



OPEN

## FDSNet: dynamic multimodal fusion stage selection for autonomous driving via feature disagreement scoring

Asaad Mohammed, Hosny M. Ibrahim & Nagwa M. Omar✉

Robust and efficient 3D perception is critical for autonomous vehicles operating in complex environments. Multi-sensor fusion, such as Camera+LiDAR, Camera+Radar, or all three modalities, significantly enhances scene understanding. However, most existing frameworks fuse data at a fixed stage, categorized as early fusion (raw data level), mid fusion (intermediate feature level), or late fusion (detection output level), neglecting semantic consistency across modalities. This static strategy may result in performance degradation or unnecessary computation under sensor misalignment or noise. In this work, we propose FDSNet (Feature Disagreement Score Network), a dynamic fusion framework that adaptively selects the fusion stage based on measured semantic consistency across sensor modalities. Each sensor stream (Camera, LiDAR, and Radar) independently extracts mid-level features, which are then transformed into a common Bird's Eye View (BEV) representation, ensuring spatial alignment across modalities. To assess agreement, a Feature Disagreement Score (FDS) is computed at each BEV location by measuring statistical deviation across modality features. These local scores are aggregated into a global FDS value, which is compared against threshold to determine the fusion strategy. A low FDS, indicating strong semantic consistency across modalities, triggers mid-level fusion for computational efficiency, whereas a high FDS value activates late fusion to preserve detection robustness under cross-modal disagreement. We evaluate FDSNet on the nuScenes dataset across multiple configurations: Camera+Radar, Camera+LiDAR, and Camera+Radar+LiDAR. Experimental results demonstrate that FDSNet achieves consistent improvements over recent multimodal baselines, with gains of up to +3.0% in NDS and +2.6% in mAP on the validation set, and +2.1% in NDS and +1.6% in mAP on the test set, highlighting that dynamic stage selection provides both robustness and quantifiable advantages over static fusion strategies.

Autonomous vehicles (AVs) are poised to revolutionize transportation, promising significant improvements in road safety, reductions in traffic congestion, and enhanced mobility for diverse user groups<sup>1,2</sup>. Collision avoidance in autonomous driving unfolds through key stages enabled by robust 3D perception: (1) detection and localization of surrounding objects<sup>3</sup>, (2) interpretation of the scene context<sup>4</sup>, and (3) informed real time navigation decisions to avoid hazards and ensure safe trajectory planning<sup>5,6</sup>. This task becomes particularly challenging in complex, dynamic, and unpredictable environments, demanding consistently reliable perception under varying lighting, weather, and traffic conditions.

Relying on a single sensor modality for 3D perception often falls short of delivering the necessary robustness and consistency. Each sensor type presents inherent limitations due to its physical sensing principles. Cameras offer detailed texture and color information but remain highly susceptible to lighting variations, performing poorly under low visibility, glare, or nighttime conditions<sup>7–10</sup>. LiDAR sensors provide accurate geometric and distance measurements but struggle with adverse weather conditions, in addition to their significant power demands and cost implications<sup>11,12</sup>. In contrast, Radar demonstrates greater resilience to environmental disturbances and supports long range sensing but is hindered by limited angular resolution and sparse point cloud data, restricting fine grained object detection and classification<sup>13,14</sup>.

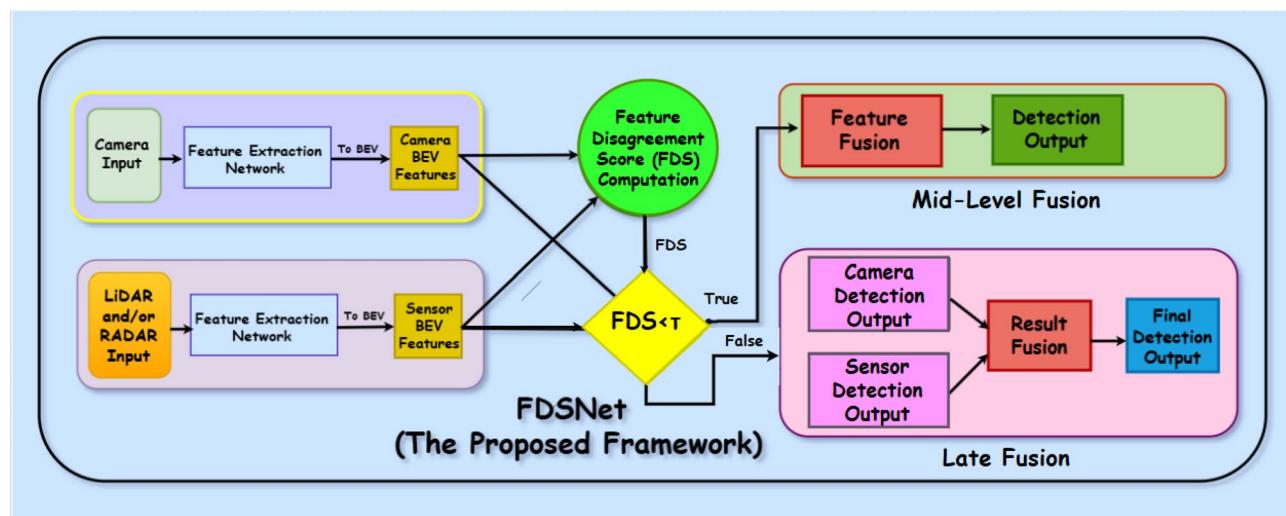
To overcome these individual sensor limitations, multi-sensor fusion has emerged as a foundational approach within modern AV perception systems. By integrating complementary information from different modalities such as Radar and Camera<sup>15,16</sup>, LiDAR and Camera<sup>17,18</sup>, or all three Camera, Radar, and LiDAR<sup>19,20</sup>,

Information Technology Department, Faculty of Computers and Information, Assiut University, Assiut 71515, Egypt. ✉email: n\_omar@aun.edu.eg

fusion frameworks construct richer and more resilient representations of the driving environment. This method improves detection accuracy, redundancy, and fault tolerance in real-world scenarios. Nonetheless, most existing fusion frameworks implement a static fusion strategy (early, mid, or late), uniformly applied regardless of scenario dynamics. Such rigid fusion designs neglect variability in cross-sensor consistency, which can fluctuate significantly due to environmental factors like sensor misalignment, occlusion, or varying illumination conditions. For instance, early fusion combining raw or low-level data may inadvertently propagate sensor misalignments and noise throughout the system, reducing overall detection quality<sup>21</sup>. Mid-level fusion tends to exhibit stronger semantic consistency but still assumes that sensor features are consistently aligned<sup>22</sup>. Conversely, late fusion integrates at the decision stage, offering robustness to sensor noise but incurring higher computational overhead and latency, which is a critical drawback in real time AV systems<sup>23</sup>.

To mitigate these trade-offs, recent studies have explored hybrid fusion architectures, merging early-to-mid or mid-to-late fusion combinations<sup>24–26</sup>. These architectures capitalize on early-stage efficiency under optimal sensor alignment and switch to robust, later-stage fusion when inconsistencies arise. However, multi-path hybrid approaches tend to be computationally inefficient, as they execute all fusion branches simultaneously, regardless of necessity. This not only results in substantial memory usage and processing latency, hindering real-time AV applications, but also fails to account for the variability of real-world driving conditions. These challenges become especially critical in adverse scenarios such as heavy rain, fog, or nighttime driving, where camera perception deteriorates, or when LiDAR measurements are disrupted by snow and dust. Likewise, sensor occlusion caused by large vehicles or roadside infrastructure can lead to incomplete observations. Static fusion strategies, which rigidly apply a fixed fusion stage, are unable to adapt to such dynamic inconsistencies, often propagating noise or incurring unnecessary computation. This underscores the necessity of an adaptive fusion mechanism capable of adjusting to varying sensor reliability in real time. In parallel, advances in lightweight convolutional neural networks (CNNs) have highlighted the importance of designing architectures that achieve high accuracy while minimizing computational complexity. Such approaches have been applied successfully in diverse domains, including instrument indication recognition<sup>27</sup>, ancient mural element detection<sup>28</sup>, and biometric verification<sup>29</sup>. These studies demonstrate that carefully designed compact architectures can deliver robust performance under constrained resources, a principle directly aligned with the requirements of autonomous driving perception. Motivated by this trend, our proposed FDSNet extends efficiency-oriented design concepts to multimodal fusion, ensuring both robustness and computational efficiency in real-time scenarios.

Addressing these limitations, we propose FDSNet (Feature Disagreement Score Network), a dynamic fusion framework that adaptively selects the optimal fusion stage either mid-level or late level, based on semantic consistency across sensor modalities. Unlike prior approaches that rigidly execute multiple fusion stages or rely on fixed fusion pipelines, FDSNet introduces a Feature Disagreement Score (FDS) computed at the Bird's Eye View (BEV) level to quantify semantic inconsistencies between modalities. A global threshold applied to the FDS determines the fusion strategy dynamically, mid-level fusion is activated when semantic consistency is high, enhancing computational efficiency, whereas late-level fusion is triggered under significant disagreement, ensuring robustness. This conditional approach retains the advantages and flexibility of multi-stage fusion methods while significantly reducing redundant computation, making it particularly suited for real time autonomous driving applications. An overview of the FDSNet conditional switching mechanism is illustrated in Fig. 1.



**Fig. 1.** Overview of the proposed FDSNet framework for adaptive sensor fusion in 3D object detection. The framework computes a Feature Disagreement Score (FDS) based on BEV features from Camera and LiDAR/Radar branches. Depending on the FDS value, the system dynamically selects between mid-level (feature fusion) and late-level (result fusion) strategies, enabling robust performance across varying sensor reliability and environmental conditions.

## Our key contributions are summarized as follows

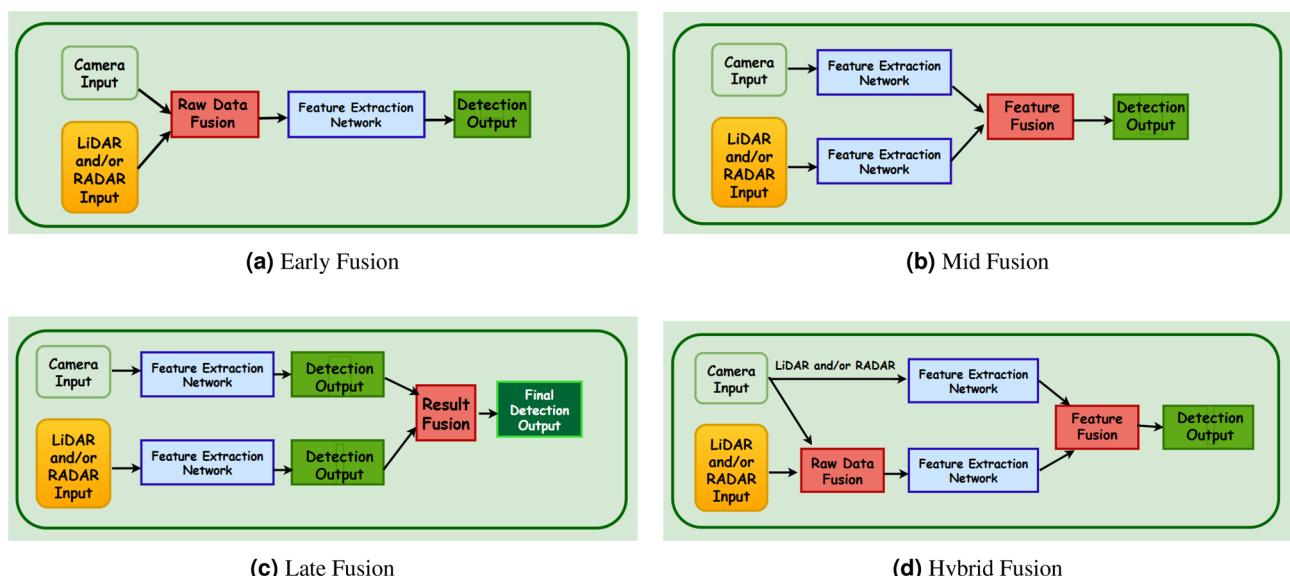
- We propose a dynamic fusion architecture that adaptively switches between mid-level and late-level fusion based on real-time semantic consistency among sensor modalities.
- We introduce the Feature Disagreement Score (FDS), a novel metric that quantifies semantic misalignment at the BEV level and guides conditional fusion decisions.
- We conduct comprehensive experiments on the nuScenes dataset<sup>30</sup>, evaluating three sensor configurations: LiDAR + Camera, Radar + Camera, and LiDAR + Radar + Camera. FDSNet achieves competitive 3D object detection accuracy while significantly reducing computational cost compared to static and hybrid fusion strategies.

## Related work

Multimodal fusion strategies are generally categorized into four main approaches: early, mid-level, late, and hybrid fusion. Each method provides unique trade-offs in terms of computational efficiency, semantic consistency, and adaptability. Fig. 2 illustrates the fundamental stages of these fusion strategies. This section reviews prominent methods in each category, highlighting their strengths and limitations.

### Early fusion

Early fusion combines multimodal sensor data at the input or raw feature stage, prior to independent feature extraction. This fusion strategy leverages the complementarity among different sensor modalities to generate richer input features but faces inherent limitations in computational efficiency and adaptability. A representative example is the **Multimodal Early Fusion with Attention** (MEFA) model proposed by Dupas et al.<sup>31</sup>. MEFA employs local and global attention modules to fuse visible, infrared, and LiDAR data into a single coherent representation at the raw data level. Specifically, the local attention mechanism captures spatially adjacent feature interactions within each modality, while the global attention mechanism facilitates cross-modal information propagation, enhancing semantic coherence across the fused data. This sophisticated hierarchical attention architecture significantly improves robustness against challenging environmental conditions, such as adverse weather or varying illumination scenarios. However, the computational complexity introduced by these multi-scale attention mechanisms results in substantial computational overhead, limiting the model's practicality for real time deployment in autonomous systems. Another prominent early fusion method is the **Virtual Sparse Convolution** (VirConv) framework presented by Wu et al.<sup>32</sup>. VirConv projects multi-view LiDAR point clouds and camera images into a common sparse voxel grid representation within a BEV. This voxel based representation enables efficient 3D object detection by leveraging sparse convolutional neural networks, achieving high processing speeds and detection accuracy in well-structured, predictable scenarios. Nonetheless, VirConv's fixed projection and fusion stages reduce the model's flexibility, particularly in dynamically changing or semantically inconsistent environments. Its rigid design lacks the capability to adapt fusion parameters based on the evolving semantic relationships among sensor inputs, resulting in potential degradation of accuracy and robustness under complex or unstructured conditions. To overcome these limitations, our proposed FDSNet introduces an adaptive fusion approach guided by a FDS, dynamically selecting between mid-level and late stage fusion based on semantic consistency. This mitigates the performance degradation and computational overhead observed in traditional early fusion methods.



**Fig. 2.** Illustration of different sensor fusion strategies: Early, Mid, Late, and Hybrid fusion schemes.

### Mid-level fusion

Mid-level fusion is a modality integration strategy wherein sensor specific features are independently extracted prior to their fusion, preserving the distinct semantic and structural properties intrinsic to each modality. Unlike early fusion, mid-level fusion capitalizes on modality specific representations, enabling more effective fusion after extracting specialized features from each sensor independently. This approach is particularly advantageous in scenarios where direct integration at the raw data level may obscure critical domain specific information. **BEVFusion4D**<sup>33</sup> exemplifies this approach. BEVFusion4D first independently extracts BEV feature representations from LiDAR and camera inputs. Subsequently, it leverages a LiDAR-guided view transformation mechanism to accurately project and spatially align image features within a unified BEV coordinate system. The aligned features are then fused using spatial and temporal fusion modules designed to incorporate both spatial contextual coherence and temporal consistency. Although it efficiently resolves issues related to cross modal misalignment and effectively captures temporal dynamics, its fusion process operates unconditionally irrespective of the semantic coherence or agreement among the modalities. Consequently, noise or erroneous data from one modality may propagate into the fused representation, incurring unnecessary computational overhead and reducing model robustness. Similarly, **DeepStep**<sup>21</sup>, utilizes an incremental step wise fusion strategy. DeepStep progressively merges modality specific 2D image features and 3D LiDAR features via a hierarchical spatiotemporal transformer architecture. This step wise fusion paradigm enhances contextual reasoning and improves semantic understanding by gradually refining multi modal feature representations. Despite these benefits, it also exhibits limitations, particularly in its absence of conditional control mechanisms. It consistently applies identical fusion processes regardless of semantic disagreements, potentially degrading overall fusion effectiveness when certain modalities are noisy, inaccurate, or semantically inconsistent. In the context of Radar and Camera fusion, **CRN**<sup>34</sup> presents a mid-level approach that effectively integrates high level modality specific features through a dynamic spatial fusion strategy. By leveraging Radar's motion stability and Camera's semantic richness, CRN improves robustness against visual degradation and sparsity. However, similar to other mid-level methods, it lacks an explicit mechanism for selectively gating or adjusting the fusion process based on semantic agreement, which may limit its adaptability in inconsistent or noisy environments. Addressing these limitations, FDSNet employs the FDS metric to dynamically select the optimal fusion stage. By retaining mid-level fusion only when cross-modal alignment is strong and shifting to late fusion in cases of significant feature disagreement, FDSNet significantly reduces redundant computation and enhances robustness.

### Late fusion

Late fusion independently processes each sensor modality through separate, dedicated feature extraction and decision making pipelines, integrating their outputs only at the final decision making stage. This approach inherently maintains the modularity of processing streams and is particularly robust against scenarios involving sensor degradation or failure, as the fusion is performed at a high level decision or prediction space. Nevertheless, this strategy may fail to fully leverage intermediate semantic cues and complementary information available at earlier stages of processing, potentially limiting overall fusion performance. The **C-CLOCs** framework<sup>35</sup>, employs a contrastive learning based approach designed to align object level predictions from LiDAR and camera modalities. It accomplishes this by performing confidence calibration and Intersection-over-Union (IoU) based matching of object proposals generated independently by each modality. By aligning predictions post-hoc, it effectively reduces false positives and enhances consistency across sensor modalities. Despite these advantages, C-CLOCs is fundamentally constrained by its reliance on fixed, post-hoc alignment techniques. As a consequence, it lacks the flexibility to dynamically adapt its fusion strategy based on real time fluctuations or variations in modality specific performance, potentially compromising performance under rapidly changing environmental conditions or varying sensor reliability. Another model is **BAFusion**, introduced by Chen et al.<sup>17</sup>, utilizes a bidirectional attention mechanism to perform modality specific late stage fusion. It independently generates predictions from LiDAR and camera inputs, subsequently employing bidirectional cross attention modules to integrate these high level modality specific predictions. This approach effectively captures high level semantic correlations and retains modularity, enabling ease of sensor specific adaptation and maintenance. However, similar to other late fusion strategies, BAFusion applies fusion unconditionally, without explicitly evaluating or quantifying the level of cross modal semantic alignment or disagreement. Consequently, in challenging scenarios characterized by conflicting or semantically misaligned sensor outputs, unconditional fusion may degrade overall detection and classification accuracy. To effectively mitigate these limitations and exploit the inherent strengths of both mid-level and late stage fusion strategies, in contrast our proposed FDSNet dynamically quantifies the degree of semantic disagreement across modalities in real time. Unlike previous late fusion methods, FDSNet selectively applies late fusion only when significant disagreement exists, otherwise favoring computationally efficient mid-level fusion. Thus, it combines the robustness benefits of late fusion with adaptive computational efficiency.

### Dynamic and hybrid fusion

Recent advances in multi modal fusion for autonomous driving have prompted the emergence of dynamic and hybrid fusion methods, which integrate sensor data across multiple abstraction levels such as (point-level, mid-level, decision-level). These hybrid approaches aim to combine the strengths of individual fusion strategies, achieving a balance between robustness, flexibility, and computational efficiency by adaptively utilizing complementary information at varying processing stages. **DecoratingFusion**<sup>25</sup> exemplifies a hybrid approach, enriching LiDAR point clouds with image aligned features at the input stage and subsequently refining them using mid-level BEV-based cross modal attention. Similarly, **MS-Occ**<sup>26</sup> proposes a multi-stage LiDAR–Camera fusion architecture, combining mid-level projection of 2D image features into 3D voxel spaces with late stage semantic aggregation from multiple viewpoints. Although effective in improving geometric and semantic

understanding, these hybrid methods execute all fusion stages unconditionally, resulting in increased latency and computational costs even when simpler fusion strategies might suffice. **RCBEV**<sup>36</sup> further contributes to this direction by addressing the spatial misalignment between Radar and Camera modalities through modality specific feature adaptation and alignment modules, improving the reliability of Radar-Camera fusion in BEV-based 3D detection. Although effective in improving geometric and semantic understanding, these hybrid methods execute all fusion stages unconditionally, resulting in increased latency and computational costs even when simpler fusion strategies might suffice. In addition, they lack an explicit mechanism to evaluate or quantify semantic consistency across modalities, meaning that fusion is applied regardless of whether sensor features are aligned or conflicting. This inability to adapt often propagates redundant or noisy computations. Conditional fusion approaches, like **HydraFusion**<sup>37</sup>, offer a promising direction by selecting among predefined fusion branches based on scene context or learned contextual features. It employs a context-aware gating mechanism, dynamically choosing fusion paths guided by external metadata (e.g., weather) or learned context. However, each selected branch still follows a rigid fusion pipeline, performing fusion irrespective of real time semantic alignment, thus risking redundant computation and suboptimal performance.

To address these shortcomings comprehensively, the proposed FDSNet utilizes a real time computed FDS to dynamically select either mid-level or late fusion stages based explicitly on semantic consistency avoiding the redundant execution of all fusion branches and directly resolving the limitations of existing hybrid strategies. This conditional fusion approach further enhances computational efficiency, and maintains robust perception tailored explicitly for real time autonomous driving applications. Table 1 summarizes the reviewed sensor fusion approaches, categorized by fusion stage, and highlights their core limitations regarding adaptability, efficiency, and semantic consistency.

## Proposed work

The proposed FDSNet architecture dynamically integrates multimodal sensor data by conditionally selecting optimal fusion stage based on real-time semantic alignment across the three sensor configurations: Camera+LiDAR (C+L), Camera+Radar (C+R), and Camera+LiDAR+Radar (C+L+R). This algorithm operates in three primary steps: (1) Independent feature extraction and spatial alignment, wherein each sensor modality is processed independently and mapped into a unified Bird's Eye View (BEV) representation. (2) Computation of the Feature Disagreement Score (FDS), a variance-based metric quantifying semantic alignment and disagreement between modalities. and (3) Conditional fusion stage selection, dynamically choosing between mid-level feature fusion and late-stage decision fusion based on the global FDS value. This adaptive mechanism enhances computational efficiency and robustness for real-time autonomous driving applications. Early-stage raw feature fusion was intentionally excluded, as it tends to amplify noise, increase dimensionality, and propagate modality-specific errors, making it less effective for reliable real-time decision-making. The pseudocode describing the overall algorithm is provided in Algorithm 1.

