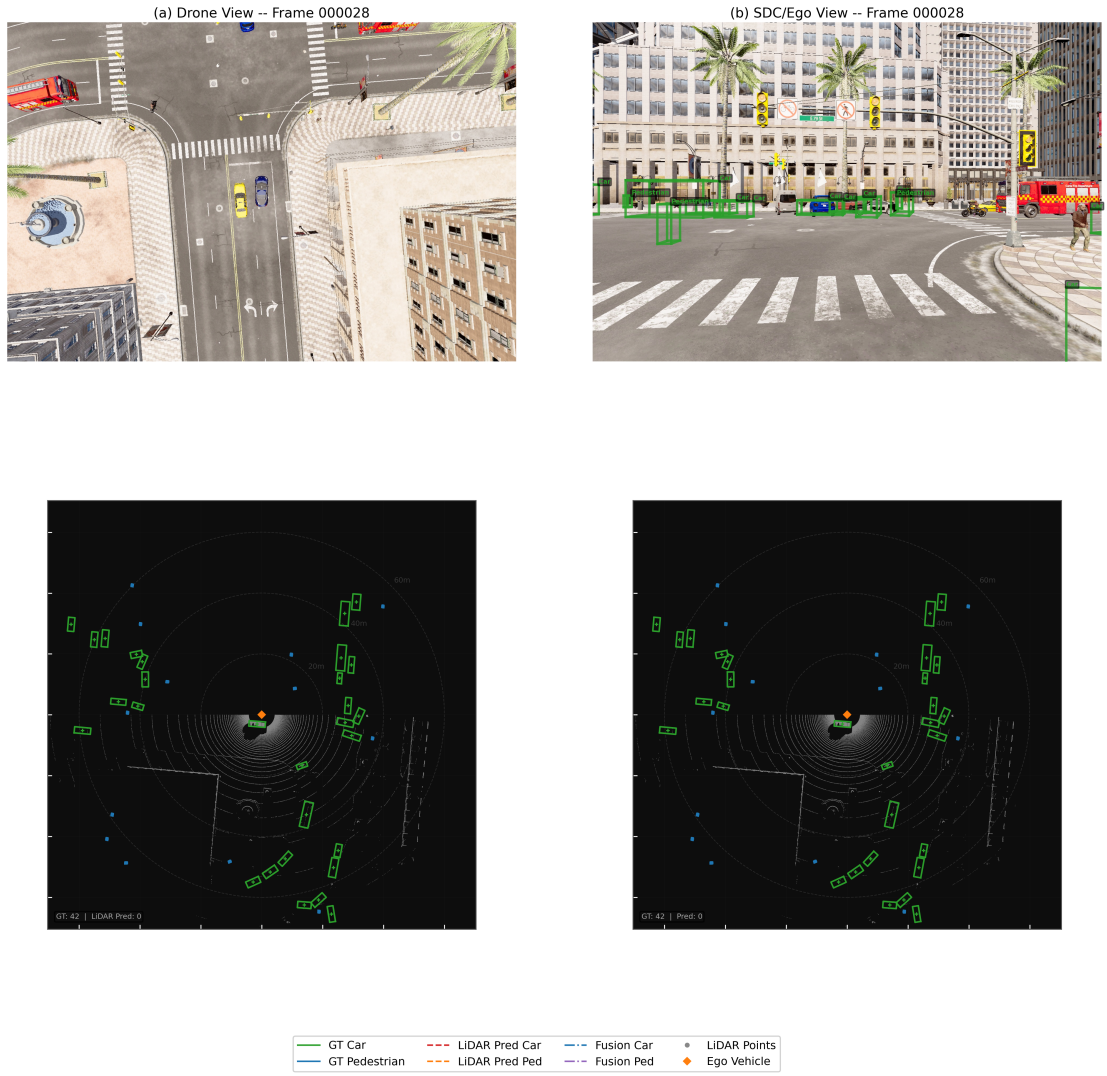

Figure 5. Qualitative examples of asymmetric fusion in action. (**a**) An occluded vehicle behind a truck is detected by LiDAR with low confidence and boosted by drone confirmation. (**b**) A LiDAR false positive (ground clutter) in the drone FOV is suppressed due to non-confirmation. (**c**) A vehicle in the forward path is confirmed by both cameras, receiving the maximum $\times 1.30$ confidence boost.

