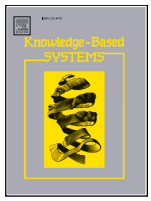
Knowledge-Based Systems 339 (2026) 115616

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



MKFusion: Multi-modal knowledge distillation for 4D radar point cloud segmentation in autonomous driving

Yunting Yang, Jun Liu *, Hongsi Liu, Guangfeng Jiang

Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, 230027, China

ARTICLE INFO

Keywords:
4D radar
Automotive radar
Point cloud segmentation
Multimodal fusion
Knowledge distillation
Autonomous driving

ABSTRACT

Semantic segmentation of 4D radar point clouds is an emerging way to exploit radar's low cost and robustness for safety-critical 3D perception in autonomous driving. However, these point clouds are extremely sparse and exhibit radar-specific artifacts such as multipath ghosts, partial penetrations, and depth ambiguity with respect to the image plane, which makes accurate point-wise segmentation and robust radar-camera fusion challenging. To tackle these challenges, we propose MKFusion, a sparse multi-modal network that combines multi-scale radar-camera fusion with knowledge distillation to significantly improve 4D radar point cloud segmentation. Specifically, our multi-modal branch incorporates camera information through multi-scale image feature sampling and depth-aware fusion (MSDAF), enriching the semantic representation of radar features and mitigating the performance degradation caused by radar noise. In addition, we design a sparse-aligned distillation (SAD) module that enables radar features to efficiently learn from LiDAR representations, facilitating semantic alignment within the sparse framework. Experimental results demonstrate that the proposed modules significantly improve radar point cloud segmentation performance and achieve state-of-the-art results on the View-of-Delft and TJ4DRadSet datasets compared to existing methods. Code is available at: https://github.com/guineapig5151/mkfusion.

1. Introduction

Semantic segmentation plays a vital role in large-scale outdoor scene understanding, with broad applications in autonomous driving [1–4]. Among the existing sensors, LiDAR has become the dominant choice for 3D point cloud segmentation [1,5] due to its accurate spatial measurements. However, the high cost of LiDAR sensors and their susceptibility to performance degradation under adverse weather conditions [6–8] hinder their scalability in real-world autonomous driving systems.

In contrast, 4D millimeter-wave radar has emerged as a promising alternative due to its cost-effectiveness and robustness under adverse weather conditions. 4D radar extends traditional radar capabilities by capturing not only the range, azimuth, and Doppler velocity of objects but also their elevation angle, enabling the generation of high-resolution 3D spatial information. However, the point clouds generated by 4D radar are still inherently sparse compared with LiDAR [9] and exhibit radar-specific artifacts, such as multipath ghosts and partial penetrations. These factors, together with the depth ambiguity of radar returns with respect to the image plane, make accurate point-wise segmentation and fine-grained radar–camera fusion particularly challenging. To mitigate this limitation, point cloud representations have been effectively enriched by leveraging the rich semantic information from cameras [9–13]. Motivated by this, our segmentation approach incorporates multi-modal information to enhance the quality of radar-based perception. Fine-grained 4D radar–camera segmentation requires robust feature fusion between highly heterogeneous modalities. A key challenge lies in bridging the modality gap between sparse radar point clouds and dense image features, while simultaneously resolving depth ambiguity at fusion time. To address the above challenges in fine-grained radar-camera fusion, we propose a multi-scale depth-aware fusion (MSDAF) module. This module effectively integrates radar and camera features into a unified and informative scene representation, enabling accurate and robust semantic segmentation. Specifically, we introduce an adaptive learner fusion mechanism to alleviate the modality gap between heterogeneous sensors. Furthermore, we incorporate a depth prior to make the fusion process depth-aware, which helps mitigate the depth ambiguity caused by perspective projection. Extensive experiments demonstrate that addressing these issues results in significant improvements in segmentation performance.

In addition, we observe that LiDAR features offer both high-level semantic understanding and precise geometric structure. To further narrow the performance gap between radar and LiDAR, knowledge distillation from LiDAR to radar networks is a promising strategy, without inference-time overhead [14]. However, existing multimodal distillation

* Corresponding author.
E-mail addresses: yangyt37@mail.ustc.edu.cn (Y. Yang), junliu@ustc.edu.cn (J. Liu), liuhs3@mail.ustc.edu.cn (H. Liu), jgf1998@mail.ustc.edu.cn (G. Jiang).

https://doi.org/10.1016/j.knosys.2026.115616
Received 10 July 2025; Received in revised form 11 February 2026; Accepted 23 February 2026
Available online 27 February 2026

methods primarily focus on object detection [14,15] or occupancy prediction [16], and typically operate in the bird's-eye view (BEV) domain, which limits their applicability to fine-grained, point-wise semantic segmentation. Considering that both LiDAR and radar are represented in the form of point clouds and can be processed in a uniform network, we propose a sparse-aligned distillation (SAD) module to enable effective, fine-grained knowledge transfer in the point-wise domain. The proposed SAD module enables knowledge distillation from a LiDAR-radar teacher network to a radar-only student network through a two-stage process. This facilitates the transfer of rich semantic information while preserving the intrinsic advantages of radar sensing, such as robustness to adverse weather, low cost, and fast response.

Our contributions are summarized as follows:

- We introduce MKFusion, a novel radar-centric semantic segmentation framework that integrates multi-modal fusion and knowledge distillation within a sparse architecture.
- We propose a multi-scale depth-aware fusion module, which effectively mitigates the modality gap and depth ambiguity in radar-camera fusion, enriching the semantic representation of radar features.
- We propose a sparse-aligned distillation module that enables efficient knowledge transfer from a LiDAR-radar teacher model in the point-wise domain. This module facilitates robust cross-modal distillation, significantly enhancing the performance of both radar-only and radar-camera student models.
- We conduct extensive experiments on the widely used View-of-Delft (VoD) and TJ4DRadSet datasets, where our method significantly outperforms baseline approaches and achieves state-of-the-art performance on 4D radar point cloud segmentation.

The rest of our article is structured as follows. Section 2 provides a brief review of recent advances in point cloud segmentation, multi-modal fusion, and knowledge distillation. Section 3 introduces our proposed network. In Section 4, we present the experimental setups, implementation details, comparisons with state-of-the-art methods, and analysis of ablation studies. Lastly, our work is summarized in Section 5.

2. Related work

2.1. 3D point cloud segmentation based on LiDAR point cloud

LiDAR semantic segmentation targets the task of predicting a semantic category for every individual point in a sequence of LiDAR point clouds. Semantic segmentation methods in the current literature are generally divided into four major categories: voxel-based [4,17], projection-based [3,18,19], and point-based approaches [20–23].

Voxel-based methods first convert raw point clouds into regular voxel grids, enabling the use of convolutional neural networks for feature extraction. MinkNet [4] is a representative voxel-based approach, which begins by voxelizing the raw point cloud. It then adopts a U-Net-like architecture and employs sparse convolutional operators to efficiently extract features from the 3D voxel grid.

Projection-based methods first transform the 3D point cloud into a 2D range image, and then extract semantic features using standard 2D convolutional networks. SqueezeSegV3 [18] is a representative projection-based approach. It transforms the 3D point cloud into a 2D range image using spherical projection, and subsequently applies a series of standard and adaptive convolutional layers for semantic feature extraction.

Point-based methods process raw point clouds using point-wise or neighborhood-based operations. PointNet++ [23] captures local geometric structures via hierarchical PointNet-based [24] set abstraction. Point Transformer v3 [20] employs a serialized neighborhood mapping mechanism to efficiently expand the receptive field, achieving state-of-the-art results on over 20 indoor and outdoor tasks.

In this work, we comprehensively consider the trade-offs between accuracy, efficiency, and modality alignment, and adopt a voxel-based framework MinkNet tailored for 4D radar segmentation.

2.2. 3D point cloud segmentation based on radar point cloud

Recent research on 3D point cloud segmentation using radar data has predominantly focused on traditional automotive radar sensors, which generate sparser point clouds with limited elevation information compared to modern 4D imaging radars. Several recent works [7,25–27] have explored radar point cloud segmentation by addressing the challenges of sparsity and irregular distribution. Datasets such as RadarScenes [28] provide real-world automotive radar point clouds with point-wise annotations and have enabled a series of semantic and instance segmentation methods on conventional 3D radar sensors. Gaussian Radar Transformer [25] adopts a self-attention-based framework for single-scan segmentation, introducing a Gaussian-weighted attention mechanism and receptive field expansion modules. STA-Net [7] aggregates multi-scan radar data to model spatiotemporal interactions and introduces prompt-based learning for improved class-wise discrimination, while RadarGNN [27] incorporates a graph neural network for radar feature extraction.

For 4D imaging radar, explicit point-level semantic segmentation is still relatively scarce. RaSS [29] constructs the ZJUSSet dataset with point-wise labels for both 4D radar and LiDAR, and introduces a cross-modal distillation framework to supervise radar point-cloud segmentation. Beyond such point-cloud-based methods, TARSS-Net [30] and TransRadar [31] perform radar semantic segmentation directly on range–azimuth (RA) and range–Doppler (RD) views. MetaOcc [32] and RadarOcc [33] instead leverage 4D imaging radar to predict semantic 3D occupancy volumes from RAD tensors or fused BEV representations. WaterScenes [34] provides a multi-task 4D radar-camera dataset and benchmarks for object detection, (instance) semantic segmentation, free-space segmentation, and waterline segmentation on water surfaces. In parallel, BEVCar [35] starts from point-encoded automotive radar and fuses it with camera images for joint BEV map and object segmentation.

However, most of the above works operate on low-resolution radar or RAD tensors and mainly focus on BEV segmentation or occupancy prediction, rather than point-level understanding on 4D radar point clouds. To bridge this gap, we propose MKFusion, a framework specifically designed for point-wise semantic segmentation on 4D radar point clouds within a fully sparse 3D architecture, with multimodal radar–camera fusion and LiDAR–radar distillation performed directly at 4D radar point clouds.

2.3. Point cloud-image fusion

Recent point cloud-image fusion methods predominantly focus on downstream tasks such as object detection [9,36–43] and occupancy prediction [44,45]. Existing fusion strategies can be broadly categorized into three types: concatenation, adaptive weighted fusion, and cross-attention.

Concatenation-based methods [10,36,37,44] typically concatenate 3D feature volumes from different modalities along the channel dimension, followed by convolutional layers to learn joint representations. Adaptive weighted fusion methods [38,45,46] employ 3D convolutions to estimate modality-specific fusion weights, which are then used to adaptively aggregate features from the point cloud and camera branches. In contrast, cross-attention-based methods [9,47] utilize deformable attention to align and fuse voxel features with corresponding image features. These approaches estimate image pixel offsets from calibrated projections and selectively aggregate image features to enhance the voxel-space representation.

More recently, several works have explored dedicated 4D radar–camera fusion for 3D detection on VoD [48] and TJ4DRadSet [49]. LXL [10] and MSSF [9] adopt sampling-based fusion to deeply interact

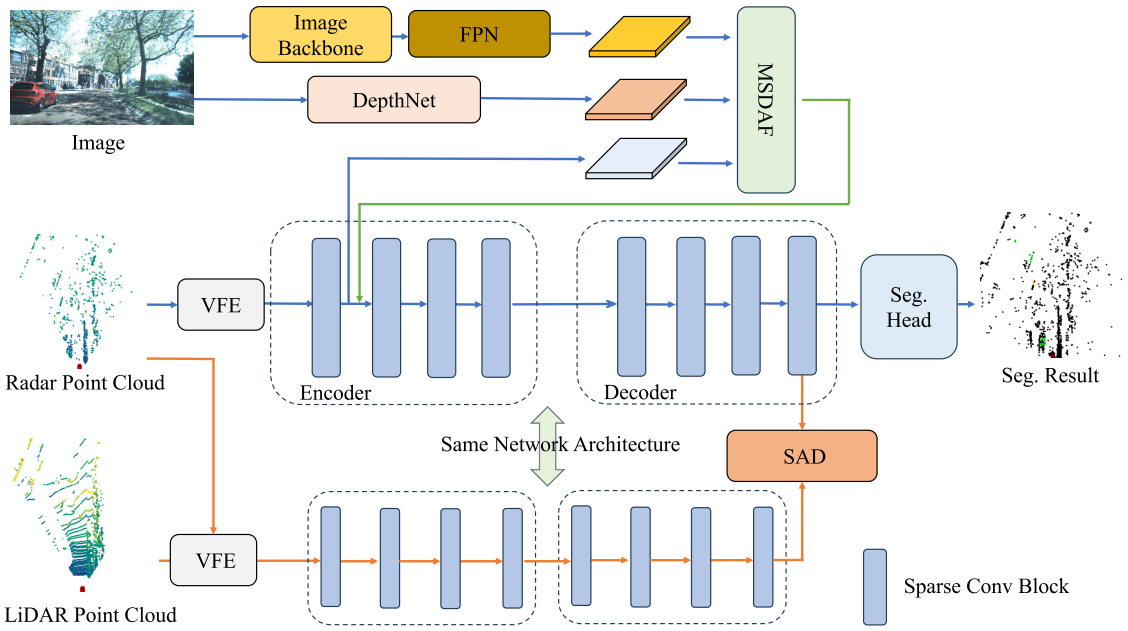


Fig. 1. Overview of the MKFusion architecture. Orange paths are active only during training. The framework comprises radar-camera feature extractors, an MSDAF module for cross-modal enhancement, and a SAD module that transfers LiDAR knowledge to the radar branch during training.

sparse radar points with image features, UniBEVFusion [37] proposes a unified radar–vision BEV fusion framework with radar-guided depth estimation and shared BEV feature extraction, while CVFusion [40] designs cross-view and dual-sampling modules for more effective radar–image alignment. RCBEVDet [41] explores BEV-based and attention-based fusion schemes for 3D radar and camera. Doracamom [42] jointly fuses multi-view 4D radars and cameras for unified 3D detection and semantic occupancy prediction.

