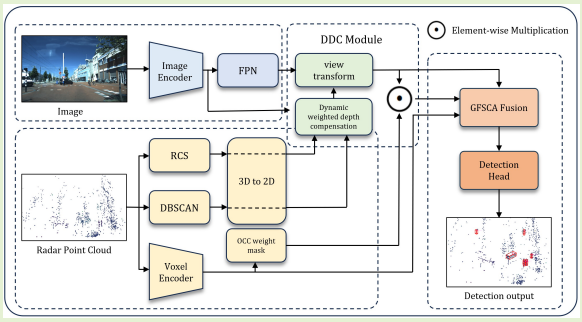
DDCFusion: Dynamic Depth Compensation Fusion for Camera-Radar 3D Object Detection

Jiahao Chen, Huanlei Chen, Ziming Zhu, Zheng Shen, Xiaofeng Ling and Yu Zhu, *Member, IEEE*

***Abstract*— The effective representation and feature extraction from sparse point clouds of 4D millimeter-wave(4D-MMW) radar pose a significant challenge in 3D object detection. This paper proposes DDCFusion, a novel radar-camera fusion network that advances measurement precision by dynamically compensating for depth errors in sparse radar data. DDCFusion achieves this by exploiting the physical properties of 4D-MMW radar to improve measurement reliability and reduce depth uncertainty, which enhances depth measurement confidence in the view transform by integrating RCS-derived reflectivity metrics. The occupancy-weighted radar branch prioritizes image regions with high-confidence radar returns, minimizing measurement noise in view transform operation. Furthermore, DDCFusion optimizes spatial measurement consistency in Bird's-Eye-View (BEV) space by modeling cross-sensor dependencies through the Global Feature Slice Coordinate Attention (GFSCA) fusion module. Experimental validation on the VoD and TJ4DRadSet datasets demonstrates superior measurement accuracy, achieving 51.08% mAP on VoD and 34.61% mAP on TJ4DRadSet—outperforming existing methods in depth error reduction and robustness to sparsity. Ablation studies verify the measurement-centric design: RCS-guided diffusion improves small-object detection (e.g., pedestrians), while DBSCAN-based clustering refines large-object localization (e.g., vehicles). The network demonstrates significant improvements in depth accuracy and robustness to sparse inputs while maintaining competitive inference latency with 138ms.**



***Index Terms*— 4D-MMW radar, camera, multi-modal fusion, Dynamic Depth Compensation, 3D object detection, autonomous driving.**

I. Introduction

WITH the rapid development of the autonomous driving industry [1], research in this field can be broadly categorized into four key areas: perception, prediction, planning, and control. Among these, perception serves as the foundational basis for the subsequent tasks, highlighting its essential role. As a result, significant efforts have been devoted to multi-sensor fusion techniques, along with the ongoing exploration of new sensor technologies. In this context, 4D millimeter-wave radar (4D-MMW radar) has emerged as a promising technology, making 3D object detection using both 4D-MMW radar and camera a highly attractive approach.

Typically, a 4D-MMW radar captures information across four dimensions: range, azimuth, elevation, and Doppler velocity. The raw radar output is often organized as a multi-dimensional data cube, which is subsequently processed through CFAR (Constant False Alarm Rate) detection and other clustering algorithms to generate a 3D point cloud augmented with Doppler velocity as an additional attribute. This post-processing chain transforms the native radar data into a structured point cloud format compatible with common autonomous driving datasets such as: View-of-Delft [19] and TJ4DRadSet [21] . It is worth noting that in widely-used public datasets, this conversion process has already been applied, and the provided data is in the form of processed point clouds(eg., x, y, z, v, RCS) readily usable for detection tasks.

Compared to other onboard sensors, traditional millimeter-wave radar [2] demonstrates superior robustness under adverse weather conditions [3]–[5]. It also provides unique measurement capabilities such as velocity and Radar Cross Section (RCS), which have led to its widespread adoption across various traffic scenarios. However, its utility has been constrained by sparse point clouds and the absence of elevation (z-axis) information, limiting further application. In contrast,

This work was supported in part by National Natural Science Foundation of China under Grant 62476088, in part by the Shanghai Automotive Industry Science and Technology Development Foundation under Grant 2304, and in part by the Science and Technology Commission of Shanghai Municipality under Grant 20DZ2254400. (Corresponding author: Yu Zhu.)

Jiahao Chen, Ziming Zhu, Zheng Shen, Yu Zhu are with the School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China (e-mail: y30230904@mail.ecust.edu.cn, zhuyu@ecust.edu.cn, zimingzhu@mail.ecust.edu.cn, 18506342832@163.com).

Xiaofeng Ling is with the School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China, and also with the Shanghai Key Laboratory of Intelligent Sensing and Detection Technology, East China University of Science and Technology, Shanghai 200237, China (e-mail: xfling@ecust.edu.cn).

Huanlei Chen, Shanghai Motor Vehicle Inspection Certification & Tech Innovation Center Co., Ltd., Shanghai, China (e-mail: huanleic@smvic.com.cn).

4D millimeter-wave radar [6] addresses these shortcomings by incorporating elevation perception and significantly increasing point cloud density, thereby emerging as a valuable component in multi-modal fusion research for 3D object detection.

Currently, the mainstream multi-sensor fusion paradigm primarily revolves around multi-camera systems [7], [8], [9], [10], [11], [12] or camera-LiDAR fusion. While these approaches dominate, they suffer from inherent drawbacks: Vision-only methods, despite their cost efficiency, struggle to achieve high robustness due to the absence of accurate depth information. Camera-LiDAR fusion, despite its strong detection capability, faces challenges in cost scalability [13].

In comparison, 4D-MMW radar offers significantly lower manufacturing costs than LiDAR, positioning it as a viable alternative for camera-LiDAR fusion in detection tasks. Nevertheless, existing 3D object detection methods for 4D-MMW radar remain relatively simplistic. For instance: Some approaches (e.g., SECOND [14], PointPillar [15], CenterPoint [16]) directly utilize LiDAR-based point cloud methods without tailored modifications. Others (e.g., BEVFusion [17], FUTR3D [18]) incorporate multi-sensor fusion but treat 4D-MMW radar features as generic input channels, failing to leverage the unique physical semantics embedded in its measurement dimensions—thereby losing valuable prior knowledge. This gap highlights the substantial untapped potential of camera-4D-MMW radar fusion for 3D object detection.

Moreover, compared to LiDAR, 4D-MMW radar suffers from sparse point clouds. Fig. I visualizes the sparsity issue of 4D-MMW radar point clouds in the View-of-Delft(VoD) dataset [19]. The points shown in Fig. 1a and Fig. 1b represent the projections of point clouds onto the image plane after extrinsic and intrinsic calibration. Only the points falling within the bounding boxes of detected targets are retained. By comparing Fig. 1a and Fig. 1b, it can be observed that the point cloud density of 4D-MMW radar is significantly lower than that of LiDAR, especially for small objects with even fewer points. The limited number of points makes it difficult for 4D-MMW radar-only networks to generate sufficient pillars or voxels and extract adequate features from spatial distributions, which also adversely affects point cloud-radar fusion methods.

Camera-Radar fusion methods can be broadly categorized into sampling-based methods and splatting-based methods. Sampling-based methods leverage the precise depth information of point clouds and are widely adopted in LiDAR-based approaches. However, for 4D-MMW radar with sparse point clouds, the significantly reduced number of samplable points on the image plane leads to performance degradation. In contrast, splatting-based methods can better exploit the dense characteristics of image features but suffer from inaccurate depth estimation, resulting in feature map confusion.

In order to effectively utilize implicit prior knowledge from radar and solve the problem of sparse point cloud for 3d object detection, we propose the DDCFusion. Contributions of this work can be summarized as follows:

- Dynamic Depth Compensation Module, which is composed with RCS-guided depth diffusion and DBSCAN-based clustering diffusion, is introduced to enrich the camera branch with enhanced depth cues. By leveraging the unique information dimensions of 4D-MMW radar, particularly the implicit prior knowledge that higher Radar Cross Section (RCS) values generally correspond to larger reflective surfaces, both depth and semantic features of objects are diffused accordingly. In addition, point features are clustered using DBSCAN [20] and subsequently projected onto the 2D image plane, resulting in increased point cloud projection density. This provides richer depth information to guide the ”Splat & Shoot” operation in view transform, thereby enhancing the expressiveness of the BEV feature representation.
- An occupancy(OCC) weight mask based on point cloud occupancy is introduced in radar branch to enhance image feature. Specifically, image features corresponding to the BEV grid cells occupied by point clouds are separately extracted and selectively enhanced using occupancy-aware weighting, which is derived from the point cloud features. This strategy allows the network to emphasize regions with reliable geometric cues, thereby improving the quality of image features and strengthening the cross-modal fusion.
- Global Feature Slice Coordinate Attention (GFSCA) is