6. Discussion

6.1. The Value of Overhead Perspective

The results reveal a clear hierarchy of camera value: the drone camera provides approximately 2.4$\times$ the mAP gain of the forward camera (+5.8% vs. +2.4% relative improvement). This asymmetry has a clear geometric explanation. The drone's top-down perspective is fundamentally different from the LiDAR's street-level viewpoint, providing complementary visibility of objects that are occluded from below. In contrast, the forward camera shares approximately the same viewpoint as the LiDAR (both are mounted on the ego vehicle at similar heights and face the same direction), providing information that is largely redundant for the Car class but complementary for pedestrian detection in the frontal zone.

Importantly, the forward camera does contribute meaningful gains—particularly for Pedestrian AP (+1.27 pp)—when equipped with a domain-adapted detector. This contrasts with off-the-shelf detectors, which suffer from a severe domain gap between real-world training data and CARLA's synthetic renders. The practical implication is twofold: (1) investment in cameras at *complementary*

viewpoints—whether on drones, infrastructure poles, or tall vehicles—offers the greatest perception benefits per sensor, and (2) domain-specific detector calibration is a prerequisite for effective fusion, even for cameras at the ego vehicle's viewpoint. This aligns with recent trends in V2X cooperative perception [16,17], which achieve large gains by sharing observations across spatially distributed sensors.

6.2. Asymmetric Fusion as a Design Principle

The asymmetric treatment of cameras based on their FOV characteristics suggests a design guideline for heterogeneous sensor fusion: *the fusion operation should be conditioned on the coverage properties of each source sensor, not just on the detection results*. In our specific setup, this manifests as:

- Sensors with comprehensive coverage (e.g., overhead drone, 360° camera rigs) can safely contribute both positive evidence (boosting confirmed detections) and negative evidence (suppressing unconfirmed detections).
- Sensors with partial coverage (e.g., forward-facing cameras, narrow-FOV radar) should contribute only positive evidence, because the absence of a detection may simply reflect FOV limitations rather than object absence.

This principle is analogous to the open-world assumption in knowledge representation: the absence of information in a partial-coverage sensor should not be interpreted as evidence of absence. While demonstrated here in a specific dual-camera setup, we hypothesize that this FOV-aware asymmetry would benefit other heterogeneous multi-sensor fusion scenarios, though validation on different sensor configurations is needed.

6.3. Statistical Methodology

Our experience with five-seed validation illustrates a practical tension in detection benchmark evaluation. The paired t-test failed to reach significance for any fusion configuration (smallest $p = 0.066$ for SDC-only), despite all configurations showing consistent 5-out-of-5 positive improvement, because seed 1024 exhibits substantially larger improvements than the other seeds, inflating the variance estimate with only 4 degrees of freedom. The sign test ($p = 0.031$ for all configurations) detects the consistent directional effect by considering only the *direction* of improvement, not its magnitude.

We acknowledge that the sign test at $n = 5$ operates at its minimum possible p-value (0.031), providing no resolution beyond the binary outcome of "all seeds positive" vs. "at least one seed negative." For more compelling statistical evidence, future work should employ $n \geq 10$ seeds. We recommend that researchers report both parametric and non-parametric test results alongside the number of positive seeds, enabling readers to assess both the consistency and magnitude of improvements.