Our method is built upon adaptive weighted fusion, a fine-grained fusion strategy tailored for 4D radar segmentation tasks rather than detection. To mitigate the modality gap between point clouds and images, we introduce an adaptive learner fusion mechanism. Furthermore, to address the inherent depth ambiguity in perspective projection, we propose a depth-aware fusion approach that integrates point cloud and image features, enabling more accurate alignment between radar points and image features in 3D space.

2.4. Knowledge distillation based on radar point cloud

Knowledge distillation was originally proposed for model compression and performance improvement in image classification tasks [50]. In the context of radar point cloud processing, most existing cross-modality distillation approaches focus on 3D object detection [14,15,51] or occupancy prediction [16]. For example, Xu et al. [15] present a semi-supervised distillation framework for four-dimensional radar-based three-dimensional detection. It employs an adaptive fusion module in the teacher network to integrate LiDAR and radar features, and proposes two feature distillation strategies: LiDAR-to-radar feature distillation and fusion-to-radar feature distillation, as well as semi-supervised output distillation to improve student performance under limited supervision. Ma et al. [16] target semantic scene completion and observe that LiDAR-camera fusion performs well in outdoor settings. They introduce a fusion-based distillation method that transfers informative cues from a LiDAR-camera teacher to both a radar-only baseline and a radar-camera fusion student. The method combines cross-model residual distillation, bird's-eye view relation distillation, and predictive distribution distillation to hierarchically guide feature and distribution learning. Similarly, Zhao et al. [14] propose a camera-radar knowledge distillation framework to narrow the performance gap between LiDAR-camera and camera-radar detectors. By leveraging bird's-eye view as a shared representation space, they introduce four customized distillation losses to effectively transfer crucial features from the teacher to the student model.

Closer to our task, RaSS [29] develops a cross-modal knowledge distillation framework for 4D radar semantic segmentation on ZJUSet [29] and VoD. It performs LiDAR-to-radar distillation with BEV feature aggregation, but supervision is limited to BEV cells occupied by both LiDAR and radar. As a result, regions occupied only in radar may receive no guidance. Moreover, RaSS does not incorporate camera cues in its distillation pipeline.

In contrast, our sparse-aligned distillation operates directly in the sparse 3D point-wise domain with a *LiDAR-radar* teacher aligned to radar space. SAD uses k-nearest neighbor feature aggregation to provide dense supervision at all radar-occupied voxels. This design reduces the modality gap between teacher and student, while preserving the intrinsic advantages of radar sensing and allowing the student to benefit from both LiDAR geometry and image semantics.

3. Proposed method

3.1. Overall architecture

The overall architecture of the network is illustrated in Fig. 1, which consists of three main components: the radar and camera feature extraction, the MSDAF module, and the SAD module.

1. The radar and camera feature extraction module extracts semantic features from both modalities independently.
2. The MSDAF module incorporates image semantics and depth cues to enhance radar feature representations through cross-modal fusion.
3. The SAD module distills LiDAR knowledge to the radar branch during training via feature-level distillation.

We provide a detailed introduction of each component in the following sections.

3.2. Radar and camera feature extraction

For a given 4D radar point cloud, we first voxelize the raw points using mean voxel feature encoding. Considering the sparsity of radar point

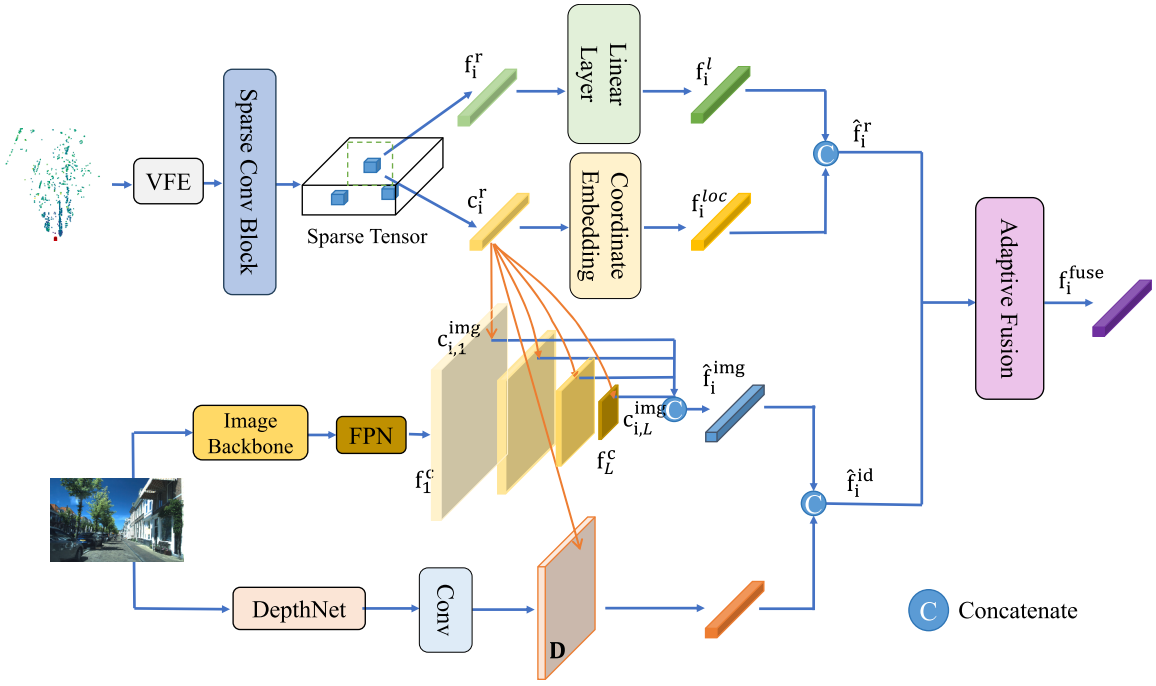


Fig. 2. The overall architecture of our MSDAF module. It consists of multi-scale image feature sampling and depth-aware fusion attention.

clouds, we adopt SparseUNet [4] as the backbone network to enable efficient and effective radar feature extraction. The encoder consists of four sparse convolutional layers with a stride of 2, which progressively reduces the spatial resolution of the input sparse tensor and captures multi-scale contextual information. For the camera images, we employ a ResNet-50 [52] backbone with a feature pyramid network (FPN) [53] as the image feature extractor.

Formally, the input to the radar branch is a sparse point cloud $\mathbf{x}^{\text{radar}} \in \mathbb{R}^{N\times C}$, where N is the number of radar points and C is the number of per-point input features. Each sparse convolutional layer outputs a sparse tensor $S_i = (\mathbf{f}_i^r, \mathbf{c}_i^r)$, where $\mathbf{f}_i^r \in \mathbb{R}^{N_i\times C_i}$ denotes the voxel-wise feature representations, and $\mathbf{c}_i^r \in \mathbb{R}^{N_i\times 3}$ contains the corresponding voxel coordinates. Here, N_i represents the number of non-empty voxels, and C_i is the feature dimension at the ith layer.

Similarly, the input to the image branch is an RGB image $\mathbf{x}^{\text{img}} \in \mathbb{R}^{H\times W\times 3}$, where H and W denote the height and width of the image, respectively. The output consists of L multi-scale feature maps, denoted as $\{\mathbf{f}_j^c \in \mathbb{R}^{H_j\times W_j\times C_j}\}_{j=1}^{L}$, where H_j, W_j, and C_j represent the height, width, and channel dimension of the image features at the jth feature level, respectively.

3.3. Multi-scale depth-aware fusion module

The radar and camera features extracted earlier primarily capture geometric and semantic cues, respectively. To fully leverage these complementary modalities, two key challenges have to be addressed: the modality gap and depth ambiguity. The proposed MSDAF module is introduced to mitigate the performance degradation caused by these issues. It consists of two main steps: multi-scale image feature sampling and depth-aware fusion attention, as demonstrated in Fig. 2.

3.3.1. Multi-scale image feature sampling

Given a sparse tensor $S_i = (\mathbf{f}_i^r, \mathbf{c}_i^r)$ at a certain layer of the SparseUNet backbone, we first extract the coordinates $\mathbf{c}_i^r$ of all non-empty voxels. These voxel indices are then converted into real-world coordinates based on the predefined voxel size and the point cloud range. This mapping is performed by a transformation matrix $\mathbf{T} \in \mathbb{R}^{4\times 4}$, which encodes the necessary scaling and translation to align voxel coordinates with real-world space. Next, to align radar and image features, the 3D coordinates of each voxel center are projected onto the image plane using the camera intrinsic matrix $\mathbf{T}_{\text{intr}} \in \mathbb{R}^{3\times 4}$ and the radar-to-camera extrinsic matrix $\mathbf{T}_{\text{r2c}} \in \mathbb{R}^{4\times 4}$. Let $\mathbf{c}_i^r \in \mathbb{R}^3$ denote the 3D coordinate of the ith voxel center in the radar coordinate system, and $\mathbf{c}_i' = [\mathbf{c}_i^r; 1] \in \mathbb{R}^4$ be its homogeneous representation. The projection onto the image plane can be formulated as:

$$\mathbf{c}_i^{\text{img}} = \mathbf{T}_{\text{intr}} \cdot \mathbf{T}_{\text{r2c}} \cdot \mathbf{T} \cdot \mathbf{c}_i', \tag{1}$$

where $\mathbf{c}_i^{\text{img}} = [u_i d_i, v_i d_i, d_i]^\top$ represents the homogeneous image coordinate. The final pixel location (u_i, v_i) is obtained by normalizing with respect to the depth value d_i.

To fully exploit the semantic richness of the image features, we sample visual information at the projected locations from multiple levels of the image feature pyramid, thereby obtaining a multi-scale representation. The aggregated image feature corresponding to the ith projected point is defined as:

$$\hat{\mathbf{f}}_i^{\text{img}} = \text{Concat}\Big(\text{Sample}(\mathbf{f}_1^c, \mathbf{c}_{i,1}^{\text{img}}), \ldots, \text{Sample}(\mathbf{f}_L^c, \mathbf{c}_{i,L}^{\text{img}})\Big), \tag{2}$$

where $\mathbf{c}_{i,j}^{\text{img}}$ denotes the projected 2D location of the ith 3D voxel center on the jth image feature level, and $\text{Sample}(\cdot,\cdot)$ represents bilinear interpolation.

3.3.2. Depth-aware fusion attention

As radar data are inherently sparse and primarily geometric, whereas camera images are dense and contain rich appearance cues, resulting in a significant modality gap between the two. To mitigate this discrepancy, we apply a learnable, lightweight linear transformation followed by a ReLU activation to the radar features, serving as a radar feature aligner that adjusts their distribution to better match that of the image features. Formally, the transformed radar feature is defined as:

$$\mathbf{f}_i^l = \sigma_r\big(L(\mathbf{f}_i^r)\big), \tag{3}$$

where $\mathbf{f}_i^r$ denotes the original radar feature, and $\mathbf{f}_i^l$ is the modality-aligned representation, which is subsequently fused with the 2D image feature. Here, $\sigma_r(\cdot)$ represents the ReLU activation function, and $L(\cdot)$ denotes a linear transformation.

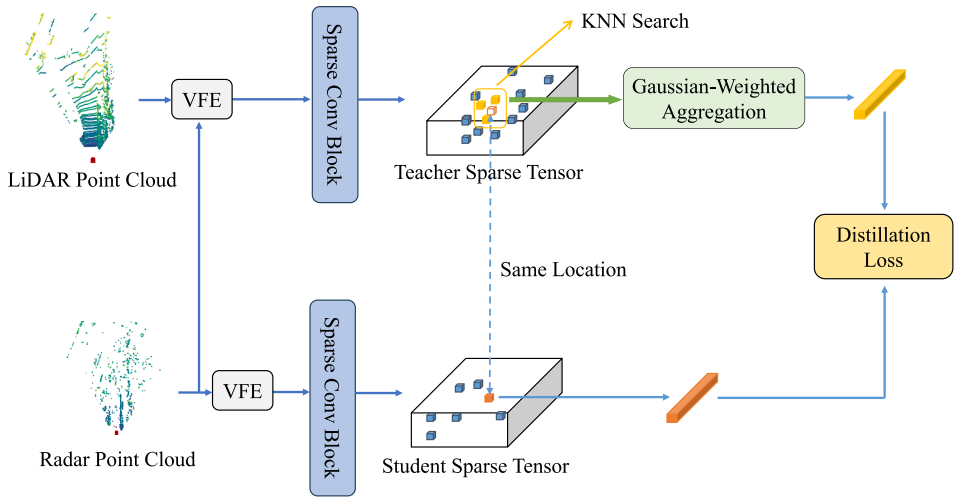


Fig. 3. The overall architecture of our SAD module. It consists of sparse k nearest neighbor (KNN)-based feature aggregation and voxel-wise feature distillation.

Besides, although radar can penetrate certain materials, its noisy and low-resolution nature often leads to depth ambiguity when projected onto the image plane via calibration matrices. To alleviate this issue, we enhance the input representations by augmenting the radar features with their 3D spatial coordinates, and embedding estimated depth values into the image features. This augmentation facilitates more accurate spatial reasoning during fusion and helps the model disambiguate depth-related uncertainty across modalities.

Specifically, for radar features, we first compute the global 3D coordinates of all non-empty voxels using the predefined voxel size and point cloud range. These global coordinates are then used to compute a positional encoding vector for each voxel. The resulting location embedding, denoted as $\mathbf{f}_i^{\text{loc}}$, is derived from passing the 3D spatial position of the ith voxel through a multi-layer perception (MLP), and is concatenated with its corresponding aligned feature before fusion:

$$\hat{\mathbf{f}}_i^r = \text{Concat}(\mathbf{f}_i^l, \mathbf{f}_i^{\text{loc}}), \tag{4}$$

where $\mathbf{f}_i^l$ is the modality-aligned radar feature and $\hat{\mathbf{f}}_i^r$ is the enhanced radar representation used in the subsequent fusion process.

In addition, for image features, we first obtain depth features utilizing a fixed-weight depth estimation network [54]. These depth maps are then processed through learnable convolutional layers to extract informative depth-aware representations $\mathbf{D}$. Then, we sample the depth feature for each non-empty voxel and concatenate it with the corresponding multi-scale image features to form depth-augmented image representations. The augmented 2D features are thus represented as:

$$\hat{\mathbf{f}}_i^{id} = \text{Concat}(\hat{\mathbf{f}}_i^{\text{img}}, \text{Sample}(\mathbf{D}, c_i^{\text{img}})), \tag{5}$$

where $\hat{\mathbf{f}}_i^{\text{img}}$ defined in (2) denotes the sampled multi-scale image features at location i, and $\mathbf{D}$ represents the learned depth feature. These enhanced features are subsequently used in the adaptive fusion block.

Finally, the enhanced radar feature $\hat{\mathbf{f}}_i^r$ and the depth-aware image feature $\hat{\mathbf{f}}_i^{id}$ are further fused through an attention-based adaptive interaction mechanism to produce the final cross-modal representation. Specifically, we first concatenate the two features and apply a feature learner $f(\cdot)$, implemented as an MLP, to model their joint representation:

$$\mathbf{f}_i^{\text{cat}} = f\Big(\text{Concat}(\hat{\mathbf{f}}_i^{id}, \hat{\mathbf{f}}_i^r)\Big). \tag{6}$$

Then, we adopt the attention-based fusion method to obtain the final features:

$$\mathbf{f}_i^{\text{fuse}} = L_3\big(\sigma_r\big(L_1(\mathbf{f}_i^{\text{cat}}) \odot \sigma\big(L_2(L_1(\mathbf{f}_i^{\text{cat}}))\big)\big)\big), \tag{7}$$

where $\sigma(\cdot)$ denotes the sigmoid function, $\sigma_r(\cdot)$ represents the ReLU activation function, $\odot$ indicates element-wise multiplication, and L_k represents the kth linear layer. Then, the fused feature $\mathbf{f}_i$ is combined with the radar coordinate $\mathbf{c}_i$ to form a new sparse tensor, replacing the original input for subsequent processing.

3.4. Sparse-aligned distillation module

LiDAR features contain both rich semantic context and precise geometric information. Knowledge distillation offers a promising approach to transferring this knowledge to radar-based models, potentially enhancing performance without adding computational overhead during inference.

We observe that both LiDAR and 4D radar can be represented in the form of point clouds, exhibiting strong modality homogeneity. This characteristic enables the use of a unified network to process both modalities simultaneously, providing a foundation for fine-grained knowledge distillation and modality alignment.

Based on the above insights, we design the SAD module, which transfers knowledge from a LiDAR and radar teacher network to the radar branch through a two-stage distillation process. The overall architecture of SAD is shown in Fig. 3. During teacher training, the segmentation loss is always computed at radar locations by extracting the fused LiDAR–radar features on radar voxels, which encourages LiDAR geometry and semantics to be expressed in radar space and reduces the modality gap before distillation.

In the first stage, rich semantic features are embedded into the radar representation via sparse k nearest neighbor (KNN)-based feature aggregation, effectively integrating cross-modal knowledge into the radar domain. In the second stage, the distilled radar features transfer knowledge to the student network, enhancing cross-modal alignment.

3.4.1. Sparse KNN-based feature aggregation

Our goal is to transfer LiDAR knowledge to radar-based models. Compared with LiDAR, radar data are sparse and noisy. To mitigate the modality gap between LiDAR and radar, the teacher network adopts the same SparseUNet architecture as the student but takes both LiDAR and radar inputs to provide richer and more informative guidance.

Specifically, given the teacher sparse tensor $\mathcal{T}$ and the student sparse tensor $\mathcal{S}$, we first eliminate duplicate voxel coordinates in $\mathcal{T}$ by applying voxel-wise feature aggregation using the inverse index map. This operation yields a unique sparse tensor with features $\mathbf{f}^{\text{teacher}} \in \mathbb{R}^{N_T \times C}$ and corresponding coordinates $\{\mathbf{c}_i^{\text{teacher}}\}_{i=1}^{N_T}$, where N_T is the number of non-empty voxels and C is the feature dimension.

To estimate the teacher-aligned features at the voxel positions $\{\mathbf{c}_j^{\text{student}}\}_{j=1}^{N_S}$, we perform KNN feature interpolation. For each student voxel j, we find its k nearest neighbors from the teacher voxel centers $\{\mathbf{c}_i^{\text{teacher}}\}$ and apply a Gaussian-weighted average to compute the aggre-

gated feature:

$$\hat{\mathbf{f}}_j^{\text{teacher}} = \sum_{i \in \mathcal{N}_k(j)} w_{ji} \cdot \mathbf{f}_i^{\text{teacher}}, \tag{8}$$

$$w_{ji} = \frac{\exp\left(-\frac{\|\mathbf{c}_j^{\text{student}} - \mathbf{c}_i^{\text{teacher}}\|^2}{2\sigma^2}\right)}{\sum_{i' \in \mathcal{N}_k(j)} \exp\left(-\frac{\|\mathbf{c}_j^{\text{student}} - \mathbf{c}_{i'}^{\text{teacher}}\|^2}{2\sigma^2}\right)}, \tag{9}$$

where $\mathcal{N}_k(j)$ denotes the index set of the k nearest teacher voxels to the jth student voxel, and w_{ji} is the normalized Gaussian weight that measures the similarity between student and teacher voxel centers. Here, σ is a fixed hyperparameter that controls the sharpness of the attention distribution.

Finally, the aggregated features $\hat{\mathbf{f}}_j^{\text{teacher}}$ are then combined with the original student coordinates to form a new sparse tensor:

$$\hat{\mathcal{T}} = \left\{ \left(\mathbf{c}_j^{\text{student}}, \hat{\mathbf{f}}_j^{\text{teacher}} \right) \right\}_{j=1}^{N_S}, \tag{10}$$

which can be used for subsequent distillation loss computation or voxel-wise fusion with student features.

3.4.2. Voxel-wise feature distillation

Following the sparse aggregation-based feature alignment, we introduce the following feature distillation loss to promote consistency between radar and LiDAR representations:

Cosine Distance Loss. Let $\mathbf{f}_j^{\text{student}} \in \mathbb{R}^C$ and $\hat{\mathbf{f}}_j^{\text{teacher}} \in \mathbb{R}^C$ denote the student and teacher features at the jth spatial position, respectively, where C is the number of channels. The cosine distance-based distillation loss is defined as:

$$\mathcal{L}_{\text{CD}} = \frac{1}{N} \sum_{j=1}^{N} \left(1 - \frac{\mathbf{f}_j^{\text{student}} \cdot \hat{\mathbf{f}}_j^{\text{teacher}}}{\left\|\mathbf{f}_j^{\text{student}}\right\| \cdot \left\|\hat{\mathbf{f}}_j^{\text{teacher}}\right\|} \right), \tag{11}$$

where N is the total number of non-empty voxels. This loss measures the angular difference between the teacher and student features, independent of their magnitudes.

L1 Loss. The L1 distillation loss is defined as:

$$\mathcal{L}_{\text{L1}} = \frac{1}{N} \sum_{j=1}^{N} \left\| \mathbf{f}_j^{\text{student}} - \hat{\mathbf{f}}_j^{\text{teacher}} \right\|_1. \tag{12}$$

This loss directly minimizes the element-wise absolute difference between student and teacher features, encouraging the student to closely match the teacher's feature representations in magnitude and structure.

Total Distillation Loss. To jointly benefit from both angular and magnitude alignment, we combine the cosine distance loss and the L1 loss as the final distillation objective:

$$\mathcal{L}_{\text{RD}} = \lambda_{\text{L1}} \mathcal{L}_{\text{L1}} + \lambda_{\text{CD}} \mathcal{L}_{\text{CD}}, \tag{13}$$

where λ_{L1} and λ_{CD} are hyperparameters that balance the contributions of the L1 and cosine distance losses, respectively. In our experiments, we simply set $\lambda_{\text{L1}} = \lambda_{\text{CD}} = 1$. The combination of L1 and cosine distance losses encourages the student to align with the teacher both in direction and in feature magnitude.

3.5. Loss

The overall training objective comprises two components: a segmentation loss that supervises the primary task, and a radar distillation loss that transfers semantic knowledge from the LiDAR modality to the radar branch.

Let $\mathcal{L}_{\text{seg}}$ denote the segmentation loss, which can be implemented as cross-entropy loss. Combined with the radar distillation loss $\mathcal{L}_{\text{RD}}$, the total training loss is defined as:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{seg}} \mathcal{L}_{\text{seg}} + \lambda_{\text{RD}} \mathcal{L}_{\text{RD}}, \tag{14}$$

where λ_{seg} and λ_{RD} are the weights used to balance these two losses. This joint optimization enables the radar branch to learn discriminative representations guided by LiDAR features while preserving its modality-specific properties. In our experiments, we set $\lambda_{\text{seg}} = \lambda_{\text{RD}} = 1$.

4. Experments and analysis

4.1. Datasets and metrics

4.1.1. Datasets

We evaluate our method on the public 4D radar datasets tailored for autonomous driving: the VoD dataset [48] and the TJ4DRadset dataset [49].