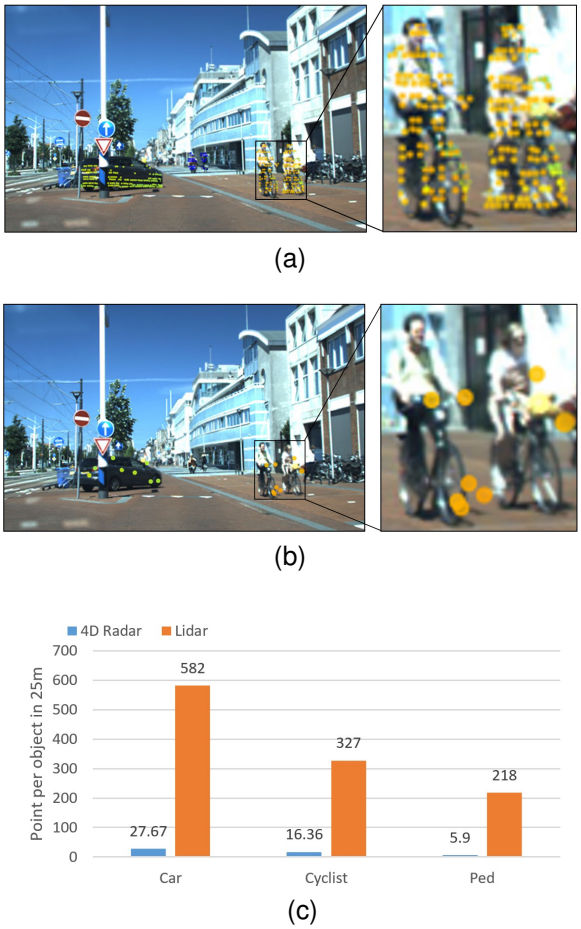


Fig. 1. Visualization of the Radar sparsity in the View-of-Delft(VoD) dataset. (a) illustrates the distribution of LiDAR point clouds within the detection boxes. (b) shows the corresponding distribution for 4D-MMW radar point clouds. (c) presents the average number of points projected onto each category of target across the entire dataset for different sensors.

incorporated into the point cloud fusion process to enhance cross-modal feature interaction. This mechanism is designed to capture global contextual information along the spatial dimensions (i.e., height and width) of the feature maps, generating attention maps that are used to reweight the input BEV spatial features. Through this attention-driven weighting, the complementary characteristics of the camera and 4D-MMW radar modalities are better integrated, resulting in more expressive and discriminative fused features.

- Extensive 3D object detection experiments are conducted on the VoD [19] dataset and TJ4DRadSet [21]. The proposed method outperforms existing radar-camera fusion approaches in terms of key evaluation metrics. Furthermore, ablation studies are performed to validate the effectiveness and contribution of each proposed module.

II. RELATED WORKS

A. Radar-only 3D Object Detection

Current research on utilizing 4D-MMW radar for 3D object detection remains in its early stages. Most existing pure radar-based approaches are adaptations of methods originally designed for LiDAR-based detection, including SECOND [14], CenterPoint [16], PointPillars [15], and RPFA-Net [22]. These methods, while proven effective in the LiDAR domain, face significant limitations when directly transferred to 4D-MMW radar data due to its inherent sparsity and noise characteristics.

Specifically, SECOND adopts a voxelization-based single-stage detection framework. Although efficient, the voxelization process may lose fine-grained details and is less effective for sparse radar point clouds. CenterPoint represents objects as keypoints (object centers) with associated attributes and models object motion directly; however, it exhibits limited robustness when dealing with the extreme sparsity of radar data. PointPillars relies on pillar-based voxelization combined with 2D CNNs, which compress height information and thus struggle to preserve elevation-related cues—particularly problematic for data with pitch angle variations. RPFA-Net focuses on local feature aggregation but heavily depends on point cloud density, making it suboptimal for radar-based inputs.

While these approaches have been partially adapted for the 4D-MMW radar domain by extending the input feature dimensions. For example, incorporating radar-specific attributes such as Doppler velocity and Radar Cross Section (RCS) into the voxel feature encoding (VFE) [23] process. However, there remains a lack of targeted optimization for the unique challenges posed by 4D-MMW radar, such as its extreme sparsity, irregular sampling, and lower spatial resolution. As a result, these adaptations often yield subpar performance compared to their LiDAR counterparts.

B. Camera-Radar Fused 3D Object Detection

In recent years, significant progress has been made in camera−point cloud fusion [24], [25], [26] for 3D object detection, and many of the proposed methodologies are transferable to camera–4D−MMW radar fusion. Multimodal fusion strategies can be broadly categorized into sensor-level fusion and feature-level fusion.

Sensor-level fusion, exemplified by methods such as PointPainting [27], integrates information by projecting radar points or voxels into the camera coordinate frame prior to feature extraction. Similar approaches include MVXNet [28], EPNet [29], and PointAugmenting [30]. However, these methods are highly sensitive to sensor misalignment, where even slight calibration errors can result in significant feature distortion and degrade detection performance. Moreover, they often fail to fully exploit the dense semantic information provided by the camera modality.

In contrast, feature-level fusion has become the mainstream paradigm. BEVFusion [17] enhances the Lift-Splat-Shoot (LSS) [31] pipeline by incorporating point cloud projections into the camera frame to assist depth prediction, thereby improving fusion quality. TransFusion [32] employs a soft association mechanism to adaptively fuse LiDAR BEV features with image features via spatial modulation cross-attention (SMCA) and an image-guided query initialization strategy, improving robustness under poor imaging conditions and sensor misalignment. FUTR3D [18] achieves end-to-end multimodal fusion by leveraging a modality-agnostic feature sampler (MAFS) and a Transformer decoder, resulting in improved detection accuracy and reduced computational cost.

However, when these frameworks are directly transferred to 4D-MMW radar-based detection tasks, they typically adopt a simplistic strategy of increasing the input feature dimensions to accommodate radar-specific attributes (e.g., velocity, RCS), without explicitly leveraging the implicit priors inherent in radar data. In this context, RCFusion [33] was among the first to consciously exploit additional radar-specific information by designing a specialized PFN layer for targeted feature extraction. Subsequently, RCBEVDet [34] introduced the idea of RCS-driven BEV feature diffusion. The core idea lies in the in-depth exploration and utilization of the inherent physical properties of radar point clouds (such as radial distance and Doppler velocity) to construct and enhance BEV features. This approach goes beyond treating radar point clouds merely as sets of 3D spatial points; instead, it transforms them into radar feature maps rich in semantic information.

III. PROPOSED METHOD

A. Overall Architecture

The overall architecture of the proposed model is illustrated in Fig. 2. The framework is composed of four primary components: The Image Branch based on Lift-Splat-Shoot (LSS) [31], the Radar Branch with occupancy(OCC) weight mask, Dynamic Depth Compensation Module, Fusion and Detection Head. A detailed explanation of each module is provided in the following subsections:

1) The image branch is responsible for visual feature extraction, where a pre-trained Swin-Transformer [35] backbone is employed to generate hierarchical features. These extracted features are subsequently projected into Bird's Eye View (BEV) space through the modified

This article has been accepted for publication in IEEE Sensors Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2025.3647645

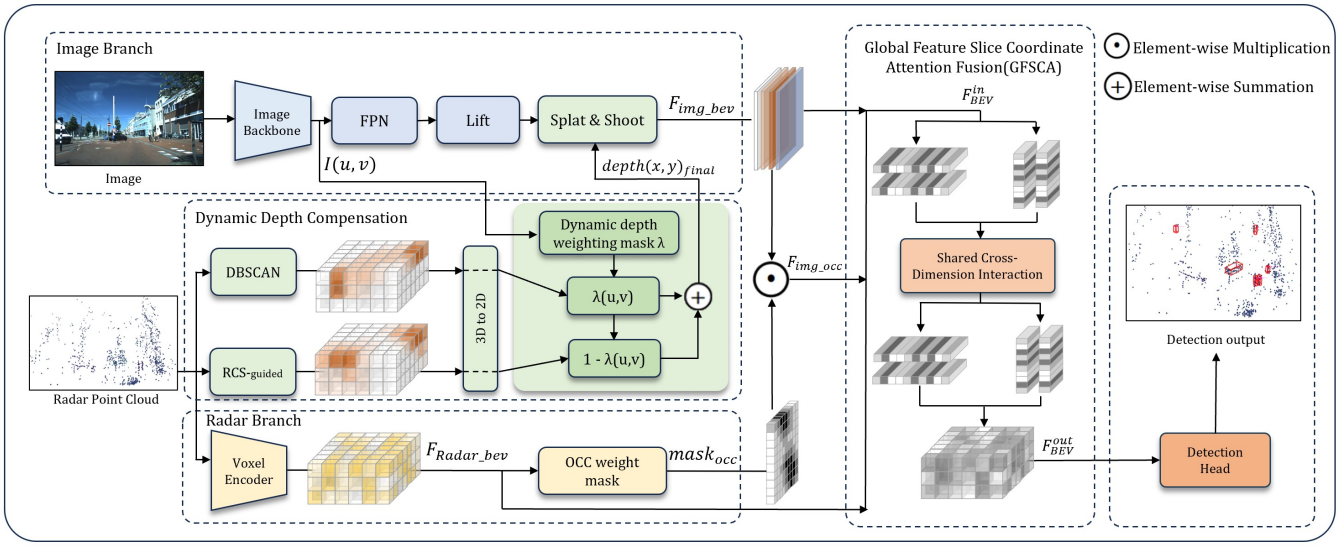


Fig. 2. The overall architecture of the proposed DDCFusion.It takes 4D-MMW radar point clouds and camera as dual-input to the network. The Dynamic Depth Compensation module enhances dual-branch feature representation through dynamically weighted depth information from DBSCAN-based clustering diffusion and RCS-guided diffusion. The Radar branch is responsible for extracting voxel feature and generate OCC mask. The Global Feature Slice Coordinate Attention (GFSCA) Fusion module processes three inputs: voxel-wise features from the radar branch, image BEV features weighted by OCC mask and original image BEV features.

LSS operation to facilitate cross-modal fusion with radar features.

2) The radar branch is designed to extract point cloud features while simultaneously generating BEV-space occupancy(OCC) weight mask based on these features to enhance the network's spatial awareness of the scene. Additionally, the module utilizes occupancy information extracted from the radar point cloud to generate weight mask, which are applied to selectively enhance the image features at locations corresponding to occupied regions.
3) The Dynamic Depth Compensation module is proposed to guide the depth reasoning of image-projected features using radar-derived cues. Specifically, the module incorporates clustering results from the radar branch based on point cloud DBSCAN-based clustering and Radar Cross Section (RCS) information to provide auxiliary supervision for pixel-level depth distribution in the image branch. Since these two mechanisms have different performance on different size of target, a depth weighting mask λ is designed to dynamically decide the weight of two augment method.
4) The GFSCA Fusion module integrates the image features and radar features extracted from their respective branches. By combining these heterogeneous modality-specific representations, the fusion module plays a critical role in enhancing the feature extraction capability within the BEV space. Effective fusion ensures that the subsequent detection head can fully exploit the complementary characteristics information provided by both modalities, thereby improving the accuracy and robustness of 3D object detection.

B. Image Branch

The image branch primarily consists of two components: an image backbone and a depth estimation module.

The image backbone is responsible for extracting multi-scale semantic features from the input RGB images. In this work, we adopt a pretrained Swin-T as the backbone due to its superior performance in both multi-scale representation learning and computational efficiency compared to traditional convolutional networks such as ResNet [36]. The hierarchical architecture and window-based self-attention mechanism of Swin-Transformer enable effective global context modeling while maintaining high resolution, which is crucial for precise spatial reasoning in the BEV space.

The depth estimation module first enhances image features using the FPN [37], then feeds the refined features into the modified LSS framework to transform the 2D image features into a structured 3D representation. The FPN employed here adopts the GeneralizedLSSFPN [17], which utilizes concatenation to preserve more original feature information compared to the addition operation in standard FPN. This mitigates the issue of semantic information in deeper features overshadowing fine-grained details.

For each pixel location (u, v), the module estimates a probabilistic depth distribution $D(u, v) \in R^Z$, where Z denotes the number of discretized depth bins. This distribution reflects the likelihood of the pixel belonging to different depth levels along the camera ray. The detailed equation is presented in Appendix A.

The lifted 3D features are projected onto the Bird's-Eye View (BEV) plane using the splatting operation, guided by the camera's intrinsic and extrinsic parameters. This operation constructs a dense BEV feature map by accumulating the depth-aware 3D features along the vertical axis. Specifically, each 3D point (x, y, z) in the camera coordinate system

is projected onto a 2D BEV coordinate (x_{bev}, y_{bev})on the horizontal plane (typically the X-Y plane), according to:

$$\begin{pmatrix} x_{\text{bev}} \\ y_{\text{bev}} \\ 1 \end{pmatrix} = K \cdot T \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{1}$$

K denote the camera intrinsic matrix, and T represent the extrinsic transformation matrix from the camera coordinate system to the BEV coordinate system.

Finally, FPN is applied to the BEV feature map to perform multi-scale feature extraction and fusion, which enhances the representational capacity of the BEV features and benefits downstream 3D object detection tasks.

C. Radar Branch

For the radar branch, the input point cloud is first voxelized using a 7-channel mean Voxel Feature Encoder (VFE) [15]. The resulting voxels are then passed into the radar backbone, specifically the VoxelResBackbone8x [39]. This backbone integrates voxelization, 3D convolutions, residual connections, and sparse convolutions to extract hierarchical features from the point cloud data, offering strong representational capacity for downstream tasks such as 3D object detection and semantic segmentation. Its $8\times$ downsampling architecture makes it particularly well-suited for processing large-scale point clouds.

Subsequently, the point cloud features are further compressed to produce the final radar feature map in the bird's eye view (BEV) space, denoted as F_{Radar_bev}, with a shape of (x, y, c), where c represents the number of feature channels. This feature map is then utilized to infer occupancy in the BEV coordinate system.

In detection tasks, the depth information provided by 4D-MMW radar is generally considered more reliable than the depth predicted by methods such as Lift-Splat-Shoot (LSS). Therefore, the spatial distribution of radar point clouds can be used to assess the reliability of the corresponding image-based depth predictions.

To this end, the OCC weight mask $mask_{occ}$ obtained from the radar branch serves as a confidence mask (see Appendix B for a more detailed equation). It assigns higher confidence to regions where radar-to-image projections are accurate, while suppressing regions exhibiting abnormal or unreliable depth projections.

D. Dynamic Depth Compensation Module

As one of the most distinctive characteristics of 4D-MMW radar compared to LiDAR, its point clouds inherently contain unique velocity and Radar Cross Section (RCS) dimensions. As Fig. 3 shown, the Dynamic Depth Compensation(DDC) Module is designed with three key components: a depth diffusion module based on DBSCAN clustering, an RCS feature diffusion module, and the image-deciding dynamic weight λ.

During the training of the depth estimation module, the performance of image-based depth prediction suffers due to the inherent lack of depth cues in RGB images. To address this, ground truth depth values from the point cloud are projected onto the image to serve as supervision signals for depth prediction. While this approach has shown promising results with LiDAR due to its high point density, it proves less effective for radar because the sparse radar points can only supervise a limited number of image pixels, significantly weakening the depth supervision effect. DDC Module can resolve this issue.

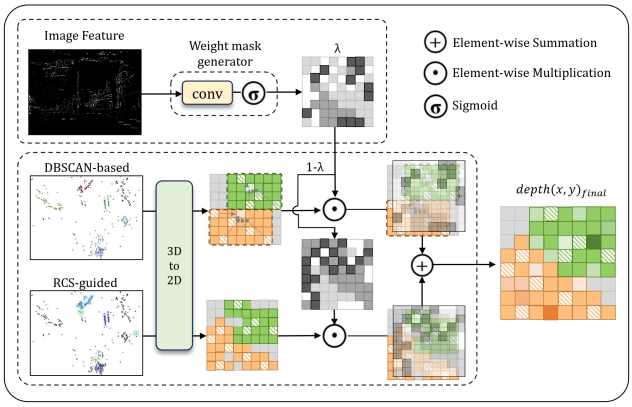


Fig. 3. The detailed architecture of Dynamic Depth Compensation. The inputs are composed of image feature, DBSCAN-based clustering diffusion and RCS-guided diffusion. The output is the final compensation depth $depth(x, y)_{final}$. The color-coded grid cells represent distinct clusters, while patterned cells denote point attributes. Specifically, textured grid cells correspond to authentic points projected onto the 2D image plane, whereas non-textured cells indicate supplemental depth values generated by the front-end branch network for density completion.

1) DBSCAN-based Clustering Diffusion: This strategy propagates sparse radar depth information to a broader set of pixels, thereby enhancing the performance of depth prediction. Benefiting from the relatively low number of radar points per frame, we apply DBSCAN [20] clustering directly to the entire point cloud of the scene, as illustrated in Fig. 4a. In the DBSCAN, Eps defines the neighborhood radius for searching adjacent points, while MinPts specifies the minimum number of points required to form a dense region. In our implementation, we set Eps = 1.0 m and MinPts = 3. The feature dimensions used for clustering include not only spatial coordinates (x, y, z) but also radar-specific attributes such as radial velocity v_r, relative radial velocity v_{cr}, and RCS. Subsequently, all non-noise points are projected onto the image plane using intrinsic and extrinsic calibration matrices. The minimum bounding rectangle enclosing each cluster on the image plane is then determined, and the pixels within these rectangles are treated as the target regions for depth diffusion.

For pixels within the target region that do not have ground-truth depth values, we perform a per-pixel traversal and fill in the missing depth values. The depth weight $weight(x, y, i)$ is computed as the reciprocal of the Euclidean distance, meaning that real depth values closer to the current pixel location are assigned higher credibility, as Eq. 2 shown. The depth information is then weighted and summed, and assigned to the corresponding pixel location, achieving the effect of depth diffusion, using the Eq. 3:

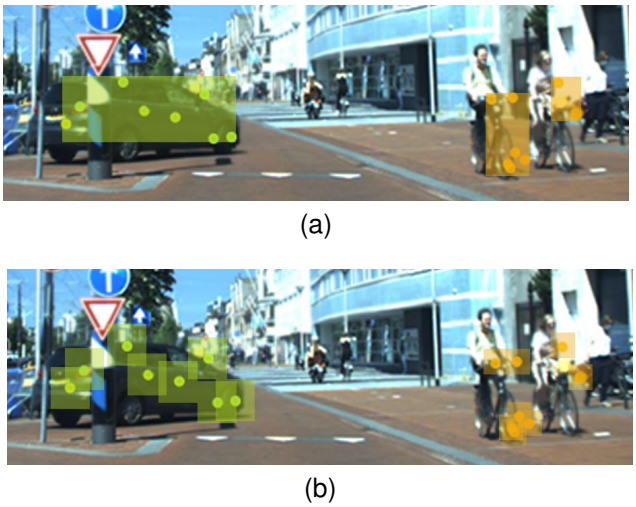


Fig. 4. Illustration of two depth enhancement strategies. (a) DBSCAN-based clustering diffusion: depth values from radar points are propagated to nearby pixels within the bounding rectangle of each point cluster. (b) RCS-guided depth diffusion: radar RCS information is spatially diffused according to the variable radius R.

$$weight(x,y,i) = \frac{1}{\sqrt{(x-x_i)^2 + (y-y_i)^2}} \tag{2}$$

$$depth(x,y)_{cluster} = \frac{\sum_1^n weight(x,y,i) \times depth_i}{\sum_1^n weight(x,y,i)} \quad i \in 1,2,\ldots,n \tag{3}$$

Here, n denotes the total number of known depth points within a single point cluster. Each (x_i, y_i)represents a pixel with an associated radar point, and $depth_i$ denotes its corresponding ground-truth depth value. The contribution of each known depth point to an unknown pixel is weighted by $weight(x,y,i)$. The final estimated depth at a given pixel, denoted as $depth(x,y)_{cluster}$, is computed using Eq. 3.

2) RCS-guided Depth Diffusion: The RCS-guided depth diffusion leverages a prior assumption: objects with larger RCS values tend to occupy more area in the BEV space and affect a greater number of grid cells. This prior is difficult to learn solely through supervised training. To integrate this prior knowledge, we perform feature diffusion from pixels with known depth values, as illustrated in Fig. 4b. We convert the RCS value in dBsm back to its linear value (i.e., in m^2), and to ensure that a larger RCS influences a broader diffusion region, the result is made inversely proportional to the depth. Finally, a hyperparameter k is multiplied to control the magnitude of the RCS diffusion. The computation is defined as Eq. 4:

$$R = k \times \frac{10^{\frac{RCS}{10}}}{depth_i} \quad i \in 1,2,\ldots,n \tag{4}$$

Surrounding pixels within the square $R \times R$ are influenced by the depth values of nearby known points through a diffusion process. The propagated depth value from the i th known point to a target pixel (x,y) is denoted as $depth(x,y)^i_{RCS}$. To prevent excessively large influence regions, an upper threshold is imposed on max radius.

3) Dynamic Weighted Depth Compensation: Naturally, a single pixel may fall within the influence zones of multiple known depth points, resulting in multiple propagated depth values. In such cases, we compute the final depth at that pixel by averaging the contributions, as formulated below:

$$\begin{aligned} depth(x,y)_{final} &= \frac{1}{1+n} \times (\alpha + \beta) \quad i \in 1,2,\ldots,\mathrm{n} \\ \alpha = depth(x,y)_{cluster}, &\quad \beta = \sum_1^n depth(x,y)^i_{RCS} \end{aligned} \tag{5}$$

Here, $depth(x,y)_{final}$denotes the final estimated depth at pixel (x,y), and n still represents the number of propagated depth values affecting that pixel.

Experimental results indicate that the DBSCAN-based clustering method is more effective for large object detection, whereas the RCS-guided depth diffusion approach demonstrates better performance for small object detection. Motivated by this observation, we adopt a more targeted strategy for depth fusion by introducing adaptive weighting. Specifically, we convolve the image-branch feature map to generate dynamic depth weighting coefficients $\lambda(x,y)$ for each pixel location.

The final depth estimation is then computed using the following formulation:

$$depth(x,y)_{final} = \frac{1-\lambda}{1+\mathrm{n}} \times \alpha + \frac{\lambda}{1+n} \times \beta \tag{6}$$

$$\lambda(u,v) = Sigmoid(conv(I(u,v))) \tag{7}$$

And $depth(x,y)_{final}$ is compensated into the "Splat & Shoot" process to achieve more accurate 2D-to-3D projection. The method employed to compensate leverages a depth prediction tensor generated by the "Lift" module. This tensor shares the channel dimensionality with the BEV grid but is flattened into a 1D vector. The radar branch's depth estimate is accordingly reshaped into an identically structured 1D tensor and concatenated with the image-based depth vector. A convolutional layer subsequently processes this concatenated tensor, reshaping it back to the original tensor dimensions to produce the final, refined depth prediction.

E. GFSCA Fusion Module

Conventional multimodal fusion modules often rely heavily on the global attention mechanism inherent in transformers [40], without explicitly modeling global information along the height and width dimensions. This lack of directional focus limits the feature representation capability under the same number of training epochs.

To address this limitation, we propose the Global Feature Slice Coordinate Attention(GFSCA) Fusion Module, which enhances feature representations in BEV space by generating attention maps that capture global spatial context and applying them to reweight the input feature maps accordingly.

Specifically, GFSCA captures global contextual information along both the height and width axes by independently applying global average pooling and max pooling across each spatial

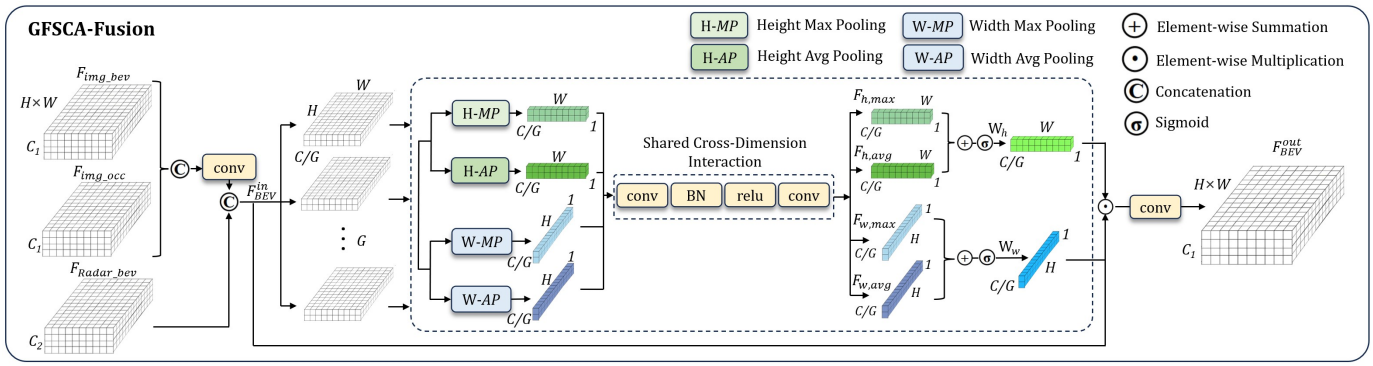


Fig. 5. The architecture of GFSCA Fusion. F_{img_bev} is original image BEV features, F_{img_occ} is image BEV features weighted by $mask_{occ}$, F_{Radar_bev} is voxel-wise features from the radar branch.

direction. This directional pooling strategy enriches the fusion module's ability to extract comprehensive global features.

Furthermore, processing full-resolution global BEV feature maps can be computationally intensive. To mitigate this, GFSCA adopts a channel-wise grouping strategy: the input feature map is divided into multiple groups along the channel dimension, reducing the computational overhead per group while preserving the diversity and richness of feature representations.

As illustrated in Fig. 5, the feature maps from the Image and Image-Occ branches are first fused and passed through a convolutional layer. The resulting feature map is then concatenated with the Radar branch features to form the unified BEV feature representation F_{BEV}^{in} .

Along the channel dimension, the unified feature map is divided into G groups, with each group containing C/Gchannels. Here, B denotes the batch size, C is the total number of channels, and $H$and W represent the height and width of the feature map respectively.

Subsequently, global average pooling and global max pooling are independently applied along the height and width dimensions of each grouped feature map. For each group, a shared convolutional module is used to process the pooled features. This shared convolutional block consists of two 1×1 convolutional layers, a batch normalization layer, and a ReLU activation function. The first 1×1 convolution reduces the channel dimensionality, while the second restores it, enabling efficient yet expressive feature transformation across groups. Full equations are provided in Appendix C, as Eq. 13 shown.

By summing the outputs of the convolutional layers and applying a Sigmoid activation function, attention weights are generated for both the height and width directions, as Eq. 14 shown in Appendix C.

Here, σ denotes the Sigmoid activation function.

Finally, the input feature map is reweighted using the attention weights. The attention maps W_h and $W_w$are broadcasted along the height and width dimensions, respectively, to match the spatial size of the input feature map. The final output feature map is then computed as Eq. 15 in Appendix C.

IV. Experiments

A. Dataset and Evaluation Metrics

In this study, we evaluate the proposed method on two publicly available datasets: View-of-Delft (VoD) [19] and TJ4DRadSet [21]. Both datasets are designed for 3D object detection and tracking tasks, and provide synchronized multi-modal sensor data, including point clouds from 4D-MMW radar and LiDAR, consecutive video frames from RGB cameras, and annotated 3D bounding boxes.

For the VoD dataset, we follow the common practice of evaluating on three object categories: Car, Pedestrian, and Cyclist. Most existing works also report results on these classes, which facilitates fair and direct comparisons. In our experiments, we use temporally accumulated radar point clouds over 3 consecutive frames. A total of 5,084 frames are used for training, and 1,351 frames are used for validation. To ensure comparability with prior work, all evaluation metrics are reported on the validation set, without distinguishing a separate test set.

To assess the model's performance, we adopt the official evaluation metric AP3D for per-class comparison, and mAP3D as the overall performance indicator. For both datasets, we use the following Intersection-over-Union (IoU) thresholds to determine true positives (TP): 0.5 for Car, 0.25 for Pedestrian, and 0.25 for Cyclist.

For the TJ4DRadSet, in addition to the three common categories, we also evaluate on the Truck class, using an IoU threshold of 0.5. The dataset is split into 5,717 frames for training and 2,040 frames for testing.

B. Implementation Details

Our model is implemented and deployed based on the open-source OpenPCDet framework [41], a widely adopted platform for 3D object detection. Our proposed model and all compared models were trained and evaluated on a computing platform equipped with four NVIDIA A40 GPUs.

Hyperparameter Settings: For the VoD dataset, the range of radar point clouds used during inference is restricted to:$0 < x < 51.2m$, -$25.6m < y < 25.6m$, -$3m < z < 2m$, and the point cloud is composed of 3 frames scan. And we also inference our method on 5-frames point cloud. For the TJ4DRadSet, the point cloud range is defined as: $0 < x < 69.12m$, -$39.68m < y < 39.68m$, -$4m < z < 2m$. The radar

TABLE I

COMPARSION WITH OTHER METHODS ON THE VALIDATION SET OF VOD DATASET, R IS RADAR AND C IS CAMERA. THE RESULTS MARKED WITH # ARE INHERITED FROM [42], [33] AND [34].

	sensor	Entire Annotation Area				Driving Corridor Area				Latency(ms)
		Car	Ped	Cyc	mAP_{3D}	Car	Ped	Cyc	mAP_{3D}	
PointPillars [15]	R	37.24	32.19	66.8	45.41	70.55	43.28	88.13	67.32	60
CenterPoint [16]	R	32.74	38.1	65.51	45.42	62.01	48.18	84.98	65.06	23
Second [14]	R	40.4	30.64	62.51	44.52	72.25	41.19	83.39	65.61	67
#SMURF(5-frame) [42]	R	42.31	39.03	71.5	50.97	71.74	50.54	86.87	69.72	44
FUTR3D [18]	R+C	**46.01**	35.11	65.98	49.03	**78.66**	43.1	86.19	69.32	137
#RCFusion [33]	R+C	41.7	38.95	68.31	49.65	71.87	47.5	88.33	69.23	/
#RCBEVDet [34]	R+C	40.63	38.86	**70.48**	49.99	72.48	49.89	87.01	69.8	/
BEVFusion [17]	R+C	37.85	40.96	68.95	49.25	70.22	45.87	89.45	68.52	140
DDCFusion(ours)	R+C	41.11	**42.29**	69.84	**51.08**	75.37	**50.17**	**90.55**	**72.03**	138
DDCFusion(5-frames)	R+C	42.24	44.51	71.93	52.89	78.65	52.84	91.73	74.4	164

Note: The entries labeled "5-frames" used 5-frames data during inference, while the unlabeled entries used 3-frame data for inference.

voxel size is set to $0.1\times0.1\times0.125$, and each voxel can contain up to 5 radar points. The stride of the radar feature extractor (including the backbone and neck) is configured to 2, resulting in a final BEV resolution of 256×256. Detection Head. We adopt the TransFusionHead [32] as the detection head. For each frame, 200 initial object proposals are generated. The Transformer uses: 128 hidden channels, 8 attention heads, a feed-forward network with 256 channels, a downsampling stride of 8, a Gaussian overlap threshold of 0.1, and a minimum Gaussian radius of 2. Training Strategy. During training, random horizontal flipping and scale augmentation (within the range of [0.95, 1.05]) are applied. Input images are normalized before being fed into the network.

We use the Adam optimizer with an initial learning rate of 0.0001 and a weight decay of 0.01. The model is trained for 80 epochs, with a batch size of 4 per GPU. A cosine annealing learning rate schedule is employed, with a warm-up phase comprising 40% of the total epochs (PCT_START).

To ensure training stability, gradient clipping is applied with a maximum norm of 35 (GRAD_NORM_CLIP). For mixed-precision training (FP16), the loss scaling factor is set to 32 (LOSS_SCALE_FP16). Additionally, the learning rate is decayed at epoch 35 and 45 following a predefined DECAY_STEP_LIST to further enhance convergence.

The overall configuration is designed to balance training stability and convergence efficiency.

C. Results and Analysis

Table I presents the quantitative results on the validation set. Our proposed method ranks first among all point cloud–only and point cloud–camera fusion approaches, achieving a 1.09% higher mAP compared to the second-best method, and outperforming the baseline BEVFusion by 1.83%. Notably, our approach achieves the highest AP on the Pedestrian class, reaching 42.29%. This improvement highlights a key limitation of previous transformer-based fusion methods: the sparse nature of 4D-MMW radar point clouds offers insufficient depth supervision, particularly for objects with weak reflectivity and small reflective surfaces, such as pedestrians. The RCS-guided depth diffusion module effectively addresses this by increasing the density of the BEV feature map. Leveraging the RCS prior, it diffuses feature information to radar-sparse pedestrian regions, leading to performance gains.

TABLE II

RUNTIME BREAKDOWN OF PROPOSED MODEL ON VOD DATASET.

Module	Latency(ms)	Percentage(%)
Image branch	65	47.1%
Radar branch	45	32.61%
DDC Module	3	2.17%
OCC Mask	2	1.44%
GFSCA Module	16	11.59%
Detection Head	7	5.07%
Sum	138	

Compared to BEVFusion, our method achieves a 3.26% improvement in the Car category, demonstrating the effectiveness of the DBSCAN-based clustering diffusion module. This module enables the network to reconstruct dense depth representations from sparse radar point clusters, enhancing the visual branch's inference capability. Since large objects typically exhibit dense radar clusters, the module is particularly advantageous for detecting such targets, thus contributing to greater improvements on larger objects.

Inference latency comparisons are also shown in Table I. The latency data for some methods is missing because it was not provided in the original papers. Our model achieves a per-frame inference time of only 138 ms, owing to the GFSCA fusion module, which performs group-wise operations on BEV feature maps to reduce computation. Getting a 1 ms decrease compared to BEVFusion, with corresponding 1.83% performance gain justifies the trade-off in real-time applications.

To provide a more detailed analysis, Table II presents the time consumption and corresponding descriptions of each major module during inference under the condition of inputting 3-frames dataset. It can be observed that the primary computational overhead originates from the image branch and the radar branch themselves, which collectively account for 79.7% of the total inference time. And the majority of computations in the DDC Module are concentrated on CPU-based processing of the raw point cloud. This computational process can be executed in parallel with the preceding image and radar branches, thereby limiting its dedicated inference time for learnable parameters to only 3 ms. Then, the GFSCA

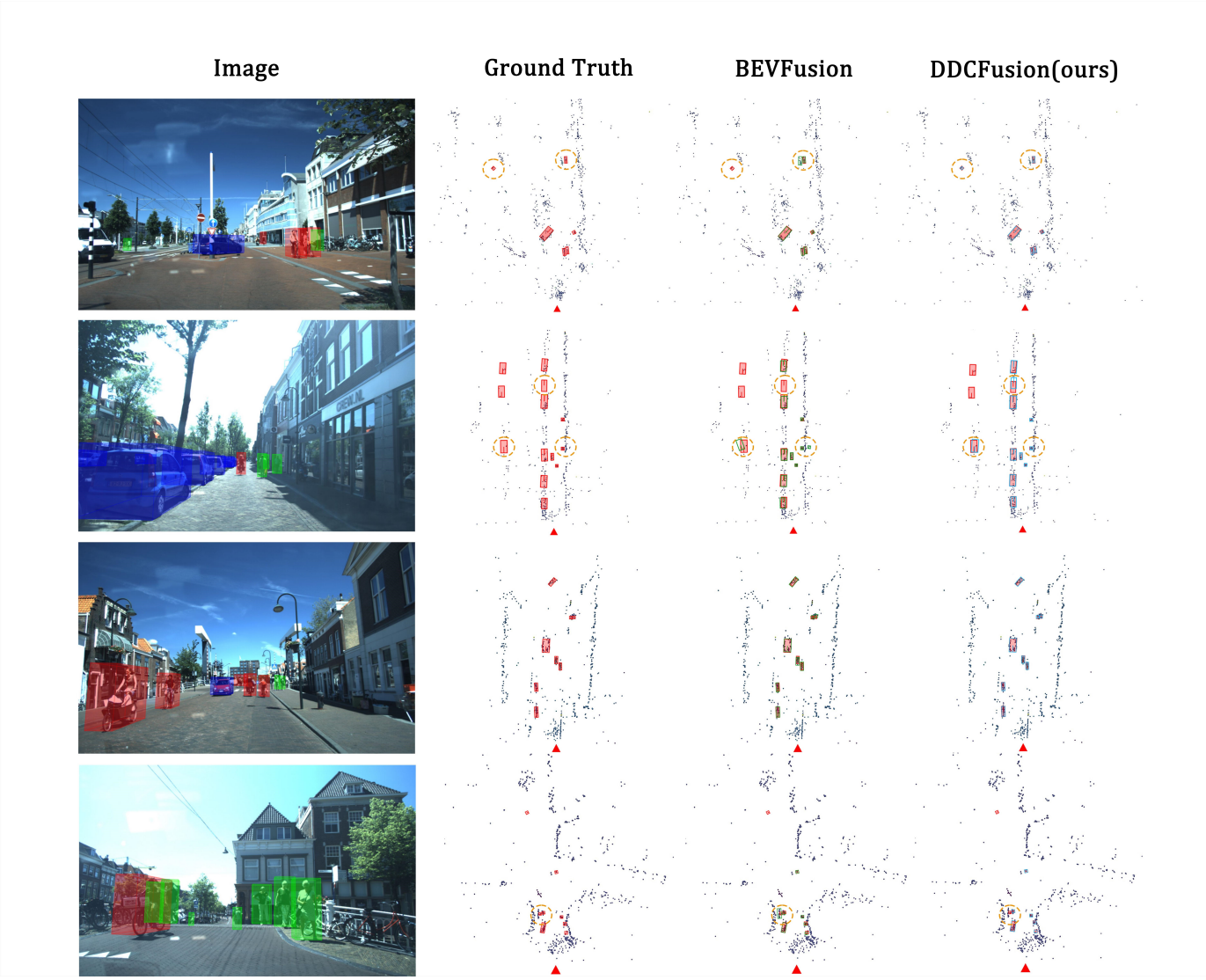


Fig. 6. Visualization results on the VoD dataset. Each column represents the detection result for a single frame. Black dots indicate radar point clouds, red boxes denote ground truth bounding boxes, and the red triangle marks the ego vehicle's position.The first row shows the 2D scene visualization overlaid with ground truth annotations.The second row displays detection results from BEVFusion, highlighted in green boxes.The third row presents detection results from our proposed method, shown in blue boxes.

module employs a grouped processing strategy, the input feature maps are divided along the channel dimension, effectively reducing the computational load per group while preserving the diversity and richness of feature representation.