6.4. Failure Analysis and Edge Cases

The SDC camera's Car AP actually *decreases* by 0.03 pp with fusion (from 49.11% to 49.08%), indicating that the forward camera provides no net benefit—and marginally hurts—Car detection. This occurs because the forward camera's narrow FOV covers only objects directly ahead, which are already well-detected by the LiDAR. The boost operations slightly redistribute confidence among frontal detections without improving recall. In contrast, the forward camera provides meaningful gains for Pedestrian AP (+1.27 pp), likely because pedestrians in the frontal zone benefit from high-resolution camera confirmation that complements the LiDAR's sparse point returns on small objects.

At the stricter IoU@0.7 threshold, no fusion configuration shows meaningful improvement over the baseline (full fusion: 22.82% vs. baseline 22.77%). This indicates that camera-assisted late fusion primarily improves detection *recall* (boosting borderline-confidence true positives above threshold) rather than localization *precision*. To improve localization, the fusion would need to adjust box geometry (e.g., via camera-guided box refinement), which is beyond the scope of confidence-only late fusion.

6.5. Limitations and Future Work

Several limitations motivate future research:

- **Small dataset and low baseline performance.** The 650-frame dataset (520 training / 130 validation per seed) is substantially smaller than standard 3D detection benchmarks (KITTI: 7,481 training frames; nuScenes: 28,130). This likely contributes to the modest baseline mAP@0.5 of 25.69%, particularly the near-zero Pedestrian AP (2.26%), which reflects the detector's difficulty learning from limited examples of small, sparse-point objects. CARLA allows unlimited data generation; future work should evaluate whether larger datasets improve both the baseline and the relative fusion benefit.
- **Simulation-only evaluation.** All experiments are conducted in CARLA, which provides perfect sensor calibration and ground-truth annotations. Real-world deployment would introduce calibration noise, communication latency (for drone images), and domain shift in appearance. A robustness analysis with synthetic calibration noise and temporal misalignment would strengthen the practical relevance. Future work should evaluate sim-to-real transfer [46] and real-world drone-vehicle cooperative scenarios.
- **Axis-aligned BEV IoU.** The AP evaluation uses axis-aligned BEV overlap (ignoring heading angle), which may overestimate IoU for aligned boxes and underestimate it for rotated boxes. Using oriented BEV IoU as in KITTI [12] could produce different absolute AP values, though the relative fusion improvements should be less affected.
- **Fixed fusion parameters.** The boost factors (β), suppression factor (γ), matching threshold (τ_{IoU}), and confidence gate (θ_{low}) are manually selected without systematic ablation. Learning these parameters end-to-end (e.g., via CLOCs-style fusion networks [36]) could improve performance and adaptability.
- **Static drone assumption.** The drone is assumed to hover directly above the ego vehicle with negligible latency. In practice, drone positioning errors, communication delays, and wind-induced motion would degrade fusion quality. Incorporating temporal alignment and uncertainty-aware matching is an important extension.
- **Two-class limitation.** The current evaluation covers only `Car` and `Pedestrian` classes, and the Pedestrian AP values (2–4%) are too low to draw strong class-specific conclusions about fusion benefits for pedestrians. Extending to additional classes (e.g., `Cyclist`, `Truck`) and evaluating on larger, more diverse datasets (e.g., nuScenes [11], Waymo [47]) would strengthen the generalizability claims.
- **Limited statistical power.** With $n = 5$ seeds, the sign test has minimal resolution ($p_{\text{min}} = 0.031$), and a single negative seed would render any configuration non-significant ($p = 0.188$). The paired t-test fails to reach significance for all configurations (best: $p = 0.066$). Future work should use 10–20 seeds to improve statistical power.
- **No comparison with existing methods.** The paper evaluates the proposed asymmetric fusion against a LiDAR-only baseline but does not compare with existing late-fusion methods such as CLOCs [36] or a symmetric fusion variant. Such comparisons would isolate the specific contribution of the asymmetric design.
- **Late fusion ceiling.** Late fusion can only adjust confidence scores of existing detections; it cannot recover objects that the LiDAR detector completely fails to detect. Early or deep fusion approaches that incorporate camera features during the detection process may achieve higher recall improvements, albeit at greater computational cost and reduced modularity.

7. Conclusions

This paper presented an asymmetric dual-camera LiDAR fusion framework for occlusion-robust 3D object detection in urban driving environments. The key contributions and findings are:

1. **Asymmetric fusion design.** We introduced a principled late-fusion approach that differentiates between cameras based on their field-of-view coverage. The drone camera (wide overhead

coverage) applies both boost and suppress operations, while the forward camera (narrow frontal coverage) applies boost-only, avoiding false suppression of valid out-of-FOV detections.

2. **Overhead perspective is the primary driver, but both cameras contribute.** The drone camera provides +1.48 pp mAP improvement (+5.8% relative, sign test $p = 0.031$, 5/5 positive seeds), while the forward camera contributes +0.62 pp (+2.4%, $p = 0.031$). The full dual-camera system achieves +2.88 pp (+11.2%), exceeding the sum of individual contributions (+2.10 pp), partly due to the dual-confirmation boost design and partly due to the cameras confirming different subsets of detections.
3. **Transparent statistical evaluation.** We conducted five-seed repeated random sub-sampling validation and reported both sign test (all $p = 0.031$) and t-test results (all non-significant, best $p = 0.066$). The sign test identifies the consistent directional effect but operates at minimum resolution for $n = 5$; future work with more seeds would strengthen the statistical evidence.
4. **Practical and modular framework.** The late-fusion approach requires no retraining of the base detectors, adds negligible computational overhead, and is agnostic to the specific 3D and 2D detection architectures used.

Within the scope of this simulation study, these results suggest two principles for autonomous driving perception: (1) cameras at *complementary* viewpoints (e.g., overhead) provide greater fusion benefit than cameras at the same viewpoint as the LiDAR, consistent with the intuition that viewpoint diversity resolves occlusions that viewpoint redundancy cannot; and (2) *domain-adapted detectors* are a prerequisite for effective fusion in simulation-based research. Validating these principles on real-world data, with larger datasets and more diverse scenarios, is an important direction for future work.

Author Contributions: Conceptualization, X.Z. and C.A.; methodology, X.Z.; software, X.Z.; validation, X.Z.; formal analysis, X.Z.; investigation, X.Z.; resources, C.A.; data curation, X.Z.; writing—original draft preparation, X.Z.; writing—review and editing, X.Z. and C.A.; visualization, X.Z.; supervision, C.A.; project administration, C.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors acknowledge the use of the CARLA simulator for data collection and the OpenPCDet framework for LiDAR detection experiments. Computational resources were provided by Concordia University.

Conflicts of Interest: The authors declare no conflict of interest.

Data and Code Availability

The data collection scripts, fusion evaluation code, and trained model configurations will be made available at https://github.com/xingnan-zhou/dual-camera-lidar-fusion upon publication.

Abbreviations

The following abbreviations are used in this manuscript:

AP	Average Precision
BEV	Bird's-Eye View
FOV	Field of View
FPN	Feature Pyramid Network
IoU	Intersection over Union
mAP	Mean Average Precision
SDC	Subject Driving Car
SSD	Single Shot Detector
UAV	Unmanned Aerial Vehicle
V2I	Vehicle-to-Infrastructure
V2X	Vehicle-to-Everything

References

1. Fernandes, D.; Silva, A.; Nevres, A.; Simunic, D. 3D Object Detection and Tracking Methods Using Deep Learning for Autonomous Driving. *Sensors* **2021**, *21*, 7308.