The VoD dataset features multi-sensor data collected from real-world driving scenarios, including 4D radar, a 64-beam LiDAR, stereo cameras, and GNSS/IMU. It provides comprehensive annotations, such as 2D/3D bounding boxes, tracking IDs, and ego-vehicle motion data. In total, the dataset contains 3D bounding box annotations for over 26,000 pedestrians, 10,000 cyclists, and 26,000 cars. VoD consists of 8682 frames, which are split into 5139 for training, 1296 for validation, and 2247 for testing. Since the official evaluation server is currently unavailable, all our evaluations and ablation studies are conducted on the validation set, following the protocol adopted by prior works [9,10,55] for fair comparison. Additionally, the dataset provides motion-compensated radar point clouds obtained by accumulating multiple consecutive scans. Specifically, both three-frame and five-frame compensated radar sequences are available, allowing evaluation of temporal fusion strategies under different time horizons.

It is important to note that the VoD dataset does not provide per-point segmentation labels. Therefore, we generate pseudo labels based on the available 3D bounding boxes: radar points located inside ground-truth bounding boxes are treated as belonging to the corresponding object class, while points outside any box are labeled as background. Furthermore, categories with insufficient point counts are discarded. The final retained classes include: background, car, pedestrian, cyclist, bicycle, bicycle rack, moped scooter, rider, motor, truck, and ride other.

The TJ4DRadSet dataset provides synchronized data collected from multiple sensors, including 4D radar, cameras, and GNSS. Notably, although its sensor configuration is similar to that of the VoD dataset, only synchronized 4D radar and camera data are available-LiDAR data are not released. As a result, our distillation module, which relies on LiDAR input, cannot be applied to this dataset. This dataset contains a total of 7746 frames, with 5706 frames used for training and 2040 for testing. Unlike the VoD dataset, TJ4DRadSet does not provide multi-frame radar accumulation or ego-vehicle motion information, and all prior work has been conducted on single-frame inputs. Therefore, we also adopt single-frame radar data as input for semantic segmentation. In our experiments, we focus on four key categories: car, pedestrian, cyclist, and truck.

4.1.2. Evaluation metrics

We adopt the mean intersection over union (mIoU) as the primary evaluation metric for semantic segmentation. Given a set of semantic classes, the intersection over union (IoU) for each class is computed as:

$$\text{IoU}_c = \frac{TP_c}{TP_c + FP_c + FN_c}, \tag{15}$$

where TP_c, FP_c, and FN_c denote the numbers of true positives, false positives, and false negatives for class c, respectively. The mIoU is then calculated as the average IoU over all C classes:

$$\text{mIoU} = \frac{1}{C} \sum_{c=1}^{C} \text{IoU}_c. \tag{16}$$

Table 1
Performance comparison across semantic classes on VoD. All values are reported as percentages. Best results per column are in bold, and second-best results are underlined. Our method achieves the best mIoU.

Method	Modality	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU
PointNet++ [23]	radar	87.4	35.1	47.9	25.1	4.6	22.6	0.0	14.0	0.0	6.2	0.0	22.1
SPVCNN [59]	radar	89.3	58.2	54.4	37.0	6.2	19.8	5.6	25.1	3.5	0.5	0.7	27.3
Cylinder3D [56]	radar	88.5	50.1	50.3	33.8	7.9	13.3	3.4	17.3	2.0	24.1	2.6	26.7
PTv2 [21]	radar	**95.1**	62.9	54.8	38.7	11.9	26.6	2.3	27.5	1.0	6.0	0.3	29.7
PTv3 [20]	radar	92.2	59.0	54.0	37.9	5.2	19.1	1.6	24.4	0.0	36.9	**10.8**	30.0
2DPASS [38]	radar	89.0	52.8	51.9	34.9	8.6	20.8	3.6	26.3	0.0	19.8	1.5	28.1
RadarGNN [27]	radar	86.5	43.5	47.3	32.5	8.2	18.0	1.9	20.1	1.7	0.0	0.0	23.6
STA-Net [7]	radar	82.5	20.2	32.2	16.8	5.5	8.9	0.5	13.3	0.0	0.0	2.5	16.6
RaSS [29]	radar	89.5	46.7	50.5	30.4	10.2	16.0	5.1	24.8	0.0	60.3	0.0	30.3
PMF [46]	radar + camera	93.4	**75.1**	57.9	44.6	23.1	41.7	10.5	25.2	6.7	28.6	1.2	37.1
TASeg [1]	radar + camera	92.5	70.8	56.1	42.7	22.9	**46.5**	4.7	14.1	16.8	49.8	1.6	38.1
MinkNet [4]	radar	89.0	55.1	52.8	39.6	6.7	19.1	4.1	23.2	4.1	25.4	0.7	29.1
Ours single-model	radar	89.5	57.2	53.8	38.3	7.2	21.6	3.5	26.9	0.3	58.2	0.6	32.5
Ours multi-model	radar + camera	93.2	74.9	**61.6**	**45.4**	**30.8**	43.9	**17.2**	**36.0**	**31.4**	**65.9**	1.2	**45.6**

This metric effectively captures the overlap between the predicted and ground truth regions, and is widely used in evaluating segmentation performance, especially in autonomous driving scenarios.

4.2. Implementation details

All experiments are conducted on a single NVIDIA GeForce RTX 3090 GPU using PyTorch. We adopt the AdamW optimizer with an initial learning rate of 0.008 and a weight decay of 0.01. Gradient clipping is applied with a maximum norm of 10 (using the L2 norm). The learning rate is scheduled using a multi-step policy, where the learning rate is reduced by a factor of 0.1 at epochs 24 and 32, with training ending at epoch 36. We set the voxel size to [0.05, 0.05, 0.125] for both VoD and TJ4DRadSet, and use the official point cloud ranges $[x_{\min}, y_{\min}, z_{\min}, x_{\max}, y_{\max}, z_{\max}]$ = [0, -25.6, -3, 51.2, 25.6, 2] for VoD and [0, -40, -4, 70.4, 40, 2] for TJ4DRadSet. Besides, We feed the camera branch with the original image resolution, and downsample the input to the depth branch by a factor of 2 for efficiency. For data augmentation, we follow prior works [4,9,56] and apply three strategies, including flipping, rotation around the z-axis, and scaling. We build our code upon MMDetection3D [57], following its default training setup. Unless otherwise specified, we do not explicitly fix a random seed. To facilitate reproducibility, we provide configuration support for setting a fixed seed in our released code.

The input point cloud from VoD and TJ4DRadSet contains the following dimensions, respectively:

$$\mathbf{f}_{\mathrm{p}}^{\mathrm{VoD}} = \{x, y, z, \mathrm{RCS}, v_r, v_{rc}, \mathrm{time}\}, \tag{17}$$

$$\mathbf{f}_{\mathrm{p}}^{\mathrm{TJ4D}} = \{x, y, z, v_r, \mathrm{Range}, \mathrm{Power}, \mathrm{Alpha}, \mathrm{Beta}, v_{rc}\}, \tag{18}$$

where RCS represents the radar cross section, v_r is the relative radial Doppler velocity, v_{rc} is the absolute radial Doppler velocity, time is the time ID, Range is the detection range to radar center, Power is in dB scale and represents the signal to noise ratio, Alpha and Beta are horizontal angle and vertical angle of the radar point, respectively. Additionally, for the image branch, the camera resolution is set to 1216×1936 and the number of stages is configured as $L = 4$. In the SAD module, we employ k-nearest neighbor aggregation with $k = 2$ neighbors and set the attention sharpness parameter to $\sigma = 1$.

Through confusion matrix analysis, we observe that the TJ4DRadSet dataset suffers from a severe class imbalance problem, which limits the performance for the pedestrian and cyclist categories. In the TJ4DRadSet experiments, we replace the cross entropy loss with a weighted cross entropy loss and additionally incorporate the Lovász loss [58] to enhance the performance on minority classes. The class weights are set to [0.3, 1.0, 2.0, 2.0, 1.0]. Additionally, we calibrate the threshold of the network's output scores to improve recall.

4.3. Experiment results

4.3.1. Main results

As only very limited prior work has explored semantic segmentation on 4D radar point clouds, we select representative approaches that are originally designed for LiDAR-based tasks (e.g., MinkUNet [4], PointNet++ [23] and Point Transformer series [20,21]). Since both LiDAR and 4D radar can be represented in the point cloud modality, these architectures can be fairly adopted for our setting. *Importantly, we reproduce and retrain them under the same datasets and experimental settings as ours to ensure a fair comparison.* In addition, we also include approaches specifically designed for radar point cloud segmentation, such as RadarGNN [27] and the recent contemporaneous RaSS [29], and reproduce their recommended implementations for comparison. We further adapt TASeg [1] and STA-Net [7] to our single-frame, radar-centric protocol (e.g., using radar inputs and variants without explicit temporal modeling where necessary) to serve as strong recent baselines for fair comparsion.

Main Results on the VoD dataset. Table 1 presents the comparison results of our method. As shown, our multi-modal method achieves the highest mIoU of 45.6%, outperforming all existing approaches. Notably, our single-modal variant also surpasses all other single-modality methods, demonstrating the effectiveness of our design even without multi-modal fusion. Compared to the baseline, our method improves the mIoU by a substantial margin of 16.5 percentage points (45.6% vs. 29.1%). Among recent works, it significantly outperforms the multi-modal method TASeg by 7.5 mIoU (45.6% vs. 38.1%) under the same evaluation setting. The contemporaneous RaSS reaches 30.3 mIoU with LiDAR-to-radar distillation, still below our radar-only student (32.5 mIoU). We attribute this gap to our LiDAR + radar teacher, which reduces the modality discrepancy to the radar student, and together with KNN-based aggregation, provides effective supervision for every non-empty radar voxel. The single-frame variant of STA-Net obtains 16.6 mIoU, where we observe that it tends to overfit on sparse 4D radar point clouds on VoD. Overall, our method establishes a new state-of-the-art in radar-based semantic segmentation.

Fig. 4 presents the visualization results of point cloud segmentation on the VoD dataset. As shown, our method achieves a clear performance improvement over the single-modal baseline, particularly for categories that are challenging to distinguish in radar point clouds, such as static cars. Moreover, MKFusion demonstrates significantly better segmentation performance for distant objects. These improvements are attributed to the fine-grained semantic features provided by images and the proposed efficient fusion module. Additionally, owing to the incorporation of the depth prior, our approach exhibits superior performance in con-

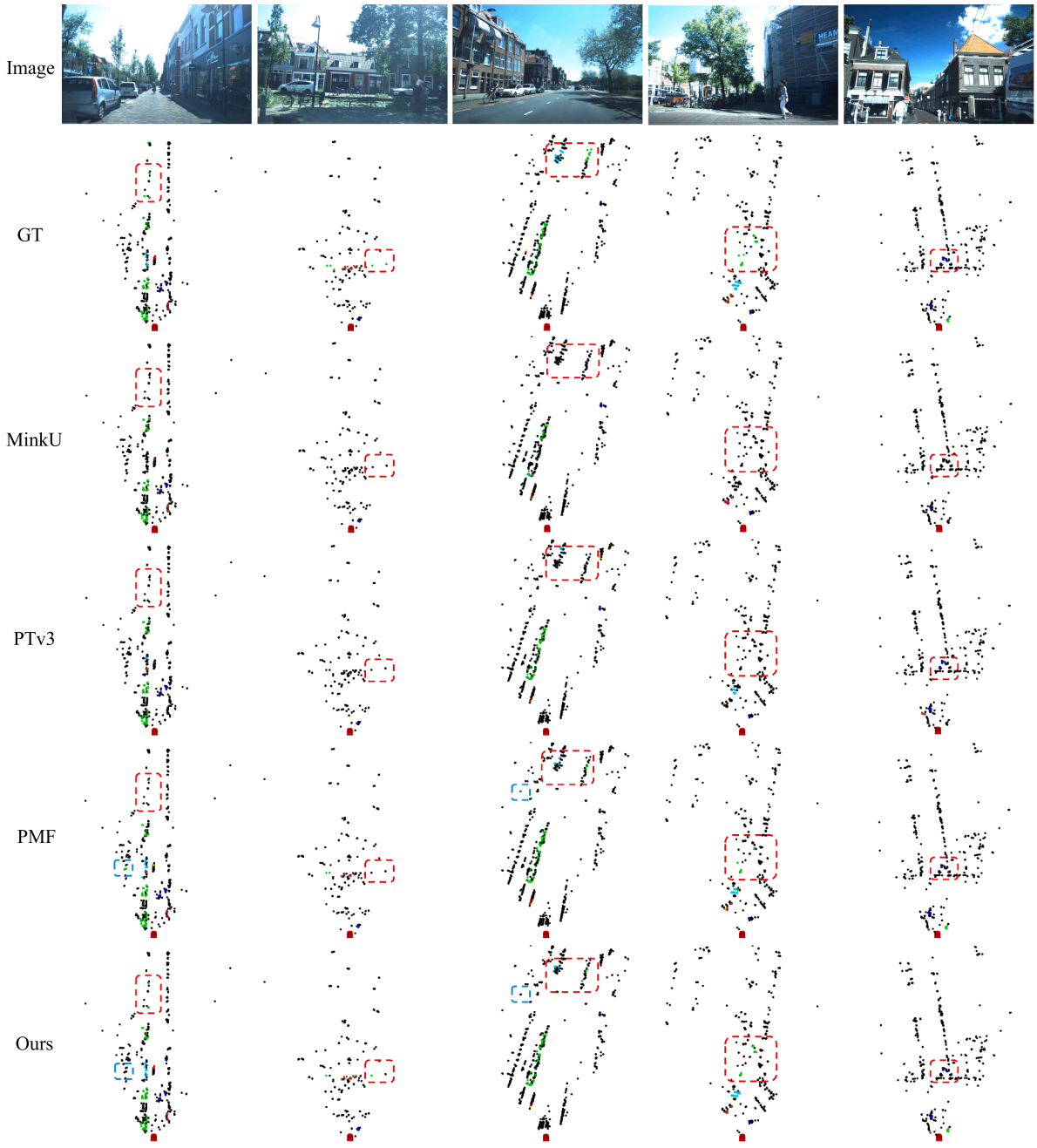


Fig. 4. Visualization results on the VoD dataset. The color for each class is as follows: ■ : background ■ : car, ■ : pedestrian, ■ : cyclist, ■ : bicycle, ■ : bicycle rack, ■ : moped scooter, ■ : rider, ■ : truck, ■ : ride other, ■ : motor. The red CAD model in the figure represents the ego vehicle. Blue boxes illustrate the false alarm cases. Best viewed in color and by zooming in.

Table 2
Performance comparison across semantic classes on TJ4DRadSet. Best results per column are in bold, and second-best results are underlined.

Method	Modality	Background	Car	Pedestrian	Cyclist	Truck	mIoU	Acc	Acc_cls
PointNet++ [23]	radar	90.5	19.4	25.3	50.9	46.4	46.5	89.1	48.5
SPVCNN [59]	radar	89.6	9.6	24.5	41.4	51.3	43.3	89.3	44.3
Cylinder3D [56]	radar	86.6	4.2	23.6	32.9	21.3	33.7	86.3	39.7
MinkUNet [4]	radar	90.2	7.7	20.2	50.6	48.0	43.3	89.2	42.8
RadarGNN [27]	radar	86.1	4.0	31.7	41.8	17.7	36.3	84.4	45.4
PTv2 [21]	radar	<u>92.7</u>	12.8	33.4	**61.5**	49.9	50.1	91.9	49.6
PTv3 [20]	radar	92.1	17.5	<u>37.7</u>	57.5	57.5	52.5	91.8	51.8
STA-Net [7]	radar	89.2	19.1	32.8	50.0	50.6	48.3	88.9	53.2
PMF [46]	radar + camera	91.9	57.6	29.7	47.7	43.8	54.1	92.2	50.2
TASeg [1]	radar + camera	**93.3**	<u>59.1</u>	33.9	54.0	**62.1**	<u>60.5</u>	<u>93.5</u>	<u>57.7</u>
Ours	radar + camera	**93.3**	**64.8**	**39.9**	<u>59.0</u>	<u>61.8</u>	**63.8**	**93.6**	**64.5**

Table 3
Ablation study of key modules on semantic segmentation performance. The combination of MSDAF and SAD achieves the best results across most metrics.

Method	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU	Acc	Acc_cls
Baseline	89.0	55.1	52.8	39.6	6.7	19.1	4.1	23.2	4.1	25.4	0.7	29.1	88.3	34.9
MSDAF	93.0	72.3	55.6	38.6	29.8	39.9	**21.3**	34.6	**43.8**	38.9	**11.3**	43.6	91.5	**54.8**
SAD	89.5	57.2	53.8	38.3	7.2	21.6	3.5	26.9	0.3	58.2	0.6	32.5	88.6	39.2
MSDAF + SAD	**93.2**	**74.9**	**61.6**	**45.4**	**30.8**	**43.9**	17.2	**36.0**	31.4	**65.9**	1.2	**45.6**	**92.2**	54.4

trolling false alarms caused by depth ambiguity (as highlighted by the blue box in the figure).

Main Results on the TJ4DRadSet dataset. Table 2 presents the semantic segmentation results of our method in comparison with several strong baselines on the TJ4DRadSet dataset. As illustrated in Table 2, our method consistently outperforms all existing approaches across multiple key metrics. In particular, it achieves the highest mIoU of 63.8%, surpassing TASeg by a margin of 3.3 percentage points (63.8% vs. 60.5%). It is important to note that TJ4DRadSet does not provide LiDAR data, making it infeasible to apply distillation-based methods that rely on LiDAR as the teacher modality. This constraint further emphasizes the practicality of our design, which operates effectively without external supervision. Compared to traditional point-based baselines such as MinkUNet and SPVCNN, our model exhibits markedly better performance on large object categories such as car and truck, with significant IoU improvements. Overall, our approach establishes a new state-of-the-art in radar-based semantic segmentation on the TJ4DRadSet benchmark.

4.3.2. Ablation study of each module

To comprehensively evaluate segmentation performance, we report both overall accuracy (Acc) and mean class accuracy (Acc_cls). Acc measures the percentage of correctly classified points across the entire scene, while Acc_cls reflects the average accuracy across all classes, providing insight into per-class performance balance. All ablation experiments are conducted on the VoD dataset.

To evaluate the effectiveness of each proposed module, we conduct ablation studies on MSDAF and SAD, as shown in Table 3. Compared to the Baseline, incorporating MSDAF alone leads to a substantial performance gain of +14.5 mIoU, highlighting the importance of depth-aware multi-modal feature integration. Similarly, introducing SAD alone improves mIoU by +3.4, demonstrating its effectiveness in enhancing feature alignment between modalities through cross-modal supervision.

When combining both MSDAF and SAD, the model achieves the highest performance across all metrics, including mIoU (45.6%), overall accuracy (92.2%), and class-wise accuracy (54.4%). Notably, the full model brings significant improvements on challenging dynamic object classes such as Pedestrian (+8.8), Cyclist (+5.8), and Truck (+40.5), indicating strong modeling capacity for fine-grained and moving objects. These results confirm that MSDAF and SAD are complementary and jointly contribute to the final performance.

4.3.3. Effect of depth estimation network

Fig. 5 provides an example of visualization to demonstrate the role of the depth estimation network. As observed, point clouds located at different distances but sharing the same azimuth are projected onto the same image region due to projection ambiguity, resulting in potential false alarms (see Fig. 5 (c)). With the incorporation of the depth estimation network, this phenomenon is effectively mitigated.

Quantitative experimental results are presented in Table 4. Overall, when depth information is not utilized, categories dominated by static objects, such as bicycle and bicycle rack, exhibit performance degradation, whereas categories with a higher proportion of dynamic objects, such as pedestrian and cyclist, show relatively minor changes. We attribute this to the fact that dynamic objects can be more easily distinguished from radar features, where the contribution of depth is limited. In contrast, static objects are more difficult to resolve using radar alone, and the network can leverage depth information to better align with radar points. To further assess the choice of depth backbone, we also evaluate DepthAnything-V2 [60] (DAv2) as a drop-in replacement, considering both its relative and metric variants (Table 4). Under the same fusion interface, the relative DAv2 variant slightly underperforms the no-depth baseline, and the metric DAv2 variant only matches but does not surpass Metric3D-V2 [54]. This is consistent with their design goals: the base DAv2 predicts scale-ambiguous depth, making alignment with absolute radar geometry more difficult, and the metric variant can carry domain-specific biases, whereas Metric3D-V2 explicitly targets camera-model ambiguity and provides more stable metric depth for our radar-centric fusion.

4.3.4. Effect of k in feature aggregation

To analyze the influence of the neighborhood size k in our KNN-based distillation module, we conduct experiments with different values of $k = \{1, 2, 3, 4\}$. As shown in Table 5, the mIoU first increases with larger k, reaching a peak at $k = 2$, and then declines as k continues to grow. Specifically, using $k = 2$ achieves the best performance with an mIoU of 32.5%, suggesting that it strikes a balance between local context aggregation and noise sensitivity. In contrast, too small ($k = 1$) or too large ($k = 4$) neighborhoods lead to underfitting or over-smoothing, respectively. Therefore, we adopt $k = 2$ as the default setting in all subsequent experiments.

4.3.5. Distillation type

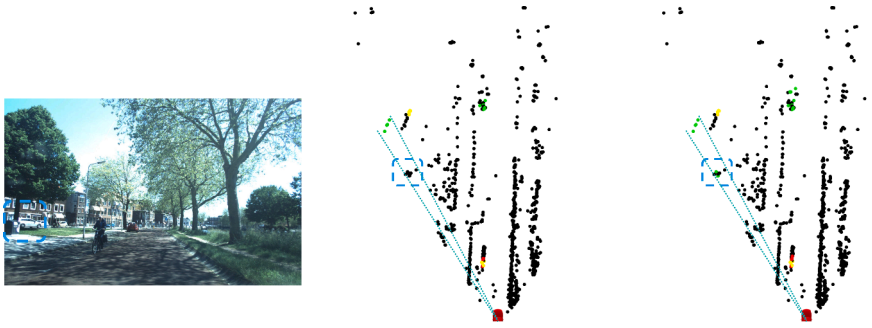
To evaluate the effectiveness of different distillation objectives, we conduct ablation experiments using a variety of loss functions, including L1, L2, KL divergence, cosine distance, affinity loss, and attention transfer loss. These results are obtained under the distillation setting with $k = 2$ nearest neighbors.

As shown in Table 6, while traditional losses such as L2 and KL divergence yield moderate improvements over the baseline, they fall short in capturing the fine-grained semantic structures that are critical for radar-based segmentation. Notably, L1 loss exhibits more stable performance across most classes, particularly in the pedestrian and rider categories, which are especially susceptible to noise and sparsity in radar signals. Furthermore, the combination of Cosine Distance Loss with L1 loss achieves the best overall performance, attaining the highest mIoU (32.5), overall accuracy (88.6), and class-average accuracy (39.2).

These findings demonstrate that the robustness of L1 loss to outliers and the structural regularization capability of Cosine Distance Loss complement each other, resulting in superior performance for fine-grained radar-camera segmentation.

4.3.6. Fusion type

We compare different fusion strategies to assess their impact on semantic segmentation performance, including direct concat, adaptive weighted fusion, and cross attention. As shown in Table 7, our proposed multimodal fusion method outperforms these classic fusion approaches, achieving the best overall performance. Notably, our method is built upon adaptive weighted fusion, with targeted improvements that effectively mitigate the modality gap and depth ambiguity. Extensive experiments demonstrate that addressing these issues leads to significant improvements in segmentation performance.



(a) Image (b) with depth (c) without depth

Fig. 5. Visual explanation of the depth estimation network's influence on segmentation. Blue boxes illustrate the false alarm cases.

Table 4
Effectiveness of depth estimation network.

Method	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU	Acc	Acc_cls
w/o depth	93.0	74.6	61.6	46.2	29.1	41.4	20.6	34.6	22.7	66.1	0.8	44.6	92.1	53.0
w/ DepthAnything-V2 (relative)	93.4	76.2	61.0	46.2	26.0	39.2	20.1	29.3	23.7	62.6	4.5	43.8	92.0	53.7
w/ DepthAnything-V2 (metric)	93.1	75.4	61.5	45.9	29.8	41.5	19.4	34.8	24.4	66.1	8.1	44.8	92.2	53.5
w/ Metric3D-V2	93.2	74.9	61.6	45.4	30.8	43.9	17.2	36.0	31.4	65.9	1.2	45.6	92.2	54.4

Table 5
Performance comparison under different k values for KNN feature aggregation in the distillation module.

k	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU	Acc	Acc_cls
1	89.3	53.3	48.7	37.5	7.3	**29.5**	1.1	23.7	0.0	57.2	0.0	31.6	88.3	38.4
2	**89.5**	**57.2**	**53.8**	**38.3**	7.2	21.6	3.5	**26.9**	**0.3**	**58.2**	0.6	**32.5**	**88.6**	**39.2**
3	89.1	53.3	51.3	37.0	12.0	18.1	**7.0**	22.8	0.0	57.9	**0.7**	31.7	88.2	39.1
4	89.3	54.9	49.7	33.4	**15.8**	18.6	1.7	25.9	0.0	45.3	0.3	30.4	88.2	36.7

Table 6
Comparison of different distillation losses.

Distillation Type	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU	Acc	Acc_cls
Baseline	89.0	55.1	52.8	**39.6**	6.7	19.1	4.1	23.2	**4.1**	25.4	0.7	29.1	88.3	34.9
L2 Loss	89.4	56.4	53.4	36.8	8.6	20.7	4.2	26.9	2.1	15.4	0.5	28.6	88.5	35.0
L1 Loss	89.1	55.1	53.0	37.1	6.8	18.4	1.3	26.0	0.4	34.6	1.3	29.4	88.2	35.2
KL Loss	89.4	57.5	**56.3**	38.7	7.5	18.1	**4.5**	26.2	0.0	15.1	**3.8**	28.8	88.5	35.3
Cosine Distance Loss	89.4	**57.8**	52.9	37.3	7.6	20.7	4.0	**29.8**	1.0	43.8	0.5	31.3	88.6	38.1
Affinity Loss	**89.7**	56.1	51.2	37.6	**10.1**	21.1	2.5	26.4	0.0	20.1	0.3	28.6	88.5	34.9
Attention Transfer Loss	89.4	56.5	52.0	36.7	7.2	21.5	2.4	27.6	0.5	22.5	0.0	28.8	88.4	34.8
Ours	89.5	57.2	53.8	38.3	7.2	**21.6**	3.5	26.9	0.3	**58.2**	0.6	**32.5**	**88.6**	**39.2**

Table 7
Comparison of different fusion types.

Fusion Type	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU	Acc	Acc_cls
Directly Concat	92.2	69.9	55.7	40.6	20.2	32.2	12.0	20.8	16.8	30.8	2.0	35.7	90.7	44.7
Adaptive Weighted Fusion	92.3	68.7	52.9	35.5	26.7	42.4	12.8	30.7	0.9	**45.1**	0.5	37.1	91.2	44.7
Cross Attention	92.8	**72.9**	**61.6**	**41.2**	28.8	36.0	14.5	31.9	0.0	32.4	0.3	37.5	91.5	46.3
Ours	**93.0**	72.3	55.6	38.6	**29.8**	**42.4**	**21.3**	**34.6**	**43.8**	38.9	**11.3**	**43.6**	**91.5**	**54.8**

4.3.7. Effect of fusion and distillation position

We conduct ablation studies on different positions for feature fusion and distillation. Here, xconv$i$ denotes the ith encoder layer (from shallow to deep), while xdeconv$j$ refers to the jth decoder layer.

As shown in Table 8, for feature fusion, the shallow encoder layer xconv1 yields the best performance, likely due to its fine-grained spatial resolution and sufficient cross-modal interaction. In contrast, xconv4 performs worse due to its coarse granularity, and xdeconv4 also underperforms due to limited feature interaction at the decoder stage.

For knowledge distillation, we observe that distillation in the decoder generally achieves better performance than that in the encoder. Within the encoder, increasing the number of feature layers used leads to further performance improvements, whereas this effect is less pronounced in the decoder. This is because radar point clouds are extremely sparse and noisy, causing the radar network to extract far less informative features in its early layers compared to the LiDAR network. Deeper layers contain more semantically rich features, making them more suitable targets for effective knowledge transfer. We adopt xdeconv4 for distillation, which balances performance and training efficiency.

4.3.8. Modality ablation

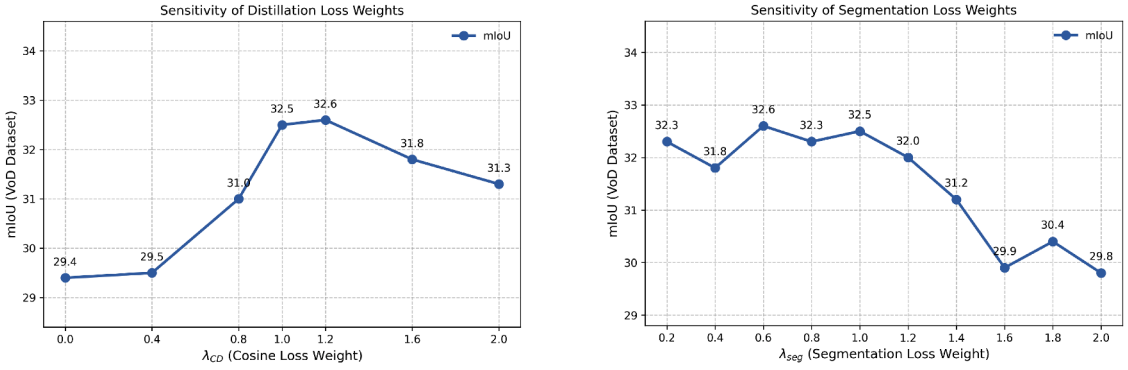
Table 9 reports a modality ablation using radar (R), camera (C), and LiDAR (L), where "Eval. Mod." stands for "Evaluation Modality" and indicates the modality in which metrics are computed. Using the same radar backbone, distillation from L+R to R improves the radar-only model from 29.1 to 32.5 mIoU and from 34.9 to 39.2 Acc_cls, while

Table 8
Comparison of different fusion and distillation positions.

	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU	Acc	Acc_cls
Fusion Pos.														
xconv1	**93.0**	**72.3**	55.6	38.6	**29.8**	**39.9**	**21.3**	**34.6**	**43.8**	38.9	**11.3**	**43.6**	**91.5**	**54.8**
xconv4	92.1	69.9	54.9	**42.2**	20.1	34.5	8.4	29.8	26.3	**60.9**	0.2	39.9	90.9	47.9
xdeconv4	92.4	67.0	**58.8**	40.9	25.3	**39.9**	9.6	29.7	29.9	50.7	5.0	40.8	91.0	53.1
Distillation Pos.														
xconv1	88.7	53.6	50.7	33.5	4.6	22.7	0.2	20.8	0.0	53.1	0.0	29.8	88.1	35.9
xconv4	89.2	56.3	49.7	33.4	5.7	25.5	1.4	20.5	1.3	46.9	**3.1**	30.3	88.3	37.1
xconv{1234}	**89.5**	55.0	48.7	34.8	6.7	**29.4**	3.4	25.2	0.0	57.1	0.6	31.8	88.4	39.5
xdeconv3	89.4	51.1	50.0	31.0	7.1	24.2	**4.7**	**31.0**	4.1	54.5	0.2	31.5	88.1	38.4
xdeconv4	**89.5**	57.2	**53.8**	**38.3**	7.2	21.6	3.5	26.9	0.3	58.2	0.6	**32.5**	**88.6**	39.2
xdeconv{1234}	89.2	56.2	51.3	27.2	9.4	25.4	4.4	26.9	0.1	**59.6**	0.8	31.9	88.2	**41.4**
all	89.4	**57.8**	52.0	38.0	**8.9**	19.8	2.8	28.0	**11.7**	48.3	0.0	32.4	88.5	39.7

Table 9
Modality ablation.

Method	Eval. Mod.	Background	Car	Pedestrian	Cyclist	Bicycle	Bicycle rack	Moped scooter	Rider	Motor	Truck	Ride other	mIoU	Acc	Acc_cls
R (w/o KD)	R	89.0	55.1	52.8	39.6	6.7	19.1	4.1	23.2	4.1	25.4	0.7	29.1	88.3	34.9
R (w/ KD)	R	89.5	57.2	53.8	38.3	7.2	21.6	3.5	26.9	0.3	58.2	0.6	32.5	88.6	39.2
R + C	R	93.2	74.9	61.6	45.4	30.8	43.9	17.2	36.0	31.4	65.9	1.2	45.6	92.2	54.4
L	L	96.8	76.0	79.2	38.4	23.3	55.7	24.6	42.8	32.9	37.5	4.2	46.5	95.5	58.9
Teacher (L + R)	R	94.0	72.0	69.4	51.8	32.5	45.9	21.0	40.0	34.6	17.4	0.1	43.6	92.6	54.1



(a) Sensitivity of distillation loss weights (b) Sensitivity of segmentation loss weights

Fig. 6. Sensitivity of loss weights.

keeping inference cost unchanged. Gains are especially notable for large objects such as truck and car, although some tail classes remain unstable under sparse returns. With the proposed MSDAF fusion, the R+C model further boosts performance to 45.6 mIoU, nearly closing the gap to the LiDAR reference (46.5 mIoU) and substantially improving almost all classes compared to distilled radar-only. For context, we also report a non-deployable teacher (L+R) evaluated in radar space (43.6 mIoU), which illustrates why distillation is effective: the fused teacher injects dense LiDAR geometry at radar locations and provides supervision for all non-empty radar voxels. Overall, MKFusion narrows the LiDAR-radar gap from 17.4 mIoU with the radar baseline to 14.0 mIoU with distillation and to 0.9 mIoU with radar+camera fusion, while preserving a deployable sensor suite.

4.3.9. Sensitivity analysis of loss weights

λ_{L1} and λ_{CD}:

Fig. 6 (a) presents the parameter sensitivity analysis of λ_{L1} and λ_{CD}. The horizontal axis denotes λ_{CD}, while the vertical axis shows the mIoU on the VoD dataset, with $\lambda_{L1} = 2 - \lambda_{CD}$. As illustrated, using only L1 loss or only cosine loss results in performance degradation, whereas combining the two losses leads to performance improvement. The best performance is achieved around $\lambda_{CD} = 1.2$. This indicates that L1 loss and cosine loss provide complementary information for distilling LiDAR features into radar features. Additionally, cosine loss plays a stronger role than L1 loss.

λ_{seg} and λ_{RD}:

Similarly, Fig. 6 (b) presents the parameter sensitivity analysis of λ_{seg} and λ_{RD}. λ_{seg} ranges from 0.2 to 2.0, with $\lambda_{RD} = 2 - \lambda_{seg}$. As shown, a smaller segmentation loss weight leads to better performance, whereas an excessively large segmentation loss weight results in a noticeable performance drop. This observation indicates the LiDAR teacher derived from high-fidelity geometry, can provide stable semantic and boundary priors. When the segmentation loss weight becomes too large, the learning process tends to overfit radar-specific noise, thereby weakening the geometric and semantic priors provided by distillation.

4.3.10. Computational cost

Table 10 summarizes computational cost under identical hardware, input resolution, and single-frame protocol. With MinkUNet as the radar student, our LiDAR-radar distillation raises mIoU from 29.1 to 32.5 without changing parameters (21.7M) or latency (20.5 ms / 48.8 FPS), since the teacher is used only during training. In the multi-modal setting, our Radar+Camera variant without depth prior offers a favorable

Table 10
Computational cost.

Method	Modality	Params. (M)	Latency (ms)	FLOPs (G)	FPS	mIoU
MinkUNet [4]	radar	**21.7**	**20.5**	**1.37**	**48.8**	29.1
PTv3 [20]	radar	46.2	45.3	43.23	22.1	30.0
PMF [46]	radar + camera	49.2	68.9	329.35	14.5	37.1
TASeg [1]	radar + camera	55.6	94.2	329.92	10.6	38.1
Ours	radar	**21.7**	**20.5**	**1.37**	**48.8**	32.5
Ours (w/o depth)	radar + camera	49.1	68.7	329.59	14.6	44.6
Ours (w/ depth)	radar + camera	84.1	118.6	982.50	8.4	**45.6**

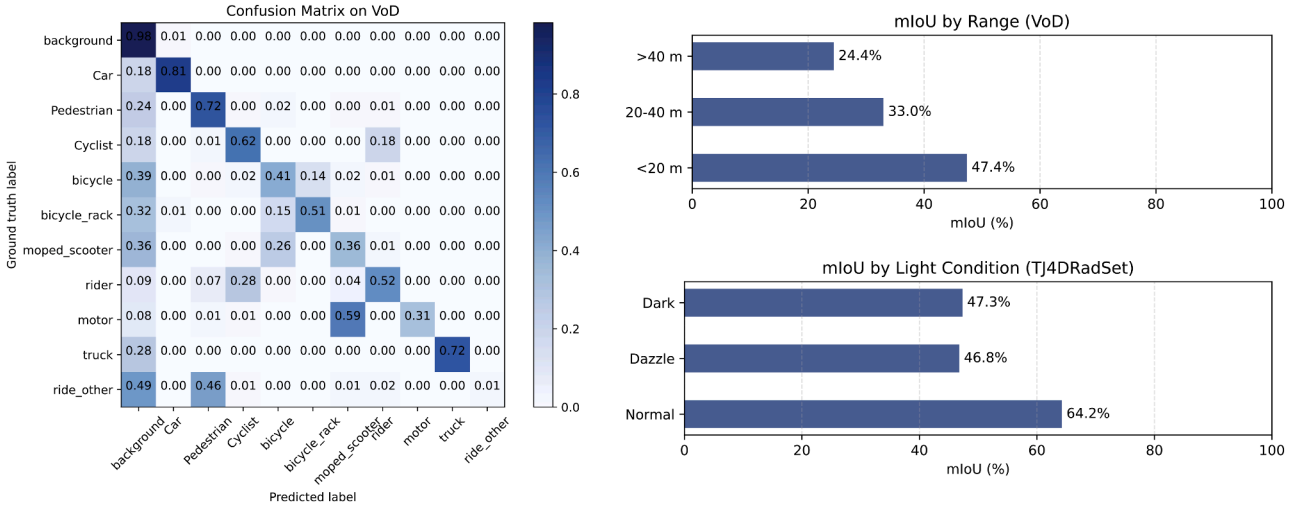


Fig. 7. Confusion matrix and condition-wise mIoU analysis on VoD and TJ4DRadSet.