Stage	Model	Strengths	Limitations
Early	MEFA <sup>31</sup>	High-resolution fusion, fine-grained spatial feature alignment	Rigid fusion, high computational cost, limited real-time use
	VirConv <sup>32</sup>	Lightweight, efficient voxel feature propagation	Limited adaptability, struggles with semantic inconsistencies
Mid-Level	BEVFusion4D <sup>33</sup>	Temporal BEV fusion, consistent scene aggregation	Executes fusion regardless of feature alignment, redundant computations
	CRN <sup>34</sup>	Radar-Camera feature fusion with dynamic spatial reasoning	Lacks semantic consistency gating, sensitive to modality disagreement
	DeepStep <sup>21</sup>	Progressive integration, improved temporal continuity	Unconditional fusion execution, limited adaptability
Late	C-CLOCS <sup>35</sup>	Modality-specific decision making, robustness to misalignment	Post-hoc alignment, limited semantic feature utilization
	BAFusion <sup>17</sup>	Uncertainty-aware fusion, reliability weighting	Unconditional fusion, limited adaptability to sensor disagreement
Hybrid	DecoratingFusion <sup>25</sup>	Multi-stage refinement, dense feature propagation	Executes all fusion stages unconditionally, increased latency
	MS-Occ <sup>26</sup>	Multi-scale occupancy reasoning, strong spatial coverage	Performs all fusion stages unconditionally, computationally intensive
	HydraFusion <sup>37</sup>	Flexible per-branch processing, adaptable fusion routes	Static fusion pipelines within each branch, unnecessary computations
	RCBEV <sup>36</sup>	Radar-Camera feature alignment, modality-specific spatial adaptation	Limited to Radar-Camera pairs, lacks generalized modality fusion

**Table 1.** Sensor fusion models with their strengths and limitations.

---

**Require:** Multimodal sensor inputs: Camera images, LiDAR, and RADAR point clouds.  
**Ensure:** Robust multimodal fusion output  $D_{final}(x, y)$

- 1: **Step 1: Independent Feature Extraction and BEV Projection**
- 2:  $F_{BEV}^{Cam}(c, x, y) \leftarrow$  Lift-Splat-Shoot(Camera Images)
- 3:  $F_{BEV}^{LiDAR}(c, x, y) \leftarrow$  PointPillars(LiDAR Points)
- 4:  $F_{BEV}^{Radar}(c, x, y) \leftarrow$  RadarBEVNet-based(Radar Points)
- 5: **Step 2: Feature Disagreement Score (FDS) Calculation**
- 6: **for** each spatial location  $(x, y)$  **do**
- 7:      $\mu(x, y) \leftarrow \frac{1}{M} \sum_{m=1}^M F_m(:, x, y)$
- 8:     **for** each modality  $m \in \{Cam, LiDAR, Radar\}$  **do**
- 9:          $\delta_m(x, y) \leftarrow F_m(:, x, y) - \mu(x, y)$
- 10:         Compute squared deviation:  $\|\delta_m(x, y)\|_2^2$
- 11:     **end for**
- 12:      $FDS_{var}(x, y) \leftarrow \frac{1}{M} \sum_{m=1}^M \|\delta_m(x, y)\|_2^2$
- 13: **end for**
- 14: Compute global FDS score:  $FDS_{global} \leftarrow \frac{1}{XY} \sum_{x=1}^X \sum_{y=1}^Y FDS_{var}(x, y)$
- 15: **Step 3: Conditional Fusion Stage Selection**
- 16: **if**  $FDS_{global} < \tau$  **then**
- 17:     Perform mid-level fusion:  $F_{fused}(x, y) \leftarrow \frac{1}{M} \sum_{m=1}^M F_m(:, x, y)$
- 18:     Predict detection results from  $F_{fused}(x, y)$
- 19: **else**
- 20:     **for** each modality  $m \in \{Cam, LiDAR, Radar\}$  **do**
- 21:         Generate modality-specific predictions  $D_m(x, y)$  and confidences  $\gamma_m(x, y)$
- 22:     **end for**
- 23:     Perform late fusion:  $D_{final}(x, y) \leftarrow \frac{\sum_{m=1}^M \gamma_m(x, y) D_m(x, y)}{\sum_{m=1}^M \gamma_m(x, y)}$
- 24: **end if**
- 25: **return**  $D_{final}(x, y)$

---

**Algorithm 1.** Proposed FDSNet fusion Algorithm.

### Feature extraction from multi-modal sensors

To extract semantically meaningful representations from heterogeneous sensor data, we adopt a three-branch design independently extracting mid-level features from Camera, LiDAR, and RADAR sensors. Each modality-specific representation is subsequently projected into a unified BEV representation to facilitate consistency assessment and spatial alignment. The use of BEV is motivated by its ability to provide compact, spatially coherent, and semantically rich representations—ideal for multi modal fusion. In contrast, voxel-based representations, which often memory intensive and computationally expensive due to their sparse 3D structure<sup>38</sup>. Image plane projections, meanwhile, suffer from geometric distortions and misalignment across modalities, as direct projection into camera viewpoints can warp spatial relationships and degrade 3D accuracy<sup>33</sup>. By projecting into BEV, we avoid these pitfalls while preserving both geometric precision and semantic density. Thus, BEV offers a robust fusion foundation compared to alternative fusion schemes.

#### Camera stream

To generate BEV features from multi-view camera images, we adopt the Lift-Splat-Shoot (LSS) mechanism<sup>39</sup>. Compared to transformer based view transformation methods like BEVFormer<sup>40</sup> or PETR<sup>41</sup>, which offer strong performance but come at a significantly higher computational cost, LSS offers a strong balance between accuracy and real time performance due to its simplified architecture and reduced latency. The process begins by the Lift step, in this step image features  $f \in \mathbb{R}^{C \times H \times W}$ , (where  $C$  is the number of feature channels,  $H$  and  $W$  are the spatial dimensions) are extracted using a backbone ResNet<sup>42</sup> and followed by FPN neck<sup>43</sup> module to enhance spatial detail and contextual representation. A depth classifier then predicts a discrete depth distribution  $\alpha \in \mathbb{R}^D$  for each pixel, where  $D$  is the number of sampled depth bins. Each pixel at image coordinates  $(u, v)$  is lifted into a set of 3D frustum points  $(x, y, z)$  using the camera intrinsic  $K$  and extrinsic  $[R \mid t]$ . Here,  $K \in \mathbb{R}^{3 \times 3}$  encodes the camera's internal parameters (focal lengths and principal point), while  $R \in \mathbb{R}^{3 \times 3}$  is a rotation matrix that orients the camera in 3D space, and  $t \in \mathbb{R}^{3 \times 1}$  is a translation vector specifying the camera's position in the world coordinate system. The full projection to 3D is given in (1), following the standard pinhole camera model and coordinate transformation defined in<sup>44</sup>:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R^\top \left( \alpha_d \cdot K^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} - t \right) \quad (1)$$

where  $\alpha_d$  scales each 3D point according to depth probability. In the Splat step, the lifted 3D points from all views are aggregated into a voxel grid  $V(c, x, y, z)$ . To obtain the final BEV feature map, a vertical pooling operation is applied along the height (z) axis while preserving the channel dimension:

$$F_{\text{BEV}}^{\text{Cam}}(c, x, y) = \sum_z V(c, x, y, z) \quad (2)$$

This BEV representation encodes both semantic and geometric information and serves as the unified spatial format for downstream fusion assessment. A visual illustration of the full camera-to-BEV transformation pipeline is provided in Fig. 3.

#### *LiDAR stream*

For the LiDAR We adopt the PointPillars framework<sup>45</sup> to efficiently convert LiDAR point clouds into BEV features. Compared to voxel-based encoders like VoxelNet<sup>46</sup> or sparse 3D convolutional neural network (CNN) models such as SECOND<sup>47</sup>, PointPillars eliminates the need for expensive 3D convolutions by operating in the 2D BEV plane, achieving a favorable trade-off between accuracy and speed as reported in the prior benchmarks<sup>48</sup>. The input point cloud  $P \in \mathbb{R}^{N \times D}$ , where  $N$  is the number of points and  $D$  includes spatial coordinates and intensity values, is first discretized into vertical columns (pillars) across the XY plane. Each non-empty pillar is encoded using PointNet<sup>49</sup> to produce a fixed length feature vector. These features are then scattered into a 2D pseudo-image, preserving spatial structure. Finally, the pseudo-image is processed by a 2D CNN to produce the final BEV feature map:

$$F_{\text{BEV}}^{\text{LiDAR}} \in \mathbb{R}^{C \times X \times Y} \quad (3)$$

By capturing local geometry and intensity, this representation facilitates spatial alignment essential for computing the FDS across sensor modalities.

#### *RADAR stream*

To efficiently extract BEV features from radar point clouds in real time, we employ a streamlined point-based backbone inspired by RadarBEVNet<sup>50</sup>. In contrast to alternative radar processing approaches such as heatmap-based representations<sup>34</sup>, which suffer from limited geometric precision and low spatial resolution, the point-based method provides a more spatially accurate and lightweight feature encoding strategy suitable for sparse radar data. The input data is represented as  $R \in \mathbb{R}^{K \times 5}$  where  $K$  is the number of radar points in a frame, and each point is represented by a 5-dimensional vector comprising its 3D spatial coordinates  $(x, y, z)$  and Doppler-compensated velocity components  $(v_x, v_y)$ . Each point is encoded using a simplified PointNet<sup>49</sup> architecture, which applies a shared Multi-Layer Perceptron (MLP) to project raw inputs into a higher dimensional feature space, followed by a global max pooling operation to capture contextual information. The radar feature encoding is given by:

$$f = \text{Concat}[\text{MLP}(f), \text{MaxPool}(\text{MLP}(f))] \quad (4)$$

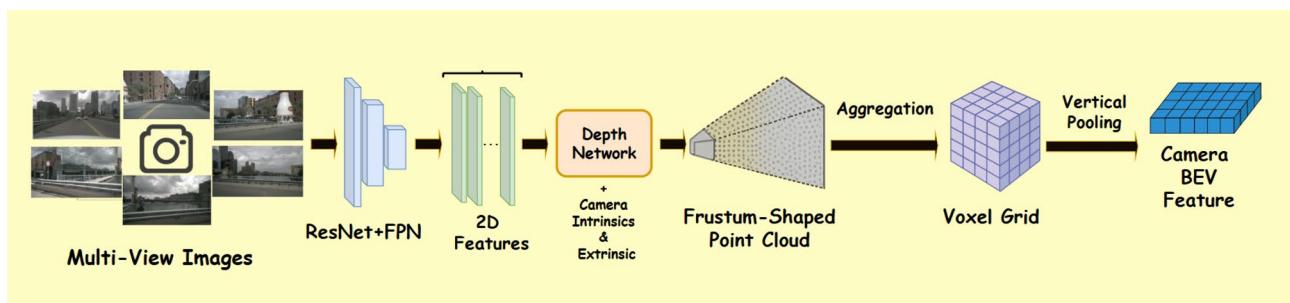
The encoded features are then scattered into a structured 2D grid using RCS-aware scattering, which spreads each point's influence over multiple BEV locations according to its Radar Cross Section (RCS). The resulting BEV feature map:

$$F_{\text{BEV}}^{\text{Radar}} \in \mathbb{R}^{C \times X \times Y} \quad (5)$$

serves as a spatially aligned representation used in downstream fusion and FDS computation.

#### **FDS calculation**

To quantify modality disagreement over spatially aligned BEV features, we adopt a variance-based FDS that captures semantic inconsistencies across the Camera, LiDAR, and Radar streams. Each modality-specific BEV feature map is denoted as  $F_m \in \mathbb{R}^{C \times X \times Y}$ , where  $F_m \in \{F_{\text{BEV}}^{\text{Cam}}, F_{\text{BEV}}^{\text{LiDAR}}, F_{\text{BEV}}^{\text{Radar}}\}$ , as introduced in



**Fig. 3.** BEV feature generation from multi-view camera inputs. 6 surround view images are first processed using a ResNet+FPN backbone to extract 2D image features. A depth network lifts each pixel into a 3D frustum using camera intrinsics and extrinsics. The lifted points are then aggregated into a voxel grid and vertically pooled to produce the final BEV representation.

Equations (2), (3), and (5), respectively. From each BEV map, the local feature value at spatial coordinate  $(x, y)$  and channel  $c$  for modality  $m$  is denoted as  $f_m^{(c)}(x, y) \in \mathbb{R}$ . The average channel-wise feature at each location is computed using the following formulation:

$$\mu^{(c)}(x, y) = \frac{1}{M} \sum_{m=1}^M f_m^{(c)}(x, y), \quad \text{for } c = 1, \dots, C \quad (6)$$

Here,  $M \in \{1, 2, 3\}$  represents the number of modalities (Camera, LiDAR, Radar) participating in the current fusion instance. Next, we determine each modality deviation from the mean feature vector calculated in Eq. (7) as:

$$\delta_m^{(c)}(x, y) = f_m^{(c)}(x, y) - \mu^{(c)}(x, y), \quad \text{for } c = 1, \dots, C \quad (7)$$

The level of disagreement per modality at each location is then computed by the squared L2-norm of these deviations, as defined in Eq. (8):

$$\|\delta_m(x, y)\|_2^2 = \sum_{c=1}^C (\delta_m^{(c)}(x, y))^2 \quad (8)$$

This formulation quantifies the extent to which modality  $m$  deviates from the average representation across all modalities at a specific spatial location. A higher value indicates greater disagreement or inconsistency in the feature encoding of that modality relative to the others. Finally, the overall **FDS** at location  $(x, y)$  is computed by averaging the squared deviations derived in Eq. (8) across all sensor modalities, as defined in Eq. (9):

$$\text{FDS}_{\text{var}}(x, y) = \frac{1}{M} \sum_{m=1}^M \|\delta_m(x, y)\|_2^2 \quad (9)$$

This metric quantifies the consistency of feature representations across sensors at each spatial location and enables downstream tasks to identify ambiguous or uncertain regions. To extend this to the scene level, we define **Global Feature Disagreement Score** ( $\text{FDS}_{\text{global}}$ ) by spatially averaging local values across the entire BEV grid:

$$\text{FDS}_{\text{global}} = \frac{1}{X \cdot Y} \sum_{x=1}^X \sum_{y=1}^Y \text{FDS}_{\text{var}}(x, y) \quad (10)$$

This scalar reflects the overall semantic consistency across all modalities. Local noise or isolated disagreements are smoothed, while systematic cross-modal misalignments remain emphasized, making global FDS a robust metric for guiding adaptive fusion stage selection.

### Conditional fusion stage selection

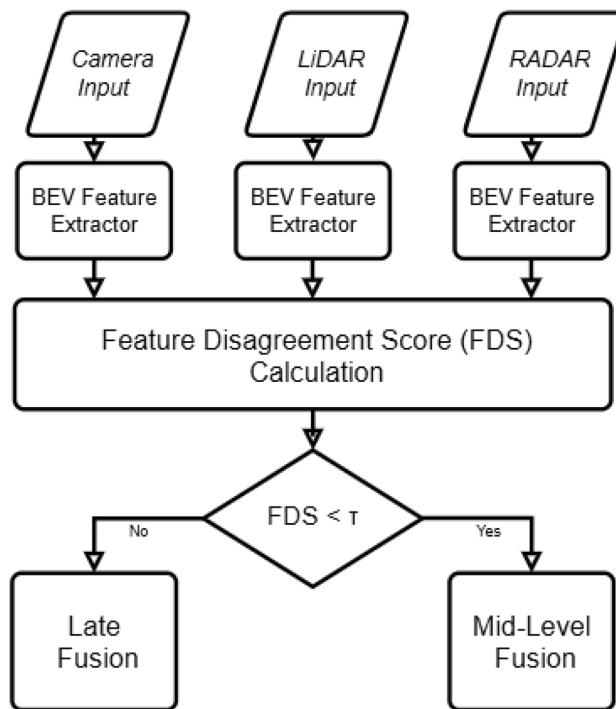
Building upon the calculated  $\text{FDS}_{\text{global}}$ , we propose a dynamic fusion control strategy that conditionally selects between mid-fusion and late fusion stages. This approach leverages the global agreement across spatially aligned BEV features to determine whether to fuse at the feature level for confident, consistent scenes or defer to decision-level fusion for ambiguous, uncertain regions. A low  $\text{FDS}_{\text{global}}$  indicates strong modal agreement, while a high value suggests potential semantic misalignment or sensor disagreement. We then apply a conditional rule to select the fusion stage  $S$  based on a experimentally tunable threshold  $\tau$  as:

$$S = \begin{cases} \mathcal{F}_{\text{mid}}, & \text{if } \text{FDS}_{\text{global}} < \tau \\ \mathcal{F}_{\text{late}}, & \text{otherwise} \end{cases} \quad (11)$$

The overall conditional switching process is illustrated in Fig. 4, where BEV features from Camera, LiDAR, and Radar are evaluated through the FDS and compared against a threshold  $\tau$  to select the appropriate fusion stage. In the mid-fusion stage, denoted by  $\mathcal{F}_{\text{mid}}$ , spatially aligned BEV features from all available modalities are combined through an element-wise average:

$$F_{\text{fused}}(x, y) = \frac{1}{M} \sum_{m=1}^M f_m(x, y) \quad (12)$$

This simple yet effective fusion strategy assumes equal trust across modalities in regions where the FDS indicates strong semantic consistency. Compared to more complex fusion mechanisms such as attention-based fusion, or convolutional encoders as used in BEVFusion<sup>51</sup>, this approach introduces minimal computational overhead and is highly suitable for real time deployment. By avoiding additional parameters  $\mathcal{F}_{\text{mid}}$  provides a fast, deterministic alternative while still benefiting from the redundancy and complementarity of multi-modal inputs under low disagreement conditions. A detailed comparison of this method against alternative mid-fusion strategies is provided in Table 2.



**Fig. 4.** Flowchart of FDSNet’s conditional fusion, where the FDS threshold  $\tau$  determines mid-level or late fusion selection.

Fusion method	Fusion operation	Complexity	Real-time suitability	Parameter overhead
Element-wise Average <sup>52</sup>	Sum over spatially aligned BEV features	Low	High	Low
Attention-based Fusion <sup>34</sup>	Local and global attention over concatenated features	High	Low	High
Convolutional Fusion <sup>51</sup>	Concatenation followed by convolutional encoding in BEV space	Medium	Medium	Medium

**Table 2.** Comparison of mid-level fusion strategies.

In contrast to mid-level fusion, late fusion operates at the decision level, where each modality: Camera, LiDAR, and Radar independently generates complete detection outputs, including class probability scores and corresponding bounding box parameters at each BEV location  $(x, y)$ . Each modality  $m$  provides a prediction  $D_m(x, y)$  alongside a confidence estimate  $\gamma_m(x, y) \in [0, 1]$ , indicating its reliability at that position. The final fused decision  $D_{\text{final}}(x, y)$  is computed using a confidence-weighted aggregation:

$$D_{\text{final}}(x, y) = \frac{\sum_{m=1}^M \gamma_m(x, y) \cdot D_m(x, y)}{\sum_{m=1}^M \gamma_m(x, y)} \quad (13)$$

this approach ensures that predictions from more reliable modalities dominate the final outcome, while those from less certain sources are down weighted. Compared to simple averaging or rule based voting, the confidence weighted fusion mechanism offers a more adaptive and fine grained strategy, dynamically reflecting sensor reliability at each spatial location. This makes it particularly effective under adverse conditions, such as sensor occlusion, misalignment, or environmental interference, thereby enhancing the robustness and reliability of multi-sensor object detection in real time driving environments. A detailed comparison of this method with other representative late-stage fusion approaches is presented in Table 3. In addition, for clarity, we provide a summary table of the key terms and symbols used in FDSNet, which consolidates the mathematical notations introduced in Table 4.

## Experiments

In this section, we begin by presenting the benchmark dataset employed to evaluate the proposed FDSNet. We then describe the experimental settings and implementation details.

### Dataset

We conduct our experiments on the nuScenes dataset<sup>30</sup>, a large scale benchmark for autonomous driving perception tasks. It provides synchronized multi-sensor data, including six surround view cameras, five

Fusion method	Fusion operation	Complexity	Real-time suitability	Parameter overhead
Confidence-Weighted Fusion <sup>35</sup>	Weighted sum of predictions using modality-wise confidence scores	Medium	High	Low
Simple Averaging <sup>53</sup>	Uniform average of outputs without confidence weights	Low	Medium	None
Rule-Based Voting <sup>54</sup>	Majority or priority-based fusion of class labels or boxes	Low	Medium	None
Learned Fusion Gate <sup>55</sup>	Task-specific gating network trained to select or weight predictions from each modality	High	Low	High

**Table 3.** Comparison of late fusion strategies.

Symbol/term	Definition	Role in FDSNet
$F_m \in \mathbb{R}^{C \times X \times Y}$	Modality-specific BEV feature map (Camera, LiDAR, Radar)	Input feature representation per modality
$f_m^{(c)}(x, y)$	Feature value at channel $c$ , location $(x, y)$ , modality $m$	Local spatial feature element
$\mu^{(c)}(x, y)$	Mean feature across modalities at $(x, y)$ , Eq. (6)	Baseline for measuring disagreement
$\delta_m^{(c)}(x, y)$	Deviation of modality $m$ from mean at $(x, y)$ , Eq. (7)	Captures modality-specific differences
$\ \delta_m(x, y)\ _2^2$	Squared L2-norm of deviations, Eq. (8)	Quantifies per-modality disagreement
$FDS_{\text{var}}(x, y)$	Average disagreement across modalities at $(x, y)$ , Eq. (9)	Local Feature Disagreement Score
$FDS_{\text{global}}$	Spatial average of $FDS_{\text{var}}(x, y)$ , Eq. (10)	Global consistency metric for fusion stage decision
$\tau$	Threshold parameter, Eq. (11)	Controls mid- vs late-fusion switching
$F_{\text{fused}}$	Mid-level fusion output, Eq. (12)	Element-wise average when agreement is high
$D_m(x, y), \gamma_m(x, y)$	Detection output and confidence for modality $m$ , Eq. (13)	Inputs for late fusion
$D_{\text{final}}(x, y)$	Final decision after confidence-weighted fusion, Eq. (13)	Robust fused output under uncertainty

**Table 4.** Summary of symbols and terms in FDSNet.

RADAR sensors, and one 32-beam LiDAR scanner, captured at 2 Hz across diverse urban scenarios in Boston and Singapore. Each scene offers full 360° coverage and is annotated with 3D bounding boxes for 10 object categories, including cars, pedestrians, trucks, and bicycles. A representative samples from the nuScenes dataset are provided in Fig. 5, including RGB images from six camera views (top two rows) and a LiDAR top view (bottom), to visualize the spatial coverage and complementary characteristics of these two sensor modalities.

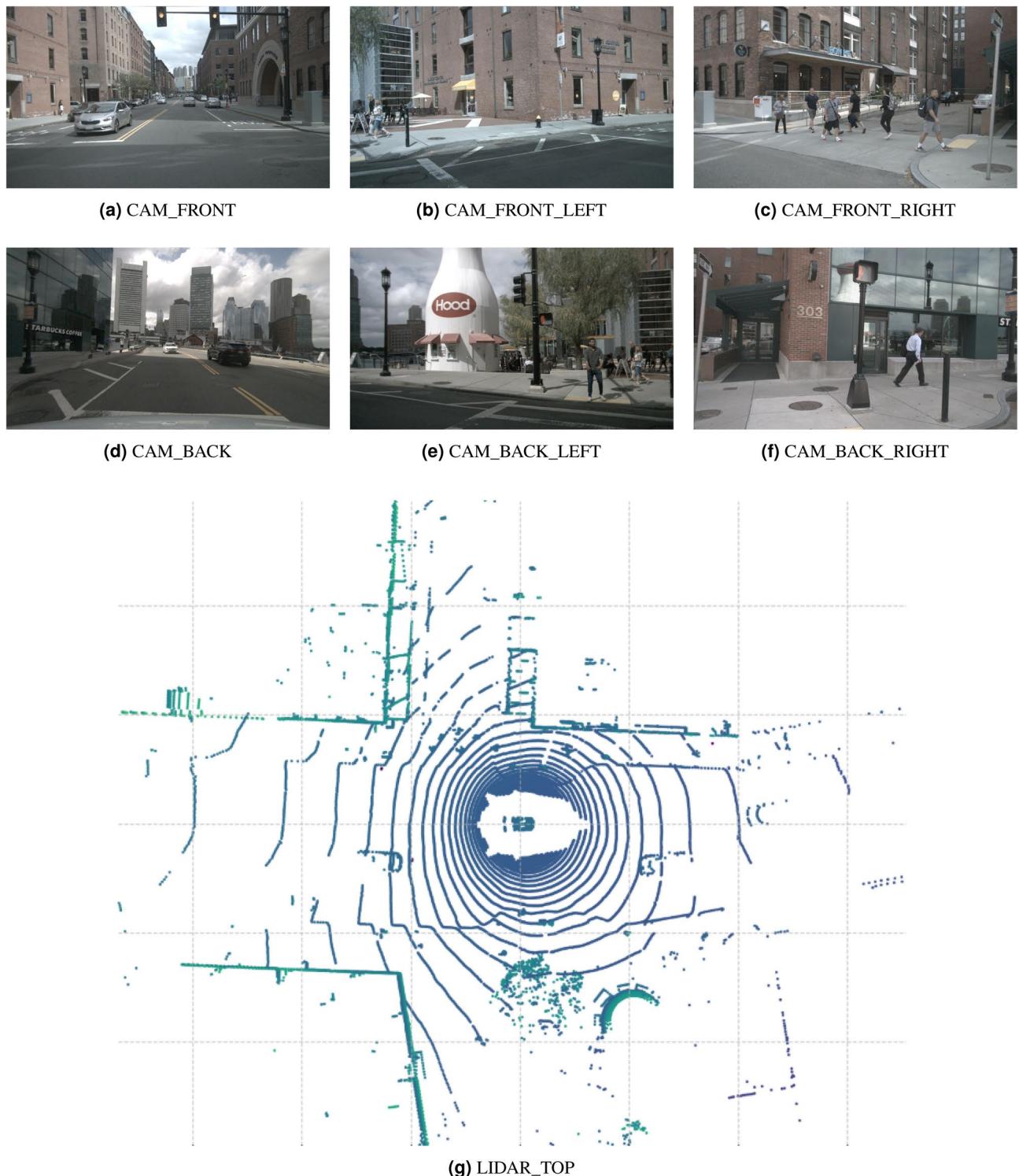
## Experimental setup and implementation details

### Computational environment

All experiments were performed on a high performance computing workstation equipped with an Intel Core i7 CPU running at 3.40 GHz, 64 GB of system memory, and two NVIDIA GeForce GTX 1080 Ti GPU. The proposed framework was implemented in PyTorch<sup>56</sup> using open-sourced MMDetection3D<sup>57</sup>, which provides modular support for 3D object detection and multi-sensor fusion tasks.

### Implementation details

The proposed FDSNet includes three modality specific backbone networks, a ResNet<sup>42</sup> followed by FPN neck<sup>43</sup> module for camera input initialized from ImageNet<sup>58</sup> pretrained weights, a PointPillars backbone for LiDAR (trained from scratch following Lang et al.<sup>45</sup>), and a streamlined RadarBEVNet inspired point-based backbone for RADAR data (trained from scratch with He initialization as in Zhou et al.<sup>50</sup>). Training was structured into three sequential stages to ensure robust feature learning and stable cross-modal convergence: First, the LiDAR backbone was trained individually for 20 epochs to extract robust 3D representations from point clouds. Second, the camera backbone was independently trained for 12 epochs utilizing standard augmentations (random flipping with 0.5 probability, rotation  $\pm 0.3925$  rad, and scaling [0.95, 1.05]). Finally, the multi-sensor fusion network was trained jointly for an additional 18 epochs, integrating Camera, LiDAR, and RADAR features into a unified BEV representation, with parameters from camera and LiDAR backbones frozen as recommended by prior fusion studies<sup>59,60</sup>. For evaluation, we split the nuScenes dataset, which comprises roughly 1000 scenes, into 700 for training, 150 for validation, and 150 for testing, ensuring no overlap across subsets, following the official nuScenes<sup>30</sup>. All training stages employed the same optimization setup. AdamW<sup>61</sup> with an initial learning rate of  $1 \times 10^{-3}$ , weight decay of 0.01, gradient clipping at norm 35, linear warm-up (1000 iterations), and multi-step decay at epochs 14 and 17 by a factor of 0.1, with a batch size of 4 samples per GPU (on 2 GPUs). Point cloud data from LiDAR and RADAR was voxelized into pillars of size  $0.25 \text{ m} \times 0.25 \text{ m} \times 8 \text{ m}$ , covering  $[-50 \text{ m}, 50 \text{ m}]$  in the X/Y axes and  $[-5 \text{ m}, 3 \text{ m}]$  in the Z-axis. For the threshold parameter  $\tau$ , we experimentally evaluated values in the range [0.3–0.7] on the validation set and selected  $\tau = 0.5$  as the default setting, since it provided the best balance between mid- and late-stage fusion. This regimen ensured effective modality specific learning, enhanced robustness, computational efficiency, and reproducibility.



**Fig. 5.** Representative samples from the nuScenes Dataset<sup>30</sup>.

## Evaluation metrics

To rigorously quantify the performance of the proposed FDSNet for 3D object detection, we adopt the official nuScenes detection evaluation protocol<sup>[30]</sup>, which offers a multifaceted assessment across spatial localization, object orientation, scale estimation, velocity prediction, and attribute classification. The evaluation focuses on a comprehensive set of metrics, including the *mean Average Precision* (mAP), *mean Average Translation Error* (mATE), *mean Average Scale Error* (mASE), *mean Average Orientation Error* (mAOE), *mean Average Velocity Error* (mAVE), *mean Average Attribute Error* (mAAE), and the overall *nuScenes Detection Score* (NDS). These

metrics collectively provide a holistic view of a model's capability to accurately detect and characterize objects in 3D space, reflecting not only detection precision but also geometric fidelity and semantic richness.

### Mean average precision (mAP)

In alignment with the nuScenes evaluation protocol, mean Average Precision is computed using a center-distance-based matching approach rather than the traditional IoU, reducing sensitivity to object size and orientation. A detection is considered a true positive if its 2D center lies within a threshold distance  $d \in D = \{0.5, 1, 2, 4\}$  meters from the ground-truth box center. For each class  $c \in C$  and each threshold  $d \in D$ , the Average Precision  $AP_{c,d}$  is defined as the area under the precision-recall curve, excluding operating points where precision or recall falls below 10%. The final mean Average Precision (mAP) is then computed as the average across all classes and thresholds:

$$mAP = \frac{1}{|C||D|} \sum_{c \in C} \sum_{d \in D} AP_{c,d} \quad (14)$$

### Mean average scale error (mASE)

The mean Average Scale Error (mASE) quantifies the deviation in object dimensions between predicted and ground truth 3D bounding boxes. Unlike traditional *IoU* metrics, mASE isolates scale discrepancies by first aligning the predicted box with the ground truth in terms of translation and orientation, so that only scale contributes to the residual error. It is formally defined as:

$$ASE = 1 - IoU(\hat{B}_{\text{pred}}, B_{\text{gt}}) \quad (15)$$

Where  $\hat{B}_{\text{pred}}$  denotes the rescaled predicted box aligned with the ground truth box  $B_{\text{gt}}$  in both position and yaw orientation.

### Mean average translation error (mATE)

This metric measures the average Euclidean distance (in meters) between predicted and ground truth bounding box centers:

$$mATE = \frac{1}{N} \sum_{i=1}^N \|t_i^{\text{pred}} - t_i^{\text{gt}}\|_2 \quad (16)$$

### Mean average orientation error (mAOE)

The mean Average Orientation Error (mAOE) quantifies the accuracy of predicted object orientations by measuring the smallest angular difference between the predicted yaw angle  $\theta_{\text{pred}}$  and the ground truth yaw angle  $\theta_{\text{gt}}$ , expressed in radians. This error is computed over a full  $360^\circ$  period for most object classes, except for symmetric objects such as barriers, where a  $180^\circ$  period is used. The orientation error for each matched prediction is defined as:

$$AOE = \min(|\theta_{\text{pred}} - \theta_{\text{gt}}|, 2\pi - |\theta_{\text{pred}} - \theta_{\text{gt}}|) \quad (17)$$

The final mAOE is obtained by averaging the AOE values across all valid classes and matched detections, providing a robust measure of directional estimation performance that is independent of translation or scale.

### Mean average velocity error (mAVE)

The mean Average Velocity Error (mAVE) evaluates the accuracy of predicted object motion by calculating the L2 norm (Euclidean distance) between the predicted and ground truth velocity vectors in the 2D plane, expressed in meters per second (m/s). This metric captures discrepancies in both magnitude and direction of motion and is defined for each matched detection as:

$$AVE = \|v_{\text{pred}} - v_{\text{gt}}\|_2 \quad (18)$$

where  $v_{\text{pred}}$  and  $v_{\text{gt}}$  represent the predicted and ground truth velocity vectors, respectively. The final mAVE is computed by averaging AVE values across all matched predictions and valid object classes that exhibit motion.

### Mean average attribute error (mAAE)

The mean Average Attribute Error (mAAE) quantifies the accuracy of attribute prediction by computing the complement of classification accuracy. Specifically, the per sample attribute error is defined as:

$$AAE = 1 - acc \quad (19)$$

where **acc** denotes the proportion of correctly predicted attributes over the total number of valid attribute annotations. The final mAAE is obtained by averaging AAE values across all matched detections and applicable object classes (excluding classes like cones or barriers where attributes are undefined).

### nuScenes detection score (NDS)

The nuScenes Detection Score (NDS) provides a unified metric to evaluate both detection accuracy and the quality of 3D bounding box estimation. It combines the mAP with five True Positive (TP) error metrics translation (mATE), scale (mASE), orientation (mAOE), velocity (mAVE), and attribute classification (mAAE) into a single score. These TP metrics, denoted collectively as mTP, capture the estimation fidelity of critical object properties. The NDS is computed as:

$$\text{NDS} = \frac{1}{10} \left( 5 \cdot \text{mAP} + \sum_{\text{mTP}} (1 - \min(1, \text{mTP})) \right) \quad (20)$$

Here, the summation over mTP refers to the set  $\{\text{mATE}, \text{mASE}, \text{mAOE}, \text{mAVE}, \text{mAAE}\}$ . Each metric is clipped to the  $[0, 1]$  range to ensure stability and comparability. The final score balances classification performance (via mAP) with the regression accuracy of object properties.

## Results and discussion

All experiments were conducted on the nuScenes benchmark<sup>30</sup>, where we evaluated our proposed FDSNet framework under three sensor fusion configurations: Camera+RADAR (C+R), Camera+LiDAR (C+L), and Camera+LiDAR+RADAR (C+L+R). As summarized in Table 5 and Table 6, the complete fusion configuration (C+L+R) achieved the highest performance across both the validation set (75.9% mAP, 78.1% NDS) and test set (76.1% mAP, 78.2% NDS), outperforming recent state-of-the-art multi-modal detectors such as PolarFusion and IS-Fusion. This superior performance arises from the complementary strengths of the three modalities. The Camera provides dense semantic and texture information, LiDAR contributes precise geometric depth and structural cues, while Radar offers robustness under adverse conditions such as rain, fog, or poor illumination, together with FDSNet's adaptive fusion strategy, which aligns the fusion stage with the level of cross-modal consistency. The (C+L) configuration also showed competitive performance (74.4% mAP, 76.5% NDS on validation, 74.9% mAP, 76.8% NDS on test), exceeding PolarFusion and IS-Fusion by noticeable margins and reinforcing the effectiveness of integrating high-resolution geometry with visual context. Despite RADAR's sparsity and noisier returns, the (C+R) setting yielded promising results (56.7% mAP, 64.4% NDS on test), outperforming top radar fusion baselines such as RCBEVDet and CRN. This indicates that even in sparse modalities, our adaptive fusion mechanism effectively suppresses cross-modal inconsistencies and enhances robustness. Qualitative results are shown in Fig. 6. A detailed breakdown of mAP and NDS under each fusion configuration is illustrated in Fig. 7.