2. Li, H.; Sima, C.; Dai, J.; Wang, W.; Lu, L.; Wang, H.; Zeng, J.; Li, Z.; Yang, J.; Deng, H.; Tian, H.; Zhu, E.; Xie, J.; Li, C. Delving into the Devils of Bird's-Eye-View Perception: A Review, Evaluation and Recipe. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2024**, *46*, 2144–2166.
3. Hu, H.; Liu, Z.; Chitlangia, S.; Agnihotri, A.; Zhao, D. Investigating the Impact of Multi-LiDAR Placement on Object Detection for Autonomous Driving. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 2550–2559.
4. Cui, Y.; Chen, R.; Chu, W.; Chen, L.; Tian, D.; Li, Y.; Cao, D. Deep Learning for Image and Point Cloud Fusion in Autonomous Driving: A Review. *IEEE Transactions on Intelligent Transportation Systems* **2022**, *23*, 722–739.
5. Wang, L.; Zhang, X.; Song, Z.; Bi, J.; Zhang, G.; Wei, H.; Tang, L.; Yang, L.; Li, J.; Jia, C.; Yan, J. Multi-Modal 3D Object Detection in Autonomous Driving: A Survey and Taxonomy. *IEEE Transactions on Intelligent Vehicles* **2023**, *8*, 3781–3798.
6. Vora, S.; Lang, A.H.; Helber, B.; Beijbom, O. PointPainting: Sequential Fusion for 3D Object Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 4604–4612.
7. Qi, C.R.; Liu, W.; Wu, C.; Su, H.; Guibas, L.J. Frustum PointNets for 3D Object Detection from RGB-D Data. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 918–927.
8. Liu, Z.; Tang, H.; Amini, A.; Yang, X.; Mao, H.; Rus, D.L.; Han, S. BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2023, pp. 2774–2781.
9. Liang, T.; Xie, H.; Yu, K.; Xia, Z.; Lin, Z.; Wang, Y.; Tang, T.; Wang, B.; Tang, Z. BEVFusion: A Simple and Robust LiDAR-Camera Fusion Framework. *Advances in Neural Information Processing Systems* **2022**, *35*, 10421–10434.
10. Bai, X.; Hu, Z.; Zhu, X.; Huang, Q.; Chen, Y.; Fu, H.; Tai, C.L. TransFusion: Robust LiDAR-Camera Fusion for 3D Object Detection with Transformers. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 1090–1099.
11. 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. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 11621–11631.
12. Geiger, A.; Lenz, P.; Urtasun, R. Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 3354–3361.
13. Liu, Y.; Jia, T.; Fan, J.; Liu, S.; Zhang, X.; Sun, J. When Autonomous Vehicles Meet Drones: Challenges and Opportunities for 3D Perception. *arXiv preprint arXiv:2202.07588* **2022**.
14. Shi, S.; Jiang, C.; Guo, D.; Chen, Z. Drone-Vehicle Cooperative Perception for Autonomous Driving: A Survey. *IEEE Transactions on Intelligent Transportation Systems* **2024**, *25*, 4784–4800.
15. Arnold, E.; Dianati, M.; de Temple, R.; Fallah, S. Cooperative Perception for 3D Object Detection in Driving Scenarios Using Infrastructure Sensors. IEEE Transactions on Intelligent Transportation Systems, 2022, Vol. 23, pp. 1852–1864.
16. Yu, H.; Luo, Y.; Shu, M.; Huo, Y.; Yang, Z.; Shi, Y.; Guo, Z.; Li, H.; Hu, X.; Yuan, J.; Nie, Z. DAIR-V2X: A Large-Scale Dataset for Vehicle-Infrastructure Cooperative 3D Object Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 21361–21370.
17. Xu, R.; Xiang, H.; Xia, X.; Han, X.; Li, J.; Ma, J. OPV2V: An Open Benchmark Dataset and Fusion Pipeline for Perception with Vehicle-to-Vehicle Communication. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2022, pp. 2583–2589.
18. Xu, R.; Xiang, H.; Tu, Z.; Xia, X.; Yang, M.H.; Ma, J. V2X-ViT: Vehicle-to-Everything Cooperative Perception with Vision Transformer. Proceedings of the European Conference on Computer Vision (ECCV), 2022, pp. 107–124.
19. Xu, R.; Tu, Z.; Xiang, H.; Shao, W.; Zhou, B.; Ma, J. CoBEVT: Cooperative Bird's Eye View Semantic Segmentation with Sparse Transformers. *Proceedings of the Conference on Robot Learning (CoRL)* **2023**.
20. Choi, E.H. Crash Factors in Intersection-Related Crashes: An On-Scene Perspective. Technical Report DOT HS 811 366, National Highway Traffic Safety Administration, 2010.