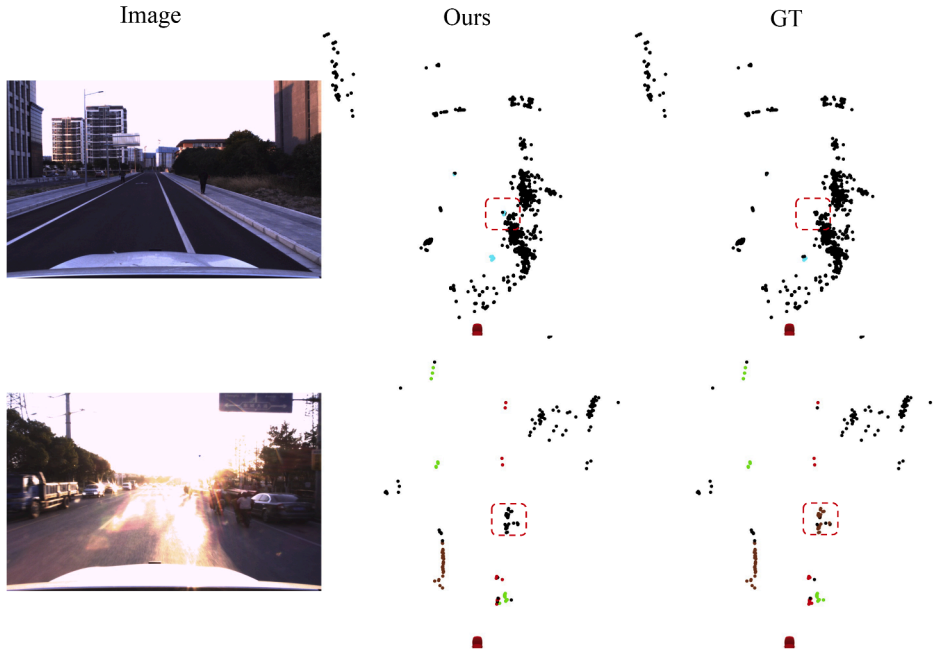


Fig. 8. Visualizations of the failure cases on TJ4DRadSet. The color for each class is as follows: ■: background ■: car, ■: pedestrian, ■: cyclist, ■: truck. The red CAD model in the figure represents the ego vehicle. Red boxes illustrate the failure cases. Best viewed in color and by zooming in.

accuracy-efficiency trade-off, outperforming PMF and TASeg in mIoU (44.6 vs 37.1 / 38.1) at comparable or lower cost. The depth-augmented Radar + Camera model attains the best accuracy (45.6 mIoU) with increased latency and parameters, while PTv3 serves as a heavier radar backbone that roughly doubles latency and model size but brings limited benefit on sparse VoD point clouds. We also report GFLOPs in Table 10. It can be seen that, in multi-modal networks, the camera branch dominates FLOPs due to dense image computation while the sparse radar branch remains lightweight. Nevertheless, the latency remains practical in our measurements due to highly optimized GPU kernels for common image operators, e.g., convolutions and attention. It is important to note that GFLOPs are only a coarse proxy, and the end-to-end runtime also depends on factors such as preprocessing, memory bandwidth, and kernel-launch/parallelism effects.

4.3.11. Discussion

Failure cases and future work. We analyze typical failure modes using the multimodal variant with depth priors (Figs. 7-8). On VoD dataset, errors arise mainly in small, fine-grained wheeled classes (e.g., bicycle, bicycle-rack, motor, rider/moped) with mutual confusion and leakage to background. Additionally, the mIoU shows a clear decay with distance, reflecting sparsity and low signal-to-noise ratio of long-range radar returns. On TJ4DRadSet dataset, illumination-wise mIoU drops under dazzle and low-light conditions, and qualitative examples reveal false positives caused by multipath/penetration "ghosts" and missed detections under severe glare. These observations suggest three promising directions: incorporating temporal cues (e.g., egomotion-aligned accumulation and Doppler-guided warping), exploiting richer RAD or even raw radar representations under practical computation budgets, and scaling/diversifying 4D radar datasets to better cover adverse weather, nighttime, even indoor, and roadside [61].

Practical deployment and sensor cost. MKFusion is designed as a radar-centric semantic perception module rather than a full highway-urban autonomy stack. In practice, a development fleet can be equipped with 4D radar, camera, and LiDAR, where LiDAR is used offline to supervise MKFusion via distillation, while deployed platforms run the student with radar-only or radar + camera without requiring LiDAR or any online teacher. This keeps the production sensor suite close to existing radar-camera advanced driver-assistance systems (ADAS) and avoids recurring per-vehicle cost, power, and thermal overhead from LiDAR. Such radar-first configurations are especially attractive in cost-sensitive domains where radar's robustness and long-range Doppler cues are most relevant, such as campuses and industrial parks, parking and indoor garages, roadside/vehicle-to-everything (V2X) perception, heavy-truck blind-spot coverage, and far-range highway awareness. In these scenarios, MKFusion's per-point predictions can be further aggregated into occupancy or free-space cues and combined with existing image-based and planning modules to support higher-level functions. Moreover, our approach can also benefit from future advances in 4D imaging radar hardware, such as improved angular resolution, elevation fidelity, Doppler quality, and frame rate.

Limitations. We note that our segmentation labels are derived from the bounding-box annotations provided by the dataset, and may therefore inherit noise from the original detection labels. In addition, fine-grained annotations for background elements such as buildings, sidewalks, and traffic signs are still lacking. Moreover, our SAD module relies on LiDAR data to obtain the teacher model, which limits its applicability to privacy-sensitive scenarios or datasets without LiDAR sensors, such as TJ4DRadSet. Leveraging camera semantic information to obtain finer-grained segmentation labels for 4D radar point clouds, as well as incorporating temporal cues for self-distillation, are promising directions for future work.

5. Conclusion

In this work, we present MKFusion, a novel framework for 4D radar point cloud semantic segmentation that effectively combines multi-modal fusion and knowledge distillation within a sparse architecture. By leveraging camera features through multi-scale depth-aware fusion and transferring structural knowledge from LiDAR via our sparse-aligned distillation module, MKFusion addresses the challenges of radar sparsity and noise. Extensive experiments on the VoD and TJ4DRadSet dataset demonstrate that our approach achieves superior segmentation performance and establishes new state-of-the-art results. These findings validate the effectiveness of integrating cross-modal cues and distillation techniques in enhancing the reliability of radar-based perception for autonomous driving.

CRediT authorship contribution statement

Yunting Yang: Writing – original draft, Methodology, Investigation, Formal analysis; **Jun Liu:** Writing – review & editing, Resources, Methodology, Funding acquisition, Conceptualization; **Hongsi Liu:** Writing – review & editing, Visualization, Validation, Data curation; **Guangfeng Jiang:** Writing – review & editing, Validation, Investigation.

Data availability

The authors do not have permission to share data.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Contract 62471450, and the Natural Science Foundation of Anhui Province under Grant 2208085J17.

References

[1] X. Wu, Y. Hou, X. Huang, B. Lin, T. He, X. Zhu, Y. Ma, B. Wu, H. Liu, D. Cai, et al., TASeg: temporal aggregation network for LiDAR semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 15311–15320. https://doi.org/10.1109/CVPR52733.2024.01450

[2] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, H. Li, End-to-end autonomous driving: challenges and frontiers, IEEE Trans. Pattern Anal. Mach. Intell. 46 (12) (2024) 10164–10183. https://doi.org/10.1109/TPAMI.2024.3435937

[3] A. Milioto, I. Vizzo, J. Behley, C. Stachniss, RangeNet + +: fast and accurate LiDAR semantic segmentation, in: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2019, pp. 4213–4220. https://doi.org/10.1109/IROS40897.2019.8967762

[4] C. Choy, J. Gwak, S. Savarese, 4D spatio-temporal ConvNets: minkowski convolutional neural networks, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 3070–3079. https://doi.org/10.1109/CVPR.2019.00319

[5] Y. Yang, J. Liu, H. Liu, G. Jiang, Radar M3-Net: multi-scale, multi-layer, multi-frame network with a large receptive field for 3D object detection, Expert Syst. Appl. 286 (C) (2025) 127515. https://doi.org/10.1016/j.eswa.2025.127515

[6] M. Dreissig, D. Scheuble, F. Piewak, J. Boedecker, Survey on LiDAR perception in adverse weather conditions, in: 2023 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2023, pp. 1–8. https://doi.org/10.1109/IV55152.2023.10186539

[7] Z. Zhang, J. Liu, G. Jiang, Spatial and temporal awareness network for semantic segmentation on automotive radar point cloud, IEEE Trans. Intell. Veh. 9 (2) (2024) 3520–3530. https://doi.org/10.1109/TIV.2023.3347692

[8] J. Karangwa, J. Liu, Z. Zeng, Vehicle detection for autonomous driving: a review of algorithms and datasets, IEEE Trans. Intell. Transp. Syst. 24 (11) (2023) 11568–11594. https://doi.org/10.1109/TITS.2023.3292278

[9] H. Liu, J. Liu, G. Jiang, X. Jin, 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 (6) (2025) 8641–8656. https://doi.org/10.1109/TITS.2025.3554313

[10] W. Xiong, J. Liu, T. Huang, Q.-L. Han, Y. Xia, B. Zhu, LXL: LiDAR excluded lean 3D object detection with 4D imaging radar and camera fusion, IEEE Trans. Intell. Veh. 9 (1) (2023) 79–92. https://doi.org/10.1109/TIV.2023.3321240

[11] G. Jiang, J. Liu, Y. Wu, W. Liao, T. He, P. Peng, MWSIS: multimodal weakly supervised instance segmentation with 2D box annotations for autonomous driving, in: Proceedings of the AAAI Conference on Artificial Intelligence, 38, 2024, pp. 2507–2515. https://doi.org/10.1609/aaai.v38i3.28027

[12] Y. Lu, B. Jiang, N. Liu, Y. Li, J. Chen, Y. Zhang, Z. Wan, CrossPrune: cooperative pruning for camera–LiDAR fused perception models of autonomous driving, Knowl. Based Syst. 289 (2024) 111522. https://doi.org/10.1016/j.knosys.2024.111522

[13] Y. Wan, P. Lv, L. Sun, Y. Yang, J. Hao, Bi-Interfusion: a bidirectional cross-fusion framework with semantic-guided transformers in LiDAR-camera fusion, Knowl. Based Syst. 305 (2024) 112577. https://doi.org/10.1016/j.knosys.2024.112577

[14] L. Zhao, J. Song, K.A. Skinner, CRKD: enhanced camera-radar object detection with cross-modality knowledge distillation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 15470–15480. https://doi.org/10.1109/CVPR52733.2024.01465

[15] R. Xu, Z. Xiang, C. Zhang, H. Zhong, X. Zhao, R. Dang, P. Xu, T. Pu, E. Liu, SCKD: semi-supervised cross-modality knowledge distillation for 4D radar object detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, 39, 2025, pp. 8933–8941. https://doi.org/10.1609/aaai.v39i9.32966