## Ablation and efficiency analysis

To comprehensively evaluate the adaptability and efficiency of FDSNet, we conducted an extensive ablation study covering four aspects: (1) the effect of the threshold parameter  $\tau$  on adaptive stage selection, (2) the impact of fusion strategy and dynamic switching, (3) computational efficiency across different sensor configurations, and (4) category-wise detection performance. Together, these analyses validate the effectiveness of the proposed FDS in guiding real-time fusion decisions, improving perception accuracy under varying sensor agreements, and maintaining consistent detection quality across object classes.

Method	Input	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
CenterFusion <sup>62</sup>	C+R	45.3	33.2	0.649	0.263	0.535	0.540	0.142
CRAFT <sup>63</sup>	C+R	51.7	41.1	0.494	0.276	0.454	0.486	0.176
RCBEVDet <sup>16</sup>	C+R	56.3	45.3	0.492	0.269	0.449	0.230	0.188
RCBEV4D <sup>36</sup>	C+R	49.7	38.1	0.526	0.272	0.445	0.465	0.185
CRN <sup>34</sup>	C+R	54.3	44.8	0.518	0.283	0.552	0.279	0.180
CR3DT <sup>64</sup>	C+R	45.6	35.1	-	-	-	0.47	-
BEVDet <sup>65</sup>	C	39.2	31.2	0.691	0.272	0.523	0.909	0.247
BEVDepth <sup>66</sup>	C	47.5	35.1	0.639	0.267	0.479	0.428	0.198
SOLOFusion <sup>67</sup>	C	53.4	42.7	0.567	0.274	0.411	0.252	0.188
StreamPETR <sup>68</sup>	C	54.0	43.2	0.581	0.272	0.413	0.295	0.195
RCBEVDet <sup>16</sup>	C+R	56.8	45.3	0.486	0.285	0.404	0.220	0.192
PolarFusion <sup>69</sup>	C+L	75.1	73.3	-	-	-	-	-
IS-Fusion <sup>70</sup>	C+L	74.0	72.8	-	-	-	-	-
ProFusion3D <sup>71</sup>	C+L	73.6	71.1	-	-	-	-	-
FDSNet (Ours)	C+R	58.2	47.9	0.468	0.251	0.319	0.270	0.140
FDSNet (Ours)	C+L	76.5	74.4	0.398	0.228	0.288	0.240	0.110
FDSNet (Ours)	C+L+R	<b>78.1</b>	<b>75.9</b>	<b>0.385</b>	<b>0.219</b>	<b>0.275</b>	<b>0.229</b>	<b>0.105</b>

**Table 5.** Quantitative comparison of 3D object detection performance on the nuScenes validation set. 'C', 'R', and 'L' denote input from Camera, Radar, and LiDAR sensors, respectively.

Method	Input	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
KPConvPillars <sup>72</sup>	R	13.9	4.9	0.823	0.428	0.607	2.081	1.000
CenterFusion <sup>62</sup>	C+R	44.9	32.6	0.631	0.261	0.516	0.614	0.115
RCBEV <sup>36</sup>	C+R	48.6	40.6	0.484	0.257	0.587	0.702	0.140
MVFusion <sup>73</sup>	C+R	51.7	45.3	0.569	0.246	0.379	0.781	0.128
CRAFT <sup>63</sup>	C+R	52.3	41.1	0.467	0.268	0.456	0.519	0.114
BEVFormer <sup>40</sup>	C	56.9	48.1	0.582	0.256	0.375	0.378	0.126
PETRv2 <sup>41</sup>	C	58.2	49.0	0.561	0.243	0.361	0.343	0.120
BEVDepth <sup>66</sup>	C	60.5	51.5	0.446	0.242	0.377	0.324	0.135
SOLOFusion <sup>67</sup>	C	61.9	54.0	0.453	0.257	0.376	0.276	0.148
CRN <sup>34</sup>	C+R	62.4	57.5	0.416	0.264	0.456	0.365	0.130
SparseBEV <sup>74</sup>	C	63.6	55.6	0.485	0.244	0.332	0.246	0.117
StreamPETR <sup>68</sup>	C	63.6	55.0	0.493	0.241	0.343	0.243	0.123
RCBEVDet <sup>16</sup>	C+R	63.9	55.0	0.390	0.234	0.362	0.259	0.113
PolarFusion <sup>69</sup>	C+L	76.1	74.5	-	-	-	-	-
IS-Fusion <sup>70</sup>	C+L	75.2	73.0	-	-	-	-	-
FDSNet (Ours)	C+R	64.4	56.7	0.402	0.236	0.341	0.248	0.114
FDSNet (Ours)	C+L	76.8	74.9	0.384	0.227	0.323	0.233	0.108
FDSNet (Ours)	C+L+R	78.2	76.1	0.371	0.222	0.315	0.225	0.105

**Table 6.** Quantitative comparison of 3D object detection performance on the nuScenes test set. ‘C’, ‘R’ and ‘L’ denote input from Camera, Radar, and LiDAR sensors, respectively.

### Effect of threshold $\tau$

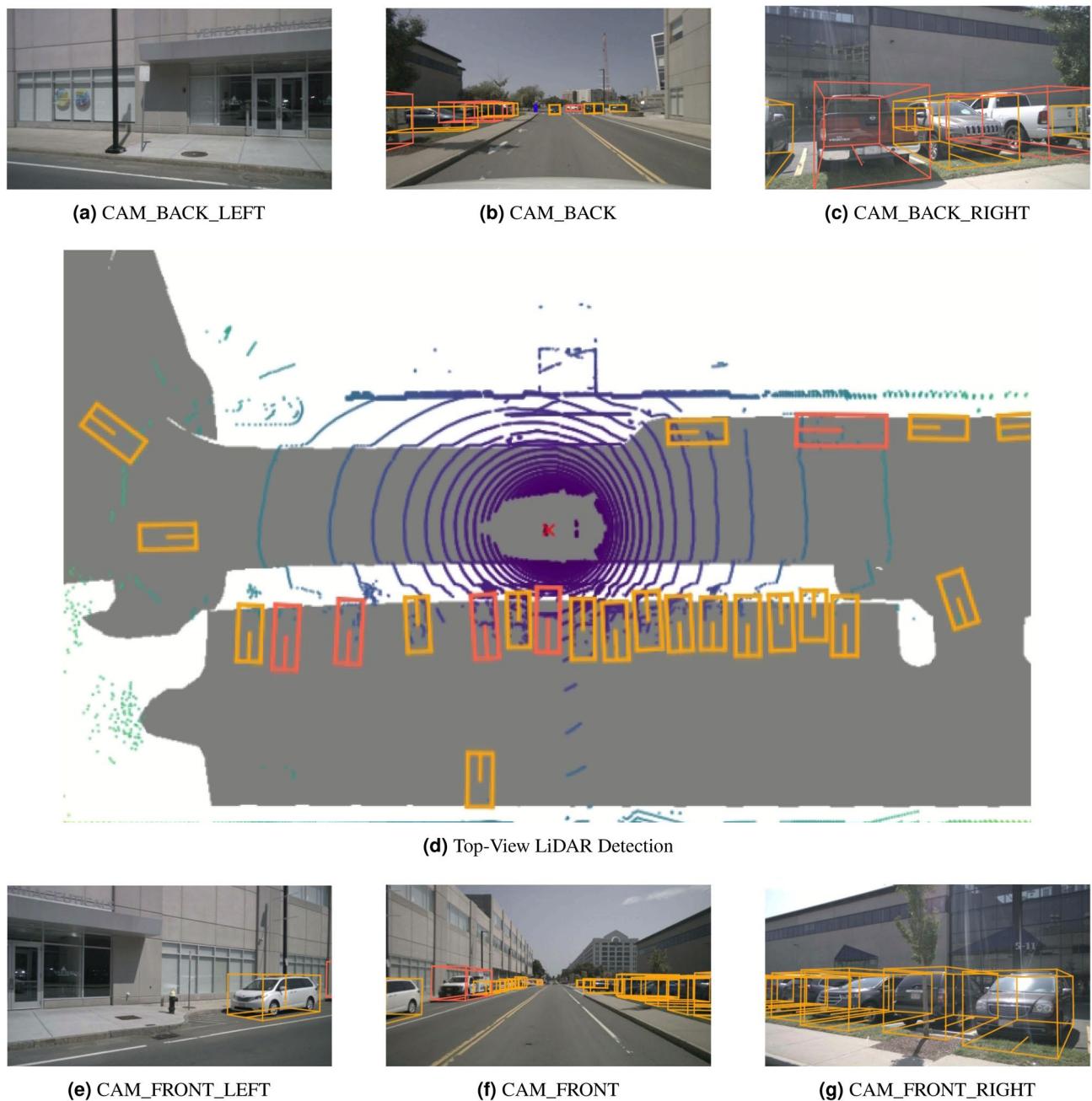
We investigate the influence of the threshold parameter  $\tau$ , which governs the balance between mid and late stage fusion across configurations in Table 7. As expected, a lower  $\tau$  favors late fusion by making it harder to satisfy the condition  $FDS_{\text{global}} < \tau$ , while a higher  $\tau$  biases toward mid fusion. For example,  $\tau = 0.3$  leads to late fusion dominance, while  $\tau = 0.7$  shifts the preference to mid fusion, with  $\tau = 0.5$  providing the best trade-off. The performance variation associated with threshold tuning is visualized in Fig. 8, which plots the effect of  $\tau$  on both mAP and NDS across all sensor configurations. Together, these results validate that FDSNet’s dynamic fusion mechanism adapts effectively to sensor agreement levels and consistently improves performance while preserving computational efficiency.

### Ablation on fusion strategy

The ablation results in Table 8 show that incorporating dynamic switching via the Feature Disagreement Score (FDS) yields consistent improvements across all evaluation metrics compared with static fusion strategies using all three modalities (Camera, LiDAR, and Radar). In the fixed mid-fusion setup, features from the three sensors are merged before detection, effectively combining the semantic richness of the camera with the geometric precision of LiDAR and the motion awareness of radar. However, this approach assumes perfect spatial alignment between modalities, but in practice, small calibration drift or the inherent sparsity of radar measurements can lead to feature misalignment, which in turn degrades orientation and velocity estimation (mAOE = 0.289, mAVE = 0.252), limiting overall accuracy (mAP = 73.2, NDS = 75.4). The fixed late-fusion configuration, which merges independent detections at the decision level, enhances robustness to sensor noise through confidence weighting. LiDAR primarily governs localization accuracy (mATE = 0.395) while Radar refines motion estimation (mAVE = 0.238), but lacks intermediate feature interaction, resulting in suboptimal semantic integration (mAP = 74.3, NDS = 76.5). In contrast, dynamic FDSNet adaptively switches between mid and late fusion according to real-time semantic consistency, using mid-fusion under strong cross-modal agreement and late-fusion when discrepancies arise. This mechanism improves detection across all metrics (mAP = 75.9, NDS = 78.1, mATE = 0.385, mASE = 0.219, mAOE = 0.275, mAVE = 0.229, mAAE = 0.105) by optimally combining the complementary strengths of camera semantics, LiDAR geometry, and radar motion cues, ensuring robust and consistent multimodal perception for autonomous driving.

### Computational efficiency analysis

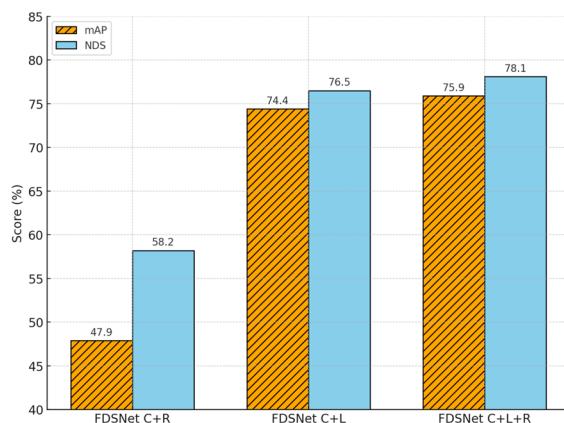
To evaluate the scalability and runtime efficiency of the proposed framework, we analyzed FDSNet’s computational complexity, throughput, and detection accuracy under the three sensor configurations: (C+R), (C+L), and (C+L+R) setup. As summarized in Table 9, FDSNet exhibits a consistent and interpretable trade-off between perception accuracy and computational cost across all modality combinations. The lightweight C+R configuration achieves robust performance (mAP = 47.9, NDS = 58.2) with the lowest computational load (318 GFLOPs, 6.2 GB) and the highest throughput (31 FPS), making it suitable for resource-constrained deployments. The C+L configuration offers a strong balance between precision and efficiency (mAP = 74.4, NDS = 76.5) with moderate requirements (365 GFLOPs, 7.0 GB, 27 FPS). Extending to the full C+L+R setup yields the highest detection accuracy (mAP = 75.9, NDS = 78.1) with only a modest increase in complexity (412 GFLOPs, 7.8 GB) and an effective runtime of 24 FPS.



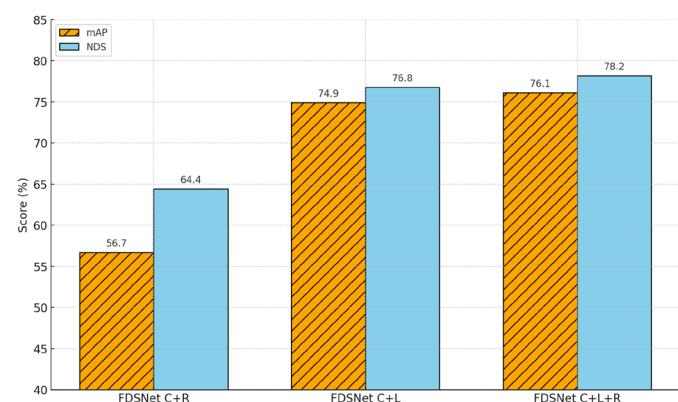
**Fig. 6.** Detection results from our proposed **FDSNet** model. The fused multi-camera, LiDAR, and Radar views showcase the effectiveness of our adaptive fusion strategy for robust 3D object detection under diverse perspectives.

#### Class-level performance analysis

To provide a deeper understanding of category-level detection behavior, Table 10 presents the per-class mAP and NDS of FDSNet across different sensor configurations on the nuScenes validation set. The lightweight (C+R) configuration achieves moderate accuracy overall but exhibits reduced precision for small targets such as pedestrian and bicycle, where the sparse radar returns and limited spatial priors constrain feature alignment. Incorporating (C+L) markedly enhances structural reasoning and object boundary localization through dense geometric cues, resulting in substantial improvements across all categories, particularly for large objects truck, bus and barrier. Extending to the full configuration (C+L+R) yields the highest per-class performance by leveraging radar-derived velocity fields to disambiguate motion states and refine temporal consistency in dynamic scenarios. These results confirm that FDSNet's adaptive fusion mechanism scales effectively with sensing diversity, integrating complementary modality characteristics to achieve balanced and robust 3D perception across all object classes.



**(a)** Validation set results for FDSNet under different fusion configurations.

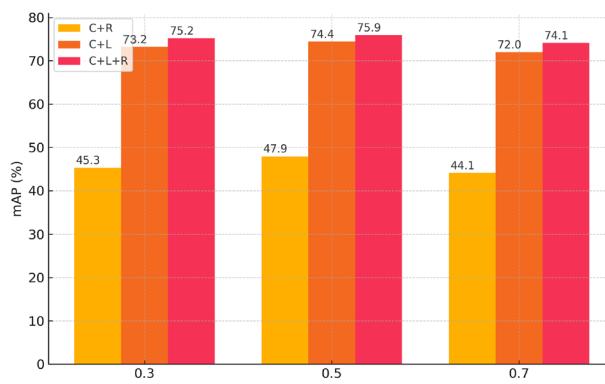


**(b)** Test set results for FDSNet under different fusion configurations.

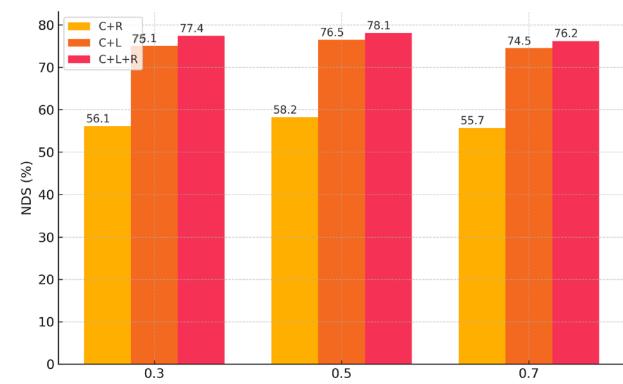
**Fig. 7.** Performance comparison of mAP and NDS for different sensor fusion configurations using FDSNet on the nuScenes dataset.

Threshold $\tau$	Sensor configuration	Fusion stage bias	$mAP \uparrow$	$NDS \uparrow$
0.3	C + R	Late-Fusion Dominant	45.3	56.1
0.5	C + R	Balanced	<b>47.9</b>	<b>58.2</b>
0.7	C + R	Mid-Fusion Dominant	46.2	57.0
0.3	C + L	Late-Fusion Dominant	72.2	75.1
0.5	C + L	Balanced	<b>74.4</b>	<b>76.5</b>
0.7	C + L	Mid-Fusion Dominant	73.3	75.9
0.3	C + L + R	Late-Fusion Dominant	74.2	77.1
0.5	C + L + R	Balanced	<b>75.9</b>	<b>78.1</b>
0.7	C + L + R	Mid-Fusion Dominant	74.8	77.3

**Table 7.** Effect of the FDS threshold  $\tau$  on detection performance across different sensor configurations. A lower threshold biases the system toward late fusion by making it harder to satisfy  $FDS_{global} < \tau$ , while a higher threshold increases mid-fusion dominance. Balanced performance is observed near  $\tau = 0.5$ .



**(a)** Effect of threshold  $\tau$  on mAP across fusion configurations.



**(b)** Effect of threshold  $\tau$  on NDS across fusion configurations.

**Fig. 8.** Performance variation with respect to fusion selection threshold  $\tau$  in FDSNet.

## Limitations and future work

### Limitations

Despite demonstrating strong performance and adaptability, the proposed FDSNet has some limitations that warrant consideration. First, the effectiveness of the FDS depends on accurate cross-modal alignment within

Fusion strategy	Switching mode	NDS $\uparrow$	mAP $\uparrow$	mATE $\downarrow$	mASE $\downarrow$	mAOE $\downarrow$	mAVE $\downarrow$	mAAE $\downarrow$
Mid-Fusion	Fixed	75.4	73.2	0.404	0.226	0.289	0.252	0.118
Late-Fusion	Fixed	76.5	74.3	0.395	0.223	0.281	0.238	0.112
<b>FDSNet (Ours)</b>	<b>Dynamic</b>	<b>78.1</b>	<b>75.9</b>	<b>0.385</b>	<b>0.219</b>	<b>0.275</b>	<b>0.229</b>	<b>0.105</b>

**Table 8.** Ablation study on fusion strategy and dynamic switching using all three modalities (Camera + LiDAR + Radar) on the nuScenes validation set.

FDSNet Configuration	Params (M)	GFLOPs	GPU Memory (GB)	FPS	mAP $\uparrow$	NDS $\uparrow$
C + R	<b>68.5</b>	<b>318</b>	<b>6.2</b>	<b>31.0</b>	47.9	58.2
C + L	78.9	365	7.0	27.1	74.4	76.5
C + L + R	82.4	412	7.8	24.3	<b>75.9</b>	<b>78.1</b>

**Table 9.** Computational efficiency, runtime, and accuracy of FDSNet under different sensor configurations on the nuScenes validation set.

FDSNet Configuration	Input	Car	Truck	Bus	Trailer	Constr.	Ped.	Motor	Bicycle	Traf.	Barrier	mAP	NDS $\uparrow$
C+R	C+R	81.2	55.3	33.1	61.4	57.9	68.2	51.7	32.8	84.1	77.6	<b>47.9</b>	58.2
C+L	C+L	88.5	63.1	38.9	74.6	67.8	79.1	82.5	59.8	89.4	88.7	<b>74.4</b>	76.5
C+L+R	<b>C+L+R</b>	<b>89.8</b>	<b>67.8</b>	<b>44.5</b>	<b>77.6</b>	<b>68.3</b>	<b>81.8</b>	<b>85.3</b>	<b>65.6</b>	<b>93.4</b>	<b>91.1</b>	<b>75.9</b>	<b>78.1</b>

**Table 10.** Per-class 3D detection results (mAP, NDS) for FDSNet under different sensor configurations on the nuScenes validation set.

the BEV representation. Calibration drift or asynchronous sensor timing can impair semantic consistency estimation, potentially leading to suboptimal fusion decisions. Second, the fusion threshold parameter  $\tau$ , which governs mid and late stage switching, is currently selected empirically. While the ablation study confirms stable performance around  $\tau = 0.5$ , automatic or data-driven threshold adaptation would further enhance robustness under diverse conditions. Third, FDSNet performs semantic consistency estimation on a per-frame basis, without explicitly modeling temporal correlations across frames. Incorporating temporal consistency could improve stability in rapidly changing environments or when sensor reliability fluctuates. Finally, although the efficiency analysis shows real-time inference (24–31 FPS), additional optimization will be needed for low-power or embedded deployments where computational resources are limited.

### Future work

Future research directions may address these limitations and enhance the generalization capabilities of the proposed framework. Incorporating advanced calibration aware mechanisms or self-supervised alignment approaches could mitigate sensor misalignment issues, thus improving the reliability of the FDS. Additionally, developing an adaptive and learnable threshold determination mechanism for instance, via reinforcement learning or meta learning strategies could enable automatic and context-aware selection of the fusion stage without extensive manual tuning.

### Conclusion

In this work, we presented FDSNet, a dynamic multimodal fusion framework designed to overcome the limitations of static fusion strategies in autonomous driving. By introducing the FDS, our approach quantifies semantic consistency across Camera, LiDAR, and Radar streams, enabling real-time selection between mid-level and late fusion. This adaptive mechanism ensures that the system maintains both robustness under sensor disagreement and efficiency in favorable conditions. The experiments across multiple sensor configurations demonstrate that FDSNet provides a unified solution that scales seamlessly from sparse to dense modalities. The framework highlights how adaptive stage selection can suppress cross-modal inconsistencies while avoiding unnecessary computation, making it particularly suitable for real-time perception tasks.

## Data availability

This study relies on the publicly available nuScenes dataset<sup>30</sup>, which can be accessed through the official website at: <https://www.nuscenes.org/nuscenes>. The dataset is openly accessible and was used in accordance with its respective terms of use. The corresponding author N. M. O should be contacted if someone wants to request the data from this study.

Received: 27 July 2025; Accepted: 23 October 2025

Published online: 19 December 2025

## References

1. Panduru, K. et al. Exploring the unseen: A survey of multi-sensor fusion and the role of explainable ai (xai) in autonomous vehicles. *Sensors (Basel, Switzerland)* **25**, 856 (2025).
2. Zhang, D. et al. Sparsead: Sparse query-centric paradigm for efficient end-to-end autonomous driving. *arXiv preprint arXiv:2404.06892* (2024).
3. Mohammed, A., Ibrahim, H. M. & Omar, N. M. Optimizing retinanet anchors using differential evolution for improved object detection. *Sci. Rep.* **15**, 10101 (2025).
4. Xu, H., Chen, J., Meng, S., Wang, Y. & Chau, L.-P. A survey on occupancy perception for autonomous driving: The information fusion perspective. *Inf. Fusion* **114**, 102671 (2025).
5. Rizehvandi, A., Azadi, S. & Eichberger, A. Decision-making policy for autonomous vehicles on highways using deep reinforcement learning (drl) method. *Automation* **5**, 564 (2024).
6. Yu, J. et al. Rdt-rrt: Real-time double-tree rapidly-exploring random tree path planning for autonomous vehicles. *Expert Syst. Appl.* **240**, 122510 (2024).
7. Shaaban, K. M. & Omar, N. M. Depth extraction of partially occluded objects using deformable net. *J. Vis. Commun. Image Represent* **39**, 1–11 (2016).
8. Shaaban, K. M. & Omar, N. M. 3d information extraction using region-based deformable net for monocular robot navigation. *J. Vis. Commun. Image Represent* **23**, 397–408 (2012).
9. Shovo, S. U. A., Abir, M. G. R., Kabir, M. M. & Mridha, M. Advancing low-light object detection with you only look once models: An empirical study and performance evaluation. *Cogn. Comput. Syst.* **6**, 119–134 (2024).
10. Hung, K.-F. & Lin, K.-P. Bio-inspired dark adaptive nighttime object detection. *Biomimetics* **9**, 158 (2024).
11. Dai, Z., Guan, Z., Chen, Q., Xu, Y. & Sun, F. Enhanced object detection in autonomous vehicles through lidar-camera sensor fusion. *World Electr. Veh. J.* **15**, 297–313 (2024).
12. Raut, G. & Patole, A. End-to-end 3d object detection using lidar point cloud. In *2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI)*, 1–7 (IEEE, 2024).
13. Liu, H., Liu, J., Jiang, G. & Jin, X. Mssf: A 4d radar and camera fusion framework with multi-stage sampling for 3d object detection in autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **26**, 8641–8656 (2025).
14. Wu, D., Yang, F., Xu, B., Liao, P. & Liu, B. A survey of deep learning based radar and vision fusion for 3d object detection in autonomous driving. *arXiv preprint arXiv:2406.00714* (2024).
15. Kalgaonkar, P. & El-Sharkawy, M. Nextfusion: attention-based camera-radar fusion network for improved three-dimensional object detection and tracking. *Futur. Internet* **16**, 114 (2024).
16. Lin, Z. et al. Rcbevdet: radar-camera fusion in bird's eye view for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14928–14937 (2024).
17. Liu, M., Jia, Y., Lyu, Y., Dong, Q. & Yang, Y. Bafusion: Bidirectional attention fusion for 3d object detection based on lidar and camera. *Sensors (Basel, Switzerland)* **24**, 4718 (2024).
18. Zhang, B. et al. Fogfusion: Robust 3d object detection based on camera-lidar fusion for autonomous driving in foggy weather conditions. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* **09544070251327229** (2025).
19. Guo, Z., Yagudin, Z., Asfaw, S., Lykov, A. & Tsatsas, D. Fadet: A multi-sensor 3d object detection network based on local feature attention. In *2025 IEEE Intelligent Vehicles Symposium (IV)*, 202–208 (IEEE, 2025).
20. Chen, X., Zhang, T., Wang, Y., Wang, Y. & Zhao, H. Futr3d: A unified sensor fusion framework for 3d detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 172–181 (2023).
21. Huch, S., Sauerbeck, F. & Betz, J. Deepstep-deep learning-based spatio-temporal end-to-end perception for autonomous vehicles. In *2023 IEEE Intelligent Vehicles Symposium (IV)*, 1–8 (IEEE, 2023).
22. Qiao, R., Yuan, H., Guan, Z. & Zhang, W. Mdfusion: Multi-dimension semantic-spatial feature fusion for lidar-camera 3d object detection. *Remote Sens.* **17**, 1240 (2025).
23. Sgaravatti, C. et al. A multimodal hybrid late-cascade fusion network for enhanced 3d object detection. In *European Conference on Computer Vision*, 339–356 (Springer, 2025).
24. Huang, X. et al. L4dr: Lidar-4dradar fusion for weather-robust 3d object detection. *Proc. AAAI Conf. Artif. Intell.* **39**, 3806–3814 (2025).
25. Yin, Z., Sun, H., Liu, N., Zhou, H. & Shen, J. Decoratingfusion: A lidar-camera fusion network with the combination of point-level and feature-level fusion. In *International Conference on Artificial Neural Networks*, 108–119 (Springer, 2024).
26. Wei, Z. et al. Ms-occ: Multi-stage lidar-camera fusion for 3d semantic occupancy prediction. *arXiv preprint arXiv:2504.15888* (2025).
27. Shen, J., Liu, N., Sun, H., Li, D. & Zhang, Y. An instrument indication acquisition algorithm based on lightweight deep convolutional neural network and hybrid attention fine-grained features. *IEEE Trans. Instrum. Meas.* **73**, 1–16 (2024).
28. Shen, J. et al. An algorithm based on lightweight semantic features for ancient mural element object detection. *Npj Herit. Sci.* **13**, 70 (2025).
29. Shen, J. et al. Finger vein recognition algorithm based on lightweight deep convolutional neural network. *IEEE Trans. Instrum. Meas.* **71**, 1–13 (2021).
30. Caesar, H. et al. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631 (2020).
31. Dupas, Y., Hotel, O., Lefebvre, G. & Cerin, C. Mefaf: Multimodal image early fusion with attention module for pedestrian and vehicle detection. In *20th International Conference on Computer Vision Theory and Applications*, 610–617 (SCITEPRESS-Science and Technology Publications, 2025).
32. Wu, H., Wen, C., Shi, S., Li, X. & Wang, C. Virtual sparse convolution for multimodal 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 21653–21662 (2023).
33. Cai, H. et al. Bevfusion4d: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation. *arXiv preprint arXiv:2303.17099* (2023).
34. Kim, Y. et al. Crn: Camera radar net for accurate, robust, efficient 3d perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17615–17626 (2023).

35. Zhang, T. et al. Contrastive late fusion for 3d object detection. *IEEE Trans. Intell. Veh.* **10**, 3442–3457 (2024).
36. Zhou, T. et al. Bridging the view disparity between radar and camera features for multi-modal fusion 3d object detection. *IEEE Trans. Intell. Veh.* **8**, 1523–1535 (2023).
37. Malawade, A. V., Mortlock, T. & Al Faruque, M. A. Hydrafusion: Context-aware selective sensor fusion for robust and efficient autonomous vehicle perception. In *2022 ACM/IEEE 13th International Conference on Cyber-Physical Systems (ICCPs)*, 68–79 (IEEE, 2022).
38. Niu, L. et al. Voxel-based navigation: A systematic review of techniques, applications, and challenges. *ISPRS Int. J. Geo-Information* **13**, 461 (2024).
39. Philion, J. & Fidler, S. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, 194–210 (Springer, 2020).
40. Li, Z. et al. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. (2022). URL <https://arxiv.org/abs/2203.17270> (2022).
41. Liu, Y. et al. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3262–3272 (2023).
42. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778 (2016).
43. Lin, T.-Y. et al. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125 (2017).
44. Hartley, R. & Zisserman, A. *Multiple view geometry in computer vision* (Cambridge university press, 2003).
45. Lang, A. H. et al. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12697–12705 (2019).
46. Zhou, Y. & Tuzel, O. Voxelnets: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4490–4499 (2018).
47. Yan, Y., Mao, Y. & Li, B. Second: Sparsely embedded convolutional detection. *Sensors* **18**, 3337 (2018).
48. Team, O. D. Openpcdet: An open-source toolbox for 3d object detection from point clouds. <https://github.com/open-mmlab/OpenPCDet> (2020).
49. Qi, C. R., Su, H., Mo, K. & Guibas, L. J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 652–660 (2017).
50. Shen, L. et al. Arc-bev: Attentive radar-camera fusion 3d object detection in bird-eye-view space for autonomous driving. In *International Symposium on Experimental Robotics*, 557–566 (Springer, 2023).
51. Liu, Z. et al. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, 2774–2781 (IEEE, 2023).
52. Sindagi, V. A., Zhou, Y. & Tuzel, O. Mvx-net: Multimodal voxelnet for 3d object detection. In *2019 International Conference on Robotics and Automation (ICRA)*, 7276–7282 (IEEE, 2019).
53. Ku, J., Mozifian, M., Lee, J., Harakeh, A. & Waslander, S. L. Joint 3d proposal generation and object detection from view aggregation. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, 1–8 (IEEE, 2018).
54. Wang, P. et al. The apolloscape open dataset for autonomous driving and its application. *IEEE transactions on pattern analysis and machine intelligence* **1** (2019).
55. Li, Y., Yu, R., Shahabi, C. & Liu, Y. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).
56. Paszke, A. et al. Pytorch: An imperative style, high-performance deep learning library. *Adv Neural Inform. Process. Systems* **32**, 8024–8035 (2019).
57. Contributors, M. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. <https://github.com/open-mmlab/mmdetection3d> (2020).
58. Deng, J. et al. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255 (Ieee, 2009).
59. Bai, X. et al. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1090–1099 (2022).
60. Yang, Z. et al. Deepinteraction: 3d object detection via modality interaction. *Adv. Neural Inf. Process. Syst.* **35**, 1992–2005 (2022).
61. Loshchilov, I. & Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).
62. Nabati, R. & Qi, H. C. Center-based radar and camera fusion for 3d object detection. *arxiv* 2020. *arXiv preprint arXiv:2011.04841* (2011).
63. Kim, Y., Kim, S., Choi, J. W. & Kum, D. Craft: Camera-radar 3d object detection with spatio-contextual fusion transformer. *Proc. AAAI Conf. Artif. Intell.* **37**, 1160–1168 (2023).
64. Baumann, N. et al. Cr3dt: camera-radar fusion for 3d detection and tracking. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4926–4933 (IEEE, 2024).
65. Huang, J., Huang, G., Zhu, Z., Ye, Y. & Du, D. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790* (2021).
66. Li, Y. et al. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. *Proc. AAAI Conf. Artif. Intell.* **37**, 1477–1485 (2023).
67. Park, J. et al. Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection. *arXiv preprint arXiv:2210.02443* (2022).
68. Wang, S., Liu, Y., Wang, T., Li, Y. & Zhang, X. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3621–3631 (2023).
69. Shi, P. et al. Polarfusion: A multi-modal fusion algorithm for 3d object detection based on polar coordinates. *Neural Netw.* **190**, 107704 (2025).
70. Yin, J. et al. Is-fusion: Instance-scene collaborative fusion for multimodal 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14905–14915 (2024).
71. Mohan, R., Cattaneo, D., Drews, F. & Valada, A. Progressive multi-modal fusion for robust 3d object detection. In *8th Annual Conference on Robot Learning* (2024).
72. Ulrich, M. et al. Improved orientation estimation and detection with hybrid object detection networks for automotive radar. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, 111–117 (IEEE, 2022).
73. Wu, Z., Chen, G., Gan, Y., Wang, L. & Pu, J. Mvfusion: Multi-view 3d object detection with semantic-aligned radar and camera fusion. *arXiv preprint arXiv:2302.10511* (2023).
74. Liu, H., Teng, Y., Lu, T., Wang, H. & Wang, L. Sparsebev: High-performance sparse 3d object detection from multi-camera videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, 18580–18590 (2023).

## Author contributions

A.M., N.O., and H.I. contributed to the conceptual development of the FDSNet framework. A.M. and N.O. designed the feature disagreement scoring mechanism and the dynamic fusion stage selection strategy. A.M. implemented the FDSNet architecture, including modality-specific BEV feature extractors and the fusion modules. A.M. and N.O. conducted the experiments, performed comparative evaluations with existing fusion methods, and analyzed the results. A.M. and N.O. provided theoretical insights, refined the design of the semantic consistency metric, and contributed to manuscript revision. A.M. and N.O. wrote the main manuscript text. All authors reviewed and approved the final manuscript.

## Funding

Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

## Declarations

### Competing interests

The authors declare no competing interests.

### Additional information

**Correspondence** and requests for materials should be addressed to N.M.O.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2025