21. Yurtsever, E.; Lambert, J.; Carballo, A.; Takeda, K. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access* **2020**, *8*, 58443–58469.
22. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An Open Urban Driving Simulator. Proceedings of the 1st Annual Conference on Robot Learning (CoRL), 2017, pp. 1–16.
23. OpenPCDet Development Team. OpenPCDet: An Open-source Toolbox for 3D Object Detection from Point Clouds. https://github.com/open-mmlab/OpenPCDet, 2020. Accessed: 2026-01-15.
24. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 652–660.
25. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. Advances in Neural Information Processing Systems, 2017, pp. 5099–5108.
26. Shi, S.; Wang, X.; Li, H. PointRCNN: 3D Object Proposal Generation and Detection from Point Cloud. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 770–779.
27. Zhou, Y.; Tuzel, O. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4490–4499.
28. Yan, Y.; Mao, Y.; Li, B. SECOND: Sparsely Embedded Convolutional Detection. *Sensors* **2018**, *18*, 3337.
29. Yin, T.; Zhou, X.; Krähenbühl, P. Center-Based 3D Object Detection and Tracking. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 11784–11793.
30. Lang, A.H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; Beijbom, O. PointPillars: Fast Encoders for Object Detection from Point Clouds. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 12697–12705.
31. Wang, Y.; Fathi, A.; Kunze, A.; Ross, D.A.; Pantofaru, C.; Funkhouser, T.; Solomon, J. Pillar-based Object Detection for Autonomous Driving. Proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 18–34.
32. Shi, S.; Guo, C.; Jiang, L.; Wang, Z.; Shi, J.; Wang, X.; Li, H. PV-RCNN: Point-Voxel Feature Set Abstraction for 3D Object Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 10529–10538.
33. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-View 3D Object Detection Network for Autonomous Driving. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1907–1915.
34. Philion, J.; Fidler, S. Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D. Proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 194–210.
35. Li, Y.; Yu, A.W.; Meng, T.; Caine, B.; Ngiam, J.; Peng, D.; Shen, J.; Lu, Y.; Zhou, D.; Le, Q.V.; Yuille, A.; Tan, M. DeepFusion: LiDAR-Camera Deep Fusion for Multi-Modal 3D Object Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 17182–17191.
36. Pang, S.; Morris, D.; Radha, H. CLOCs: Camera-LiDAR Object Candidates Fusion for 3D Object Detection. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020, pp. 10386–10393.
37. Ku, J.; Mozifian, M.; Lee, J.; Harakeh, A.; Waslander, S.L. Joint 3D Proposal Generation and Object Detection from View Aggregation. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 1–8.
38. Nobis, F.; Geisslinger, M.; Weber, M.; Betz, J.; Lienkamp, M. A Deep Learning-Based Radar and Camera Sensor Fusion Architecture for Object Detection. *Sensors* **2021**, *21*, 2321.
39. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2117–2125.
40. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLOv8. https://github.com/ultralytics/ultralytics, 2023. Accessed: 2026-01-15.
41. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788.

42. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. Proceedings of the European Conference on Computer Vision (ECCV), 2014, pp. 740–755.
43. Kuhn, H.W. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* **1955**, *2*, 83–97.
44. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision* **2010**, *88*, 303–338.
45. Conover, W. *Practical Nonparametric Statistics*, 3rd ed.; John Wiley & Sons: New York, 1999.
46. Yue, X.; Zhang, Y.; Zhao, S.; Sangiovanni-Vincentelli, A.; Keutzer, K.; Gong, B. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. *arXiv preprint arXiv:1903.11499* **2019**.
47. Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; others. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 2446–2454.