[16] Y. Ma, J. Mei, X. Yang, L. Wen, W. Xu, J. Zhang, X. Zuo, B. Shi, Y. Liu, LiCROcc: teach radar for accurate semantic occupancy prediction using LiDAR and camera, IEEE Rob. Autom. Lett. 10 (1) (2025) 852–859. https://doi.org/10.1109/LRA.2024.3511427
[17] L. Fan, F. Wang, N. Wang, Z. Zhang, FSD V2: improving fully sparse 3D object detection with virtual voxels, IEEE Trans. Pattern Anal. Mach. Intell. 47 (2) (2025) 1279–1292. https://doi.org/10.1109/TPAMI.2024.3502456
[18] C. Xu, B. Wu, Z. Wang, W. Zhan, P. Vajda, K. Keutzer, M. Tomizuka, SqueezeSegV3: spatially-adaptive convolution for efficient point-cloud segmentation, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16, Springer, 2020, pp. 1–19. https://doi.org/10.1007/978-3-030-58604-1_1
[19] Y. Zhang, Z. Zhou, P. David, X. Yue, Z. Xi, B. Gong, H. Foroosh, PolarNet: an improved grid representation for online LiDAR point clouds semantic segmentation, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 9598–9607. https://doi.org/10.1109/CVPR42600.2020.00962
[20] X. Wu, L. Jiang, P.-S. Wang, Z. Liu, X. Liu, Y. Qiao, W. Ouyang, T. He, H. Zhao, Point transformer v3: simpler faster stronger, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 4840–4851. https://doi.org/10.1109/CVPR52733.2024.00463
[21] X. Wu, Y. Lao, L. Jiang, X. Liu, H. Zhao, Point transformer v2: grouped vector attention and partition-based pooling, Adv. Neural Inf. Process. Syst. 35 (2022) 33330–33342. https://doi.org/10.5555/3600270.3602685
[22] H. Zhao, L. Jiang, J. Jia, P.H.S. Torr, V. Koltun, Point transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 16259–16268. https://doi.org/10.1109/ICCV48922.2021.01595
[23] C.R. Qi, L. Yi, H. Su, L.J. Guibas, PointNet++: deep hierarchical feature learning on point sets in a metric space, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, p. 5105-5114. https://doi.org/10.5555/3295222.3295263
[24] C.R. Qi, H. Su, K. Mo, L.J. Guibas, PointNet: deep learning on point sets for 3D classification and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660. https://doi.org/10.1109/CVPR.2017.16
[25] M. Zeller, J. Behley, M. Heidingsfeld, C. Stachniss, Gaussian radar transformer for semantic segmentation in noisy radar data, IEEE Rob. Autom. Lett. 8 (1) (2022) 344–351. https://doi.org/10.1109/LRA.2022.3226030
[26] O. Schumann, M. Hahn, J. Dickmann, C. Wöhler, Semantic segmentation on radar point clouds, in: 2018 21st International Conference on Information Fusion (FUSION), IEEE, 2018, pp. 2179–2186. https://doi.org/10.23919/ICIF.2018.8455344
[27] F. Fent, P. Bauerschmidt, M. Lienkamp, RadarGNN: transformation invariant graph neural network for radar-based perception, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 182–191. https://doi.org/10.1109/CVPRW59228.2023.00023
[28] O. Schumann, M. Hahn, N. Scheiner, F. Weishaupt, J.F. Tilly, J. Dickmann, C. Wöhler, RadarScenes: a real-world radar point cloud data set for automotive applications, in: 2021 IEEE 24th International Conference on Information Fusion (FUSION), IEEE, 2021, pp. 1–8. https://doi.org/10.23919/FUSION49465.2021.9627037
[29] C. Zhang, Z. Xiang, R. Xu, H. Shan, X. Zhao, R. Dang, RaSS: 4D mm-Wave radar point cloud semantic segmentation with cross-modal knowledge distillation, Sensors 25 (17) (2025) 5345. https://doi.org/10.3390/s25175345
[30] Y. Zhang, L. Zhang, P. Pi, T. Li, Y. Chen, S. Peng, Z. Ma, TARSS-Net: temporal-aware radar semantic segmentation network, Adv. Neural Inf. Process. Syst. 37 (2024) 4906–4933. https://doi.org/10.5555/3737916.3738075
[31] Y. Dalbah, J. Lahoud, H. Cholakkal, TransRadar: adaptive-directional transformer for real-time multi-view radar semantic segmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 353–362. https://doi.org/10.1109/WACV57701.2024.00042
[32] L. Yang, L. Zheng, W. Ai, M. Liu, S. Li, Q. Lin, S. Yan, J. Bai, Z. Ma, X. Zhu, MetaOcc: surround-view 4D radar and camera fusion framework for 3D occupancy prediction with dual training strategies, (2025). arXiv:2501.15384
[33] F. Ding, X. Wen, Y. Zhu, Y. Li, C.X. Lu, RadarOcc: robust 3d occupancy prediction with 4D imaging radar, Adv. Neural Inf. Process. Syst. 37 (2024) 101589–101617. https://doi.org/10.5555/3737916.3741138
[34] S. Yao, R. Guan, Z. Wu, Y. Ni, Z. Huang, R.W. Liu, Y. Yue, W. Ding, E.G. Lim, H. Seo, K.L. Man, J. Ma, X. Zhu, Y. Yue, WaterScenes: a multi-task 4D radar-camera fusion dataset and benchmarks for autonomous driving on water surfaces, IEEE Trans. Intell. Transp. Syst. 25 (11) (2024) 16584–16598. https://doi.org/10.1109/TITS.2024.3415772
[35] J. Schramm, N. Vödisch, K. Petek, B.R. Kiran, S. Yogamani, W. Burgard, A. Valada, BEVCar: camera-radar fusion for bev map and object segmentation, in: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2024, pp. 1435–1442. https://doi.org/10.1109/IROS58592.2024.10802147
[36] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D.L. Rus, S. Han, BEVFusion: multi-task multi-sensor fusion with unified bird's-eye view representation, in: 2023 IEEE International Conference on Robotics and Automation (ICRA), 2023, pp. 2774–2781. https://doi.org/10.1109/ICRA48891.2023.10160968
[37] H. Zhao, R. Guan, T. Wu, K.L. Man, L. Yu, Y. Yue, UniBEVFusion: unified radar-vision BEVFusion for 3D object detection, (2024). arXiv:2409.14751
[38] X. Yan, J. Gao, C. Zheng, C. Zheng, R. Zhang, S. Cui, Z. Li, 2DPass: 2D priors assisted semantic segmentation on LiDAR point clouds, in: European Conference on Computer Vision, Springer, 2022, pp. 677–695. https://doi.org/10.1007/978-3-031-19815-1_39
[39] X. Bai, Z. Yu, L. Zheng, X. Zhang, Z. Zhou, X. Zhang, F. Wang, J. Bai, H.-L. Shen, SGDet3D: semantics and geometry fusion for 3D object detection using 4D radar and camera, IEEE Rob. Autom. Lett. 10 (1) (2025) 828–835. https://doi.org/10.1109/LRA.2024.3513041
[40] H. Zhong, Z. Xiang, R. Xu, J. Fu, P. Xu, S. Wang, Z. Yang, T. Pu, E. Liu, CVFusion: cross-view fusion of 4D radar and camera for 3D object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2025, pp. 28188–28197.
[41] Z. Lin, Z. Liu, Z. Xia, X. Wang, Y. Wang, S. Qi, Y. Dong, N. Dong, L. Zhang, C. Zhu, 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, 2024, pp. 14928–14937. https://doi.org/10.1109/CVPR52733.2024.01414
[42] L. Zheng, J. Liu, R. Guan, L. Yang, S. Lu, Y. Li, X. Bai, J. Bai, Z. Ma, H.-L. Shen, et al., Doracamom: joint 3D detection and occupancy prediction with multi-view 4D radars and cameras for omnidirectional perception, (2025). arXiv:2501.15394
[43] W. Xiong, Z. Zou, Q. Zhao, F. He, B. Zhu, LXLv2: enhanced LiDAR excluded lean 3D object detection with fusion of 4D radar and camera, IEEE Rob. Autom. Lett. 10 (3) (2025) 2862–2869. https://doi.org/10.1109/LRA.2025.3536840
[44] J. Zhang, Y. Ding, Z. Liu, Occfusion: depth estimation free multi-sensor fusion for 3d occupancy prediction, in: Proceedings of the Asian Conference on Computer Vision, 2024, pp. 3587–3604. https://doi.org/10.1007/978-981-96-0972-7_14
[45] X. Wang, Z. Zhu, W. Xu, Y. Zhang, Y. Wei, X. Chi, Y. Ye, D. Du, J. Lu, X. Wang, OpenOccupancy: a large scale benchmark for surrounding semantic occupancy perception, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 17850–17859. https://doi.org/10.1109/ICCV51070.2023.01636
[46] Z. Zhuang, R. Li, K. Jia, Q. Wang, Y. Li, M. Tan, Perception-aware multi-sensor fusion for 3D LiDAR semantic segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 16280–16290. https://doi.org/10.1109/ICCV48922.2021.01597
[47] Y. Liu, R. Chen, X. Li, L. Kong, Y. Yang, Z. Xia, Y. Bai, X. Zhu, Y. Ma, Y. Li, et al., UniSeg: a unified multi-modal LiDAR segmentation network and the openpcseg codebase, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 21662–21673. https://doi.org/10.1109/ICCV51070.2023.01980
[48] A. Palffy, E. Pool, S. Baratam, J.F.P. Kooij, D.M. Gavrila, Multi-class road user detection with 3+ 1d radar in the view-of-delft dataset, IEEE Rob. Autom. Lett. 7 (2) (2022) 4961–4968. https://doi.org/10.1109/LRA.2022.3147324
[49] L. Zheng, Z. Ma, X. Zhu, B. Tan, S. Li, K. Long, W. Sun, S. Chen, L. Zhang, M. Wan, L. Huang, J. Bai, TJ4DRadSet: a 4D radar dataset for autonomous driving, in: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), IEEE, Macau, China, 2022, pp. 493–498. https://doi.org/10.1109/ITSC55140.2022.9922539
[50] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, (2015). arXiv:1503.02531
[51] J. Zhang, J. Liu, Y. Pei, J. Zhang, X. Zhao, Learn from voxels: knowledge distillation for pillar-based 3D object detection with LiDAR point clouds in autonomous driving, IEEE Trans. Intell. Veh. (2024) 1–11. https://doi.org/10.1109/TIV.2024.3397617
[52] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778. https://doi.org/10.1109/CVPR.2016.90
[53] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117–2125. https://doi.org/10.1109/CVPR.2017.106
[54] M. Hu, W. Yin, C. Zhang, Z. Cai, X. Long, H. Chen, K. Wang, G. Yu, C. Shen, S. Shen, Metric3D v2: a versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation, IEEE Trans. Pattern Anal. Mach. Intell. 46 (12) (2024) 10579–10596. https://doi.org/10.1109/TPAMI.2024.3444912
[55] J. Liu, Q. Zhao, W. Xiong, T. Huang, Q.-L. Han, B. Zhu, SMURF: spatial multi-representation fusion for 3D object detection with 4D imaging radar, IEEE Trans. Intell. Veh. 9 (1) (2024) 799–812. https://doi.org/10.1109/TIV.2023.3322729
[56] H. Zhou, X. Zhu, X. Song, Y. Ma, Z. Wang, H. Li, D. Lin, Cylinder3d: an effective 3D framework for driving-scene LiDAR semantic segmentation, (2020). arXiv:2008.01550
[57] M. Contributors, MMDetection3D: OpenMMLab next-generation platform for general 3D object detection, 2020, (https://github.com/open-mmlab/mmdetection3d).
[58] M. Berman, A. Rannen Triki, M.B. Blaschko, The Lovász-Softmax loss: a tractable surrogate for the optimization of the intersection-over-union measure in neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4413–4421. https://doi.org/10.1109/CVPR.2018.00464
[59] H. Tang, Z. Liu, S. Zhao, Y. Lin, J. Lin, H. Wang, S. Han, Searching efficient 3D architectures with sparse point-voxel convolution, in: European Conference on Computer Vision, Springer, 2020, pp. 685–702. https://doi.org/10.1007/978-3-030-58604-1_41
[60] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, H. Zhao, Depth anything v2, Adv. Neural Inf. Process. Syst. 37 (2024) 21875–21911. https://doi.org/10.5555/3737916.3738604
[61] L. Yang, X. Zhang, J. Li, C. Wang, J. Ma, Z. Song, T. Zhao, Z. Song, L. Wang, M. Zhou, et al., V2X-radar: a multi-modal dataset with 4D radar for cooperative perception, (2024). arXiv:2411.10962