Fig. 6 presents a qualitative comparison between our method and BEVFusion on the VoD dataset. Our approach consistently achieves more accurate detections across all categories, and is particularly effective at detecting small objects even without associated radar points—demonstrating the advantages of the GFSCA fusion module in leveraging dense image features. Furthermore, for both large and small objects, our method produces accurate distance-aware bounding boxes even in areas with sparse point clouds, indicating that the dynamic depth compensation mechanism significantly improves depth estimation in the image branch.

Our method demonstrates more accurate orientation estimation when detecting occluded and distant vehicles, and significantly reduces false positives on pedestrians caused by radar noise. This improvement is attributed to the proposed GFSCA fusion module, which generates attention maps through shared convolutions along both the horizontal and vertical axes of the BEV feature map. This enhances the network's ability to interpret surrounding scene features and suppresses disturbances introduced by sparse or noisy point clouds.

Moreover, our approach can still distinguish object categories and generate accurate bounding boxes for small and distant targets with sparse radar points. This capability stems from the camera-radar feature enhancement module, which dynamically expands the limited but reliable depth cues from radar across a broader region on the image plane. In doing so, the system leverages dense and accurate visual depth information to compensate for the lack of reliable point cloud data, enabling more precise detection for targets that are

TABLE III

PERFORMANCE COMPARISON WITH OTHER METHODS ON THE TJ4DRADSET TEST SET. THE RESULTS MARKED WITH # ARE INHERITED FROM [33], [22], [28] AND [42].

TJ4D	sensor	Car	Ped	Cyc	Tru	mAP_{3D}	Car	Ped	Cyc	Tru	mAP_{BEV}
#RPFA-Net [22]	R	26.89	27.36	50.95	14.46	29.91	42.89	29.81	57.09	25.98	38.94
#MVX-Net [28]	R	22.28	19.5	50.7	11.21	25.94	37.46	22.7	54.69	18.07	33.23
#SMURF [42]	R	28.47	26.22	54.61	22.64	32.99	43.13	29.19	58.81	**32.8**	40.98
PointPillar [15]	R	27.65	27.33	50.02	16.1	30.28	43.53	28.77	57.31	24.11	38.43
CenterPoint [16]	R	20.68	13.51	40.56	17.05	22.95	36.57	21.86	53.44	26.1	34.49
Second [14]	R	21.85	16.22	38.08	15.78	23.23	39.62	24.58	46.75	21.43	33.09
#RCFusion [33]	R+C	29.72	27.17	**54.93**	23.56	33.85	40.89	30.95	58.3	28.92	39.76
BEVFusion [17]	R+C	34.94	27.08	47.3	21.92	32.81	43.19	33.51	**59.7**	28.94	41.32
DDCFusion(ours)	R+C	**36.89**	**28.29**	49.64	**23.62**	**34.61**	**43.73**	**34.08**	59.49	29.7	**41.75**

otherwise challenging to localize using radar data alone.

However, our approach still exhibits certain limitations. As indicated in the last row of the Fig. 6, when a large number of small objects are heavily occluded by foreground elements, the camera branch fails to extract sufficient features for effective fusion, leading to missed detections, such as the leftmost pedestrian in the example. Furthermore, extremely distant small objects occupy too few pixels to provide meaningful texture information, which also results in false negatives for far-range pedestrians.

To further evaluate the effectiveness of our proposed method, we conducted additional experiments on the TJ4DRadSet. The performance of various methods on the test set of TJ4DRadSet is presented in Table III, where our approach achieves the highest overall mAP_{3D}, outperforming RCFusion by a margin of 0.76%. It is noteworthy that our method exhibits a noticeably lower mAP_{3D} for the cyclist category compared with the best-performing approach. This discrepancy can be attributed to two primary factors: firstly, the baseline model itself demonstrates relatively weak performance on the cyclist category; secondly, while radar point clouds for cyclists are generally sparse, the metal components of bicycles often yield high RCS values. This combination leads the RCS diffusion module to erroneously predict an abnormally large depth distribution area, thereby adversely affecting the localization accuracy.

The superiority of our method is particularly evident in large object detection, especially for the Car and Truck categories, where it surpasses all other competing methods. This further validates the effectiveness of the DBSCAN-based depth enhancement strategy.

Additionally, for Pedestrian and Cyclist categories, our method also achieves notable improvements of 1.11% and 2.3% over the baseline, respectively. These results highlight the effectiveness of both the Dynamic Depth Compensation(DDC) module and the GFSCA fusion module in enhancing the detection of smaller or less reflective targets.

Fig. 7 presents a comprehensive comparison of different methods on both 3D occupancy maps and BEV (Bird's Eye View) maps. Our approach demonstrates the best performance across all distance ranges, particularly achieving a 1.71% improvement over existing methods in the critical 0-25m range. This performance advantage stems from the denser annotation distribution in the 0-25m zone, where our method's global optimization yields the most pronounced benefits.

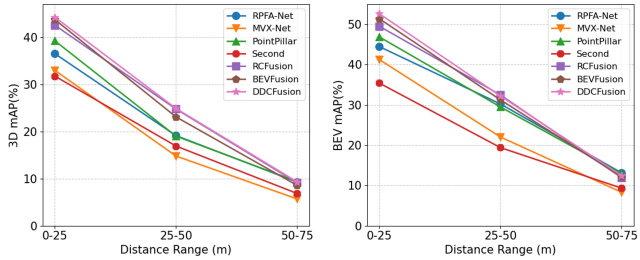


Fig. 7. 3D mAP and BEV mAP on TJ4DRadSet with different methods.

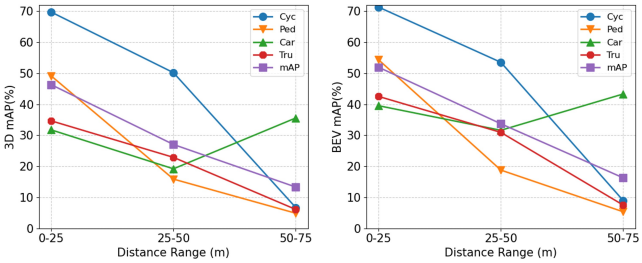


Fig. 8. 3D mAP and BEV mAP of DDCFusion on TJ4DRadSet with different classes.

Fig. 8 demonstrates the performance metrics of different classes at varying distances using DDCFusion. The detection accuracy for pedestrian (Ped), cyclist (Cyc), and truck (Tru) categories decreases with increasing distance. However, an anomalous performance improvement is observed for the Car class within the 50-75m range, which may be attributed to the relatively higher annotation density for distant vehicles in the dataset. Notably, Cyc exhibits significantly superior performance compared to other classes. This can be explained that cyclists present a larger effective radar reflection than pedestrians, resulting in more detectable points, and the assessment uses the same strict AP@0.25 IoU threshold (originally designed for pedestrian detection) for cyclist evaluation, which favors objects with more stable detection signals.

TABLE IV

PERFORMANCE ON DIFFERENT ILLUMINATION, OVER DENOTES OVER-EXPOSURE.

Models	mAP_{3D}			mAP_{BEV}		
	Dark	Standard	Over	Dark	Standard	Over
BEVFusion	20.2	41.56	18.37	25.22	48.58	27.27
DDCFusion	22.49	42.85	21.79	27.13	49.07	29.15

TABLE V

ABLATION STUDIES ON DIFFERENT MODULES. EXPERIMANTS ARE CONDUCTED ON VOD DATASET.

DDC Module		OCC mask	Fusion	Car	Ped	Cyc	mAP_{3D}
DBSCAN	RCS						
-	-	-	SENet [43]	37.85	40.96	68.95	49.25
✓	-	-	SENet	39.03	40.79	68.82	49.54
-	✓	-	SENet	38.25	40.92	69.64	49.93
Fixed λ =0.5	Fixed λ =0.5	-	SENet	39.97	41.13	69.35	50.15
Learnable λ	Learnable λ	-	SENet	40.23	41.48	69.09	50.26
Learnable λ	Learnable λ	✓	SENet	40.46	41.52	68.93	50.3
Learnable λ	Learnable λ	✓	GFSCA	41.11	42.29	69.84	51.08

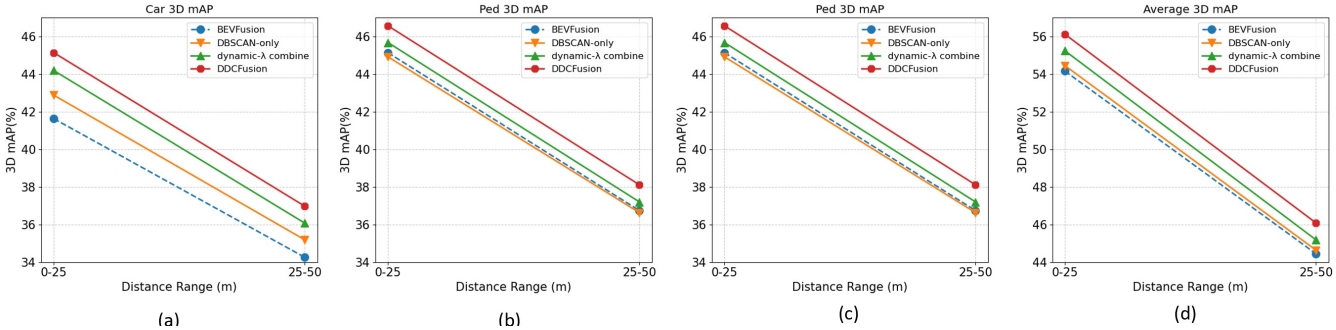


Fig. 9. Per-class distance breakdowns for the modules on the VoD dataset. The difference between the upper and lower limits of the y-axis is kept constant across all tables for comparison purposes.

To further investigate the impact of lighting conditions on the DDCFusion, we analyzed the detection performance on the TJ4DRadSet in Table IV. Specifically, the test set was partitioned into three subsets based on scene brightness: Dark, Standard and Overexposure. These subsets account for 10.9%, 74.7%, and 14.4% of the entire test set, respectively. To validate the effectiveness of the improvements in DDCFusion, we conducted a comparative analysis with the baseline.

By comparing the results of BEVFusion and DDCFusion on the same subsets, our method demonstrates improvements across all lighting conditions in Table IV. However, the performance gains are significantly more substantial under challenging illumination: the mAP_{3d} metric increases by 2.29% in Dark conditions and by 3.32% in Overexposure conditions, far exceeding the 1.29% improvement observed under Standard lighting. This can be attributed to the significant degradation of image quality in extreme lighting conditions, which renders the view transform quality relying solely on the image output less reliable. The incorporation of depth information from the radar branch compensates for the accuracy loss incurred during projection into the BEV space, thereby yielding more pronounced enhancements in these adverse scenarios.

D. Ablation on Modules

In this section, we conduct a series of ablation experiments on the VoD dataset and TJ4DRadSet to validate the effectiveness of the core modules and parameter selection proposed in our method.

We first conducted an ablation study on the core modules. Our analysis focuses primarily on three components: the Dynamic Depth Compensation(DDC) module, OCC weight mask and the GFSCA fusion module.

As shown in Table V, when the DDC Module is activated with only the DBSCAN-based clustering diffusion, we observe a significant improvement in the detection of large objects, particularly in the Car category. This is because larger objects typically receive more radar point reflections. The more points available, the broader the clustering scope, leading to richer depth estimation and better augmentation. However, for smaller objects such as pedestrians, the associated radar points are often sparse and may be treated as noise or incorrectly clustered with nearby larger objects. This results in a performance drop in small-object detection.

When using only the RCS-guided diffusion in the DDC module, we observe an overall performance improvement across all categories. This is attributed to the fact that this module relies solely on each point's intrinsic attributes and does not create uncertain inter-point connections. As a result, even small objects with few radar points can benefit from accurate depth propagation. Small and distant objects typically lack sufficient point cloud support for robust representation. By enhancing the depth at these sparse locations, we enable the image features—which have strong semantic but weak spatial accuracy—to be more effectively utilized, thereby improving the accuracy of small-object detection at longer ranges.

Given that DBSCAN-based clustering diffusion and RCS-guided diffusion yield different improvements for objects of different sizes, using fixed weights to fuse the generated depth is suboptimal. We therefore introduce a learned dynamic weighting factor λ, which enables the model to adaptively fuse the two types of enhanced depth features. This dynamic weight is computed by convolving the image-branch feature map, generating spatially-aware weights at each location in the 2D plane.

TABLE VI

ABLATION STUDIES ON FUSION MODULE. EXPERIMENTS ARE CONDUCTED ON VOD DATASET.

Fusion	Car	Ped	Cyc	mAP_{3D}	Params	Latency(ms)
SENet [43]	40.46	41.52	68.93	50.3	37.51M	140
GFSCA (grid = 4)	41.03	41.97	68.83	50.61	35.74M	138
GFSCA (grid = 8)	41.11	42.29	69.84	51.08	35.74M	138
GFSCA (grid = 16)	37.95	39.67	67.73	48.45	35.73M	133

To further mitigate erroneous depth estimations from the image branch, we incorporate an OCC weight mask derived from the point cloud BEV occupancy situation into the image feature map. Since this occupancy weight mask originates from radar data with accurate spatial information, it provides a reliable reference in the BEV space. The mask encourages the network to focus more on regions where radar confidently detects objects, and suppress attention to regions with abnormal or uncertain depth projections from the image branch.

GFSCA Fusion leverages shared convolutional layers and attention mechanisms to generate attention maps along both the height and width dimensions. These attention maps are used to reweight the input feature maps, enhancing the representation of important features while suppressing less relevant ones. Traditional single-channel or single-spatial attention mechanisms are limited in their ability to capture multi-dimensional global information. In contrast, the GFSCA Fusion module integrates multi-dimensional global context with attention mechanisms, significantly improving the accuracy of multi-modal feature fusion and enhancing the performance across all object categories.

As can be seen from Fig. 9(a), for larger targets, DBSCAN-based clustering diffusion significantly improves performance, while the combined introduction of RCS and DBSCAN diffusion yields even better results. This is because larger targets are easier to cluster and have higher RCS values, making it more feasible to generate accurate depth compensation. The incorporation of GFSCA Fusion leads to substantial improvements across all categories, particularly for pedestrians, which aligns with the fact that introducing dense camera information enhances the perception capability for small targets.

Enhancing global feature representation typically comes with increased computational costs. As shown in Table VI, the proposed grouped feature processing strategy minimizes complexity while preserving global interaction capability in the BEV feature space. Compared to SE-Net, our method achieves a better balance between detection performance and inference speed when using a group size of grid = 8. Thanks to the grouping operation, the GFSCA Fusion module achieves a 0.78% mAP improvement while reducing the parameter count by 1.77M.

When the group size is set to grid = 4, the inference time and parameter count show no significant gain, but the performance drops slightly by 0.47%. This is because too few groups limit the network's ability to observe a sufficiently diverse set of feature slices, making it difficult to fully train the shared cross-dimension interaction module. On the other hand, when the group size is increased to grid = 16 or beyond, inference time decreases significantly, but this comes at the cost of a substantial performance drop. The excessive fragmentation of feature slices leads to a loss of implicit relationships between different feature channels, thereby reducing the network's representational capacity.

TABLE VII

ABLATION STUDY ON DBSCAN HYPERPARAMETERS. EXPERIMENTS ARE CONDUCTED ON VOD.

Eps(m)	MiniPts	Car	Ped	Cyc	mAP_{3d}
0.6	3	40.15	42.53	68.97	50.55
1	3	41.11	42.29	69.84	51.08
1.4	3	41.23	42.05	69.77	51.01
1	2	36.21	37.11	48.91	40.74
1	4	41.27	27.3	60.85	43.14

As Table VII shown, to validate the rationale behind the selection of the MinPts and Eps parameters, a comparative analysis was conducted by adjusting these hyperparameters. Since the object category cannot be known in advance prior to clustering, it is necessary to identify hyperparameters that are suitable for multi-class targets. With MinPts fixed at 3, we adjusted Eps for experimentation, selecting values of 0.6, 1.0, and 1.4 for validation. When Eps was set to 0.6 m, the smaller radius aligned well with the compact dimensions of pedestrians; however, it led to the misclassification of different parts of large objects (such as the front and rear of a car) into separate clusters, resulting in performance degradation for the car and cyc categories. When Eps was increased to 1.4 m, the small physical size and highly concentrated point clouds of pedestrians and bicycles caused them to be easily merged with surrounding contexts (e.g., a pedestrian next to a bicycle), also leading to performance decline. In contrast, the larger neighborhood radius better accommodated the large-scale point clouds of car.

With eps fixed at 1.0 m, variations in MinPts significantly influenced the final performance. Setting MinPts to 2 caused a substantial drop in performance across all categories, this is attributed to the presence of numerous noisy points in 4D-MMW radar data. When MinPts was increased to 4, the degree of performance degradation was notably reduced. This can be explained by the filtering effect of a higher MinPts value, which adapts to the unstable nature of radar point clouds. Nonetheless, this setting also filtered out many points containing valid object information, thereby exacerbating the inherent sparsity issue of the point clouds.

Based on the above analysis, we ultimately selected Eps = 1.0 m and MinPts = 3 as the final hyperparameters.

V. CONCLUSION

In this paper, we propose DDCFusion, a 4D-MMW radar-camera fusion-based 3D object detection method. By ex-

tending BEVFusion with the Dynamic Depth Compensation module, radar branch with occupancy weight mask and the GFSCA Fusion module, our method achieves superior performance compared to existing radar-camera (R+C) fusion approaches. Radar branch extracts semantic and geometric features from the 4D-MMW radar point cloud. Radar occupancy weight mask is used to selectively enhance the image feature maps in the BEV space, leveraging the radar's spatial confidence to refine image features. Dynamic Depth Compensation module strengthens the interaction between camera and radar modalities. By leveraging the spatial cues provided by dynamic compensation from radar DBSCAN-based and RCS-guided depth diffusion features, it enhances the accuracy of image-to-BEV projection and facilitates more precise cross-modal alignment. This demonstrates the untapped potential of exploiting the intrinsic properties of 4D-MMW radar point clouds. By propagating known radar points to a broader spatial range through these additional modules, we effectively increase the density of 4D-MMW radar point clouds, providing a valuable supplement to the camera branch and enhancing overall detection performance.

Future work will focus on four key directions. First, we will investigate how to more effectively utilize the temporal information from 4D-MMW radar, including Doppler velocity and point cloud evolution across consecutive frames, to enhance the estimation of dynamic object motion states (such as acceleration and turning) and improve trajectory prediction performance. Second, we plan to refine the depth diffusion component within the DDC module by introducing semantic constraints from an image segmentation network, which is expected to better regulate the diffusion scope and further improve the accuracy of the view transformation. Third, investigating techniques where RCS characteristics also directly influence the scattering and weighting of features within the BEV grid itself. This would enable the model to intelligently distribute radar-derived features across the BEV space based on the physical properties of objects, potentially leading to a more refined and semantically informed. Finally, we will further explore the potential of multi-modal fusion, including tri-modal camera–radar–LiDAR systems and camera-only configurations with learned pseudo-radar representations.

APPENDIX I
DEPTH PROBABILITY DISTRIBUTION ESTIMATION

$$D(u,v) = Softmax(f_{depth}(I(u,v))) \tag{8}$$

$I(u,v)$: The image feature vector at pixel location (u,v), extracted from the image backbone. It encodes semantic and contextual information for that spatial position. f_{depth}: The depth prediction network, typically implemented as a shallow MLP [38] or convolutional head. It maps the image feature $I(u,v)$ to a set of depth logits across predefined depth bins. $Softmax$: A normalization function applied to the depth logits output by f_{depth}, converting them into a probability distribution over depth.

APPENDIX II
RADAR-BASED FEATURE MASKING MECHANISM

$$mask_{occ} = Sigmoid(conv(F_{Radar\_bev})) \tag{9}$$

$$F_{img\_occ} = F_{img\_bev} \odot mask_{occ} \tag{10}$$

Here, F_{img_bev} denotes the original image feature map in BEV space, which is a dense and continuous representation. In contrast, F_{img_occ} represents a sparsely populated feature map, stored in a continuous format but selectively activated based on the radar-derived OCC weight mask.

APPENDIX III
SUPPLEMENTARY EQUATIONS FOR THE GFSCA FUSION MODULE

$$F_{BEV}^{in} = Cat(F_{Radar\_bev}, F_{img\_bev}) \in \mathrm{R}^{B\times C\times H\times W} \tag{11}$$

$$F_{BEV}^{in} \in \mathrm{R}^{B\times G\times \frac{C}{G}\times H\times W} \tag{12}$$

$$\begin{aligned} F_{h,avg} &= conv(Height\_AvgPool(F_{BEV}^{in}) \in \mathrm{R}^{B\times G\times \frac{C}{G}\times H\times 1}),\\ F_{h,max} &= conv(Height\_MaxPool(F_{BEV}^{in}) \in \mathrm{R}^{B\times G\times \frac{C}{G}\times H\times 1}),\\ F_{w,avg} &= conv(Width\_AvgPool(F_{BEV}^{in}) \in \mathrm{R}^{B\times G\times \frac{C}{G}\times 1\times W}),\\ F_{w,max} &= conv(Width\_MaxPool(F_{BEV}^{in}) \in \mathrm{R}^{B\times G\times \frac{C}{G}\times 1\times W}) \end{aligned} \tag{13}$$

$$\begin{aligned} W_h &= \sigma\left(F_{h,avg} + F_{h,\max}\right) \in \mathrm{R}^{B\times G\times \frac{C}{G}\times H\times 1},\\ W_w &= \sigma\left(F_{w,avg} + F_{w,\max}\right) \in \mathrm{R}^{B\times G\times \frac{C}{G}\times 1\times W} \end{aligned} \tag{14}$$

$$F_{BEV}^{\text{out}} = F_{BEV}^{\text{in}} \times W_h \times W_w \in \mathrm{R}^{B\times C\times H\times W} \tag{15}$$

REFERENCES

[1] S. Sun, A. P. Petropulu, and H. V. Poor, "Mimo radar for advanced driver-assistance systems and autonomous driving: Advantages and challenges," *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 98–117, 2020.

[2] A. Caillot, S. Ouerghi, P. Vasseur, R. Boutteau, and Y. Dupuis, "Survey on cooperative perception in an automotive context," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 204–14 223, 2022.

[3] J. Mao, S. Shi, X. Wang, and H. Li, "3d object detection for autonomous driving: A review and new outlooks," *arXiv preprint arXiv:2206.09474*, vol. 1, no. 1, p. 1, 2022.

[4] L. Wang, X. Zhang, Z. Song, J. Bi, G. Zhang, H. Wei, L. Tang, L. Yang, J. Li, C. Jia *et al.*, "Multi-modal 3d object detection in autonomous driving: A survey and taxonomy," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 7, pp. 3781–3798, 2023.

[5] H. Wang, X. Chen, Q. Yuan, and P. Liu, "A review of 3d object detection based on autonomous driving," *The Visual Computer*, vol. 41, no. 3, pp. 1757–1775, 2025.

[6] Z. Han, J. Wang, Z. Xu, S. Yang, L. He, S. Xu, J. Wang, and K. Li, "4d millimeter-wave radar in autonomous driving: A survey," *arXiv preprint arXiv:2306.04242*, 2023.

[7] Y. Li, Z. Ge, G. Yu, J. Yang, Z. Wang, Y. Shi, J. Sun, and Z. Li, "Bevdepth: Acquisition of reliable depth for multi-view 3d object detection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 2, 2023, pp. 1477–1485.

[8] A. W. Harley, Z. Fang, J. Li, R. Ambrus, and K. Fragkiadaki, "Simple-bev: What really matters for multi-sensor bev perception?" in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 2759–2765.

[9] P. Li, W. Shen, Q. Huang, and D. Cui, "Dualbev: Cnn is all you need in view transformation," *arXiv e-prints*, pp. arXiv–2403, 2024.

[10] C. Yang, Y. Chen, H. Tian, C. Tao, X. Zhu, Z. Zhang, G. Huang, H. Li, Y. Qiao, L. Lu *et al.*, "Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 830–17 839.

[11] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *Conference on Robot Learning*. PMLR, 2022, pp. 180–191.

[12] Y. Liu, J. Yan, F. Jia, S. Li, A. Gao, T. Wang, and X. Zhang, "Petrv2: A unified framework for 3d perception from multi-camera images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3262–3272.

[13] M. Drobnitzky, J. Friederich, B. Egger, and P. Zschech, "Survey and systematization of 3d object detection models and methods," *The Visual Computer*, vol. 40, no. 3, pp. 1867–1913, 2024.

[14] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, p. 3337, 10 2018.

[15] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[16] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3d object detection and tracking," *CVPR*, 2021.

[17] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[18] X. Chen, T. Zhang, Y. Wang, Y. Wang, and H. Zhao, "Futr3d: A unified sensor fusion framework for 3d detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2023, pp. 172–181.

[19] A. Palffy, E. Pool, S. Baratam, J. F. P. Kooij, and D. M. Gavrila, "Multi-class road user detection with 3+1d radar in the view-of-delft dataset," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4961–4968, 2022.

[20] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Dbscan revisited, revisited: why and how you should (still) use dbscan," *ACM Transactions on Database Systems (TODS)*, vol. 42, no. 3, pp. 1–21, 2017.

[21] L. Zheng, Z. Ma, X. Zhu, B. Tan, S. Li, K. Long, W. Sun, S. Chen, L. Zhang, M. Wan, L. Huang, and J. Bai, "Tj4dradset: A 4d radar dataset for autonomous driving," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, 2022, pp. 493–498.

[22] B. Xu, X. Zhang, L. Wang, X. Hu, Z. Li, S. Pan, J. Li, and Y. Deng, "Rpfa-net: A 4d radar pillar feature attention network for 3d object detection," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 3061–3066.

[23] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 529–10 538.

[24] Y. Song and L. Wang, "Bico-fusion: Bidirectional complementary lidar-camera fusion for semantic-and spatial-aware 3d object detection," *IEEE Robotics and Automation Letters*, 2024.

[25] Y. Kim, J. Shin, S. Kim, I.-J. L. J. W. Choi, and D. Kum, "Crn: Camera radar net for accurate, robust, efficient 3d perception supplementary material."

[26] F. Fent, A. Palffy, and H. Caesar, "Dpft: Dual perspective fusion transformer for camera-radar-based object detection," *IEEE Transactions on Intelligent Vehicles*, 2024.

[27] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[28] V. A. Sindagi, Y. Zhou, and O. Tuzel, "Mvx-net: Multimodal voxelnet for 3d object detection," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7276–7282.

[29] T. Huang, Z. Liu, X. Chen, and X. Bai, "Epnet: Enhancing point features with image semantics for 3d object detection," in *Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part XV 16*. Springer, 2020, pp. 35–52.

[30] C. Wang, C. Ma, M. Zhu, and X. Yang, "Pointaugmenting: Cross-modal augmentation for 3d object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 11 794–11 803.

[31] J. Philion and S. Fidler, "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*. Springer, 2020, pp. 194–210.

[32] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1090–1099.

[33] L. Zheng, S. Li, B. Tan, L. Yang, S. Chen, L. Huang, J. Bai, X. Zhu, and Z. Ma, "Rcfusion: Fusing 4-d radar and camera with bird's-eye view features for 3-d object detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–14, 2023.

[34] Z. Lin, Z. Liu, Z. Xia, X. Wang, Y. Wang, S. Qi, Y. Dong, N. Dong, L. Zhang, and 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. 14 928–14 937.

[35] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[36] X. Shi, Z. Hao, and Z. Yu, "Spikingresformer: bridging resnet and vision transformer in spiking neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 5610–5619.

[37] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[38] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit *et al.*, "Mlp-mixer: An all-mlp architecture for vision," *Advances in neural information processing systems*, vol. 34, pp. 24 261–24 272, 2021.

[39] J. Deng, S. Shi, P. Li, W. Zhou, Y. Zhang, and H. Li, "Voxel r-cnn: Towards high performance voxel-based 3d object detection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 2, 2021, pp. 1201–1209.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[41] O. D. Team, "Openpcdet: An open-source toolbox for 3d object detection from point clouds," https://github.com/open-mmlab/OpenPCDet, 2020.

[42] J. Liu, Q. Zhao, W. Xiong, T. Huang, Q.-L. Han, and B. Zhu, "Smurf: Spatial multi-representation fusion for 3d object detection with 4d imaging radar," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 799–812, 2024.

[43] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.