RCBEVDet++: Toward High-accuracy Radar-Camera Fusion 3D Perception Network

Zhiwei Lin∗, Zhe Liu∗, Yongtao Wang✉, Le Zhang, and Ce Zhu

Abstract—Perceiving the surrounding environment is a fundamental task in autonomous driving. To obtain highly accurate and robust perception results, modern autonomous driving systems typically employ multi-modal sensors, such as LiDAR, multi-view cameras, and millimeter-wave radar, to collect comprehensive environmental data. Among these, the radar-camera multi-modal perception system is especially favored for its excellent sensing capabilities and cost-effectiveness. However, the substantial modality differences between millimeter-wave radar and multi-view camera sensors pose significant challenges in fusing information from these two types of sensors. To address this problem, this paper presents RCBEVDet, a radar-camera fusion 3D object detection framework. Specifically, RCBEVDet is developed from an existing camera-based 3D object detection model, supplemented by a specially designed radar feature extractor, RadarBEVNet, and a radar-camera Cross-Attention Multi-layer Fusion (CAMF) module. Firstly, RadarBEVNet encodes sparse radar points into a dense bird's-eye-view (BEV) feature using a dual-stream radar backbone and a Radar Cross Section (RCS) aware BEV encoder. Secondly, the CAMF module utilizes a deformable attention mechanism to align radar and camera BEV features and adopts channel and spatial fusion layers to fuse these multi-modal features. To further enhance RCBEVDet's capabilities, we introduce RCBEVDet++, which advances the CAMF through sparse fusion, supports query-based multi-view camera perception models, and adapts to a broader range of perception tasks. Extensive experiments on the nuScenes dataset demonstrate that our method integrates seamlessly with existing camera-based 3D perception models and improves their performance across various perception tasks. Furthermore, our method achieves state-of-the-art radar-camera fusion results in 3D object detection, BEV semantic segmentation, and 3D multi-object tracking tasks. Notably, with ViT-L as the image backbone, RCBEVDet++ achieves 72.73 NDS and 67.34 mAP in 3D object detection without test-time augmentation or model ensembling. The source code and models will be released at https://github.com/VDIGPKU/RCBEVDet.

Index Terms—Autonomous driving, multi-modal, millimeter-wave radar, multi-view cameras, 3D perception

✦

1 INTRODUCTION

AUTONOMOUS driving aims to improve safety, efficiency, and convenience in transportation by developing systems that allow vehicles to operate without human intervention [2], [3]. A major challenge for these systems is to perceive the surrounding environment as comprehensively as humans do, which is crucial for accurate trajectory prediction and motion planning. To achieve this, modern autonomous driving systems primarily employ three types of sensors, *e.g.*, multi-view cameras, millimeter-wave radar, and LiDAR, to gather information about the surrounding environment.

Among these types of sensors, the LiDAR sensor provides detailed geometric information that significantly enhances the perception process, resulting in optimal performance [4]. However, high-quality LiDAR sensors are expensive, which increases manufacturing costs. In contrast, multi-view cameras and millimeter-wave radar sensors offer more economical alternatives for both manufacturers and users. Compared with LiDAR, multi-view cameras capture intricate details such as color and texture, offering high-resolution semantic information, while the millimeter-wave radar sensor is superior in distance and velocity estimation [5]

and performs reliably under diverse weather and lighting conditions [6], [7]. Besides, advancements in 4D millimeter-wave radar technology are gradually overcoming its limitation of sparse radar points, positioning it as a potential substitute [8]. Despite these advantages, a notable performance gap remains between LiDAR and camera or radar-based perception models. A practical and effective strategy to bridge this gap involves integrating millimeter-wave radar with multi-view cameras, which complement each other and result in a more comprehensive and reliable perception.

To fuse radar and image data, recent works [9], [10] primarily adopt the BEVFusion pipeline [4], [11] by projecting both multi-view image features and radar features into a bird's eye view (BEV). However, simple fusion techniques such as concatenation or summation, as employed by BEVFusion, fail to address spatial misalignment between multi-view images and radar inputs. Moreover, most radar-camera fusion methods [12], [13], [14] still utilize encoders originally designed for LiDAR points, such as PointPillars, to extract radar features. Though these methods yield commendable results, the LiDAR-specific encoders they use do not account for unique radar characteristics, such as the Radar Cross Section (RCS), resulting in sub-optimal performance.

In this paper, we introduce RCBEVDet, a novel framework that effectively fuses radar and camera features in BEV space for the 3D object detection task. To address the unique characteristics of radar inputs, we specially design RadarBEVNet for efficient radar BEV feature extraction. Specifically, RadarBEVNet begins by encoding radar inputs into distinct point-based and transformer-based representations through a dual-stream radar encoder. Besides, an Injection and Extraction module is implemented to

- *Corresponding author: Yongtao Wang. * indicates equal contribution.*
- *Zhiwei Lin and Yongtao Wang are with Wangxuan Institute of Computer Technology, Peking University, Beijing 100080, China. E-mail: {zwlin, wyt}@pku.edu.cn*
- *Zhe Liu, Le Zhang, and Ce Zhu are with School of Information and Communication Engineering, University of Electronic Science and Technology of China, Sichuan 611731, China. E-mail: liuzhe@std.uestc.edu.cn, {lezhang,eczhu}@uestc.edu.cn*
- *A preliminary version of this manuscript was published in [1]*

integrate features from these two representations. Subsequently, these features are transformed into BEV features via RCS-aware scattering, which leverages RCS as a prior for object size and allocates point features throughout the BEV space. In addition to RadarBEVNet, RCBEVDet integrates a Cross-Attention Multi-layer Fusion Module (CAMF) to achieve a robust fusion of radar and camera features within the BEV space. To be more specific, CAMF employs multi-modal cross-attention to adaptively correct coordinate mismatches between the two types of BEV features. Then, channel and spatial fusion layers are applied to further consolidate the multi-modal features, enhancing the overall detection performance.

To fully leverage the capabilities of RCBEVDet, we have upgraded the CAMF module with sparse fusion to support query-based multi-view camera perception models. Additionally, we have broadened RCBEVDet's functionality to encompass a broader range of perception tasks, including 3D object detection, BEV semantic segmentation, and 3D multi-object tracking. This enhanced framework is named RCBEVDet++. Specifically, to adapt the CAMF module for query-based multi-view camera methods, which lack explicit camera BEV features, we replace the original camera BEV features with camera object queries that incorporate 3D coordinates. This modification develops a new query component in our multi-modal cross-attention mechanism. Then, we perform a project-and-sample process, where camera object queries are projected into the BEV space and matched with corresponding radar features to form radar object queries. Subsequently, the multi-modal queries are aligned using deformable cross-attention. Finally, the adjusted multi-modal queries are concatenated and sent to a simple linear layer for effective feature fusion, boosting perception performance across the expanded range of tasks.

The main contributions of this paper are listed as follows:

- We introduce RCBEVDet, a radar-camera fusion framework designed for highly accurate and robust 3D object detection, which consists of RadarBEVNet for radar BEV feature extraction and the Cross-Attention Multi-layer Fusion Module (CAMF) for robust radar-camera feature fusion in BEV space.
- Based on RCBEVDet, we further propose RCBEVDet++ perception framework, which extends the CAMF module to accommodate query-based multi-view camera perception models and unleashes the full potential of RCBEVDet in various 3D perception tasks.
- On the nuScenes benchmark, RCBEVDet improves the performance of camera-based 3D object detectors and demonstrates robust capabilities against sensor failure cases. Additionally, RCBEVDet++ further enhances camera-based perception models, achieving state-of-the-art results in radar-camera multi-modal 3D object detection, BEV semantic segmentation, and 3D multi-object tracking tasks.

2 RELATED WORK

2.1 Camera-based 3D Perception

3D object detection, BEV semantic segmentation, and 3D multi-object tracking are three fundamental perception tasks for autonomous driving. Currently, many 3D multi-object tracking methods often adopt the Tracking-By-Detection framework, which utilizes results from 3D object detection to associate objects. These tracking methods focus on object matching rather than efficiently

processing input images. Besides, more accurate detection results can bring higher tracking performance. Therefore, in this section, we mainly discuss more diverse 3D object detection and BEV semantic segmentation methods that process multi-frame multi-view camera inputs. Specifically, 3D object detection aims to predict the location and category of 3D objects, while semantic segmentation integrates vehicle recognition, semantic lane map prediction, and drivable area estimation tasks. However, detecting objects and segmenting maps in 3D space using camera images is challenging due to insufficient 3D information. In recent years, numerous studies [15], [16], [17], [18], [19], [20] have made substantial efforts to address this issue, including inferring depth from images [21], utilizing geometric constraints and shape priors [22], designing specific loss functions [23], [24], and exploring joint 3D detection and reconstruction optimization [25]. More recently, multi-view sensors have become a popular configuration for autonomous vehicles, providing more comprehensive surrounding information. The emergence of multi-view camera datasets [26], [27] has led to the development of multi-view 3D object detection and BEV semantic segmentation methods [16], [17], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], which can be broadly categorized into two approaches: geometry-based methods and transformer-based methods.

2.1.1 Geometry-based Methods

Geometry-based multi-view 3D object detection and BEV semantic segmentation predominantly utilize depth prediction networks to estimate the depth distribution in images. This facilitates the conversion of extracted 2D image features into 3D camera frustum features. Subsequently, operations such as Voxel Pooling are employed to construct features in the voxel or BEV space.

Specifically, as a pioneering work, Lift-Splat-Shoot (LSS) [28] first employs a lightweight depth prediction network to explicitly estimate the depth distribution and a context vector for each image. Then, the outer product of the depth and context vector determines the feature at each point in 3D space along the perspective ray, enabling the effective transformation of image features into BEV features. Building on LSS, FIERY [38] introduces a future instance prediction model based on BEV, capable of predicting the future instances of dynamic agents and their motions. BEVDet [16] extends the viewpoint transformation technique from LSS to detect 3D objects using BEV features. Additionally, BEVDepth [29] leverages explicit depth information from LiDAR as supervision to enhance depth estimation and incorporates camera extrinsic parameters as a prior for depth estimation. Based on BEVDet, BEVDet4D [30] performs spatial alignment of BEV features across historical frames, significantly improving detection performance. Furthermore, SOLOFusion [39] proposes to fuse high-resolution short-term and low-resolution long-term features, enhancing the inference speed of 3D detection with long-term temporal inputs.

2.1.2 Transformer-based Methods

Transformer-based methods leverage attention mechanisms to project predefined queries onto multi-view image planes using coordinate transformation matrices, subsequently updating the query features with multi-view image features. Specifically, the pioneering work DETR3D [31] employs a transformer decoder for 3D object detection, developing a top-down framework and utilizing a set-to-set loss to measure the difference between ground truth and predictions. Similarly, CVT [35] introduces a simple BEV
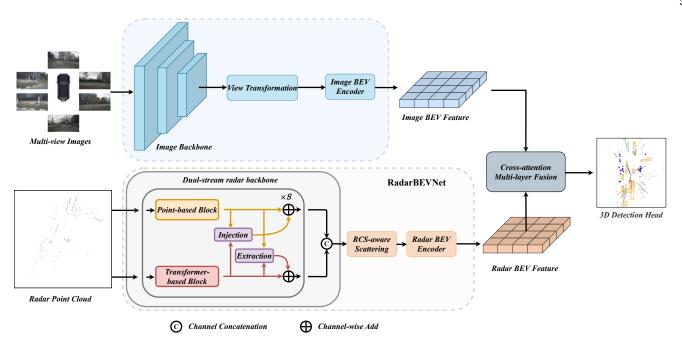
Fig. 1. **Overall pipeline of RCBEVDet.** Firstly, multi-view images are encoded and transformed into image BEV features. Concurrently, radar point clouds are processed by the proposed RadarBEVNet to extract radar BEV features. Subsequently, features from both radar and cameras are dynamically aligned and aggregated using the Cross-Attention Multi-layer Fusion (CAMF) module. The resulting semantically rich multi-modal feature is then utilized for the 3D object detection task.

semantic segmentation baseline using a cross-view transformer architecture. Following this, BEVformer [17] constructs dense BEV queries and incorporates multi-scale deformable attention to map multi-view image features to these dense queries. Moreover, PETR [32] generates multi-view image features with explicit position information derived from 3D coordinates. Building on PETR, PETRv2 [40] integrates temporal fusion across multiple frames and extends 3D positional embedding with time-aware modeling. Additionally, Sparse4D [41] allocates and projects multiple 4D key points for each 3D anchor to generate different views, aspect ratio, and timestamp features, which are then fused hierarchically to improve the overall image feature representation. Sparse4Dv2 [42] extends Sparse4D with a more efficient time fusion module and introduces camera parameter encoding and dense depth supervision. More recently, StreamPETR [34] utilizes sparse object queries as intermediate representations to capture temporal information, and SparseBEV [33] incorporates a scale-adaptive self-attention module and an adaptive spatio-temporal sampling module to dynamically capture BEV and temporal information.

2.2 Radar-camera 3D Perception

Millimeter-wave radar is a widely used sensor in autonomous vehicles due to its cost-effectiveness, long-range perception, Doppler velocity measurement, and robustness against adverse weather conditions. Although millimeter-wave radar data generally includes distance, angle, and velocity information, it performs relatively poorly in measuring targets' pitch angles. Additionally, the inherent sparsity and lack of semantic information in millimeter-wave radar data pose challenges for pure radar-based 3D perception. As a result, millimeter-wave radar is frequently employed as an auxiliary modality to enhance the performance of multi-modal 3D perception systems.

In recent years, the combination of multi-view cameras and millimeter-wave radar sensors for 3D perception has attracted

significant attention, owing to the complementary nature of the information provided by these two modalities. Concretely, Radar-Net [43] introduces a multi-level radar-camera fusion pipeline to improve the accuracy of distant object detection and reduce velocity errors. CenterFusion [14] utilizes a keypoint detection network to generate initial 3D detection results from images and incorporates a pillar-based radar association module to refine these results by linking radar features with corresponding detected boxes. Similarly, MVFusion [44] achieves semantic alignment between cameras and millimeter-wave radar, enhancing the interaction between the two modalities. Additionally, Simple-BEV [45] investigates architecture designs and hyper-parameter settings for multi-sensor BEV perception systems. CRAFT [12] proposes a proposal-level fusion framework that employs a Soft-Polar-Association and Spatio-Contextual Fusion Transformer to efficiently exchange information between the camera and millimeter-wave radar. RADIANT [46] develops a network to estimate positional offsets between radar echoes and object centers and leverages radar depth information to enhance camera features. More recently, CRN [13] generates radar-augmented image features with radar depth information for multi-view transformation and incorporates a cross-attention mechanism to address spatial misalignment and information disparity between radar and camera sensors. RCFusion [9] utilizes radar PillarNet [47] to generate radar pseudo-images and presents a weighted fusion module to effectively fuse radar and camera BEV features. BEVGuide [36] builds on the CVT [35] framework and proposes a sensor-agnostic attention module based on BEV, facilitating BEV representations learning and understanding. BEVCar [37] introduces an innovative radar-camera fusion method for BEV map and object segmentation using an attention-based image lift strategy.

3 RCBEVDET: RADAR-CAMERA FUSION IN BEV FOR 3D OBJECT DETECTION

The overall pipeline of RCBEVDet is illustrated in Figure 1. Specifically, multi-view images are processed by an image encoder to extract features, which are then transformed into the image BEV feature using a view-transformation module. Concurrently, radar point clouds are encoded into the radar BEV feature by the proposed RadarBEVNet. Next, the image and radar BEV features are fused using the Cross-Attention Multi-layer Fusion module. Finally, the fused multi-modal BEV feature is employed for the 3D object detection task.

3.1 RadarBEVNet

Previous radar-camera fusion methods typically utilize radar encoders designed for LiDAR point clouds, such as PointPillars [48]. In contrast, we introduce RadarBEVNet, specifically tailored for efficient radar BEV feature extraction. RadarBEVNet encodes sparse radar points into a dense BEV feature using a dual-stream radar backbone and an RCS-aware BEV encoder. More specifically, the dual-stream radar backbone processes radar points into two representations: a local point-based representation and a global transformer-based representation, employing an Injection and Extraction module to fuse these representations. The RCS-aware BEV encoder leverages RCS as an object size prior, distributing the feature of a single radar point across multiple pixels in the BEV space.

3.1.1 Dual-stream radar backbone

The dual-stream radar backbone consists of two components: a point-based backbone and a transformer-based backbone. The point-based backbone focuses on learning local radar features, while the transformer-based backbone captures global information.

For the point-based backbone, we adopt an architecture similar to PointNet [49]. As illustrated in Figure 2a, the point-based backbone comprises S blocks, each containing a multi-layer perceptron (MLP) and a max pooling operation. Specifically, the input radar point feature f is first processed by the MLP to increase its feature dimension. Then, the high-dimensional radar features are sent to the MaxPool layer with a residual connection. The whole process can be formulated as follows:

$$f = \text{Concat}[\text{MLP}(f), \text{MaxPool}(\text{MLP}(f))]. \quad (1)$$

As for the transformer-based backbone, it comprises S standard transformer blocks [50], [51], which includes an attention mechanism, a feed-forward network, and normalization layers, as shown in Figure 2b. Due to the extensive range of autonomous driving scenarios, optimizing the model directly using standard self-attention can be challenging. To address this issue, we propose a distance-modulated self-attention mechanism (DMSA) to facilitate model convergence by aggregating neighbor information during the early training iterations. More specifically, given the coordinates of N radar points, we first calculate the pair-distance $D \in \mathbb{R}^{N \times N}$ between all points. Then, we generate the Gaussian-like weight map G according to the pair-distance D as follows:

$$G_{i,j} = \exp(-D_{i,j}^2/\sigma^2), \quad (2)$$

where σ is a learnable parameter to control the bandwidth of the Gaussian-like distribution. Essentially, the Gaussian-like weight
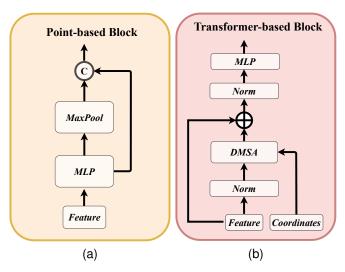

Fig. 2. **Architecture of the dual-stream radar backbone.** (a) Point-based Block. (b) Transformer-based Block.

map G assigns high weight to spatial locations near the point and low weight to positions far from the point. We can modulate the attention mechanism with the generated weight G as follows:

$$\begin{aligned} \text{DMSA}(Q, K, V) &= \text{Softmax}(\frac{QK^\top}{\sqrt{d}} + \log G)V \\ &= \text{Softmax}(\frac{QK^\top}{\sqrt{d}} - \frac{1}{\sigma^2}D^2)V, \end{aligned} \quad (3)$$

where Q, K, and V denote query, key, and value in attention mechanism [50]. To ensure DMSA can be degraded to vanilla self-attention, we replace $1/\sigma$ with a trainable parameter β during the training. When $\beta = 0$, DMSA is degraded to the vanilla self-attention. We also investigate the multi-head DMSA. Each head has unshared β_i to control the receptive field of DMSA. The multi-head DMSA with H heads can be formulated as MultiHeadDMSA$(Q, K, V) = $ Concat$[head_1, head_2, ..., head_H]$, where

$$\begin{aligned} head_i &= DMSA(Q_i, K_i, V_i) \\ &= \text{Softmax}(\frac{Q_i K_i^\top}{\sqrt{d_i}} - \beta_i D^2)V_i. \end{aligned} \quad (4)$$

To enhance the interaction between radar features from the two different backbones, we introduce the Injection and Extraction module, which is based on cross-attention, as illustrated in Figure 3. This module is applied at each block of the two backbones.

Concretely, assuming the features from the i th block of the point-based and transformer-based backbone are f_p^i and f_t^i, respectively. In injection operation, we take f_p^i as the query and f_t^i as the key and value. We use multi-head cross-attention to inject transformer feature f_t^i into the point feature f_p^i, which can be formulated as follows:

$$f_p^i = f_p^i + \gamma \times \text{CrossAttention}(LN(f_p^i), LN(f_t^i)), \quad (5)$$

where *LN* is the LayerNorm and γ is a learnable scaling parameter.

Similarly, the extraction operation extracts the point feature f_p^i with cross-attention for the transformer-based backbone. The extraction operation is defined as follows:

$$f_t^i = \text{FFN}(f_t^i + \text{CrossAttention}(LN(f_t^i), LN(f_p^i))), \quad (6)$$
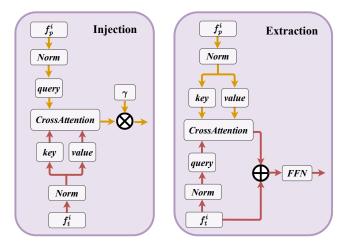
Fig. 3. **Architecture of the Injection and Extraction module.** The left figure shows the details of the injection operation. The right figure displays the structure of the extraction operation.

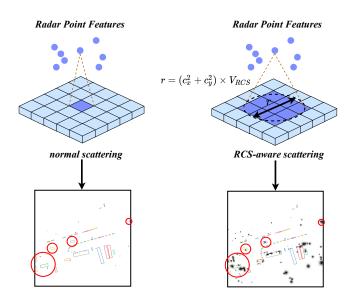$$r = (c_x^2 + c_y^2) \times V_{RCS}$$

Fig. 4. **Illustration of RCS-aware scattering.** RCS-aware scattering uses RCS as the object size prior to scattering the feature of one radar point to many BEV pixels.

where FFN is the FeedForward Network. The updated features, f_p^i and f_t^i, are sent to the next block of their corresponding backbone.

3.1.2 RCS-aware BEV encoder

Current radar BEV encoders typically scatter point features into BEV space based on the 3D coordinates of the points. However, this often results in a sparse BEV feature map, where most pixels contain zero values. This sparsity makes it difficult for some pixels to effectively aggregate features, potentially impairing detection performance. One solution is to increase the number of BEV encoder layers, but this can cause small object features to be smoothed out by background features. To address this issue, we propose an RCS-aware BEV encoder. The Radar Cross Section (RCS) measures an object's detectability by radar. For instance, larger objects generally produce stronger radar wave reflections, resulting in a larger RCS measurement. Thus, RCS can provide a rough estimate of an object's size. The key design of the RCS-aware BEV encoder is the RCS-aware scattering operation, which
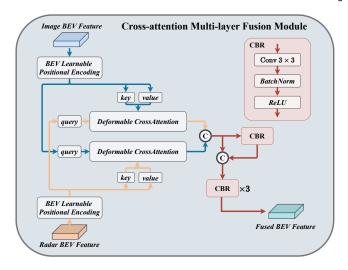

Fig. 5. **Cross-attention multi-layer fusion module.** The BEV features from radar and cameras are dynamically aligned using deformable cross-attention. Subsequently, the multi-modal BEV features are aggregated through a channel and spatial fusion module, which consists of several Convolution-BatchNorm-ReLU (CBR) blocks.

leverages the RCS as a prior estimate of the object's size, With this prior, the proposed scattering operation allows the feature of a single radar point to be scattered across multiple pixels in the BEV space, rather than being confined to a single pixel, as illustrated in Figure 4.

In particular, without loss of generality, given a specific radar point and its RCS value V_{RCS}, 3D coordinate $c = (c_x, c_y)$, BEV pixel coordinate $p = (p_x, p_y)$, and feature f, we scatter f to pixel p and nearby pixels, whose pixel distance with p is smaller than $(c_x^2 + c_y^2) \times V_{RCS}$. If a pixel in the BEV feature receives f from multiple radar features, we perform summation pooling to aggregate these features. This operation ensures that all relevant radar information is combined effectively, resulting in a comprehensive radar BEV feature f_{RCS}. Besides, we introduce a Gaussian-like BEV weight map for each point according to the RCS value as follows:

$$G_{x,y} = \exp\left(-\frac{(c_x - x)^2 + (c_y - y)^2}{\frac{1}{3}(c_x^2 + c_y^2) \times V_{RCS}}\right), \qquad (7)$$

where x, y are the pixel coordinates. The final Gaussian-like BEV weight map G_{RCS} is obtained by maximization over all Gaussian-like BEV weight maps. Subsequently, we concatenate f_{RCS} with G_{RCS} and send them to an MLP to get the final RCS-aware BEV feature as follows:

$$f'_{RCS} = \text{MLP}(\text{Concat}(f_{RCS}, G_{RCS})). \qquad (8)$$

After that, f'_{RCS} is concatenated with the original BEV feature and sent to the BEV encoder, *e.g.,* SECOND [52].

3.2 Cross-Attention Multi-layer Fusion Module

3.2.1 Multi-modal Feature Alignment with Cross-Attention

Radar point clouds often suffer from azimuth errors, causing radar sensors to detect points outside the boundaries of objects. Consequently, radar features generated by RadarBEVNet may be assigned to adjacent BEV grids, resulting in misalignment with BEV features from cameras. To address this issue, we use a cross-attention mechanism to dynamically align the multi-modal
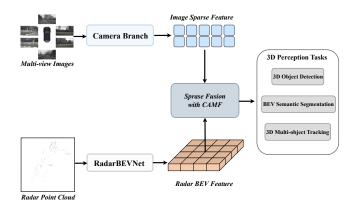
Fig. 6. **Overall pipeline of RCBEVDet++.** The image sparse feature and dense radar BEV feature are fused with the sparse fusion module. Then, the fused features are utilized for various 3D perception tasks, including 3D object detection, BEV semantic segmentation, and 3D multi-object tracking.
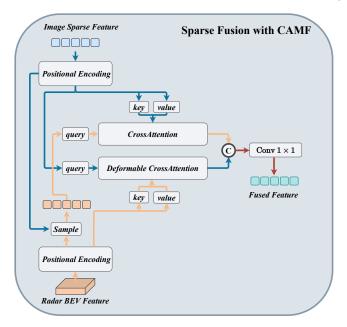

Fig. 7. **Sparse fusion with CAMF.** The dense radar BEV features and image sparse features are dynamically aligned with the cross-attention. Then, the multi-modal sparse features are fused with a simple Conv 1×1.

features. Since unaligned radar points typically deviate from their true position by a small distance, we propose employing deformable cross-attention [53] to accurately capture and correct these deviations. Besides, the deformable cross-attention can reduce the computational complexity of the vanilla cross-attention from $O(H^2W^2C)$ to $O(HWC^2K)$, where H and W represent the height and width of the BEV feature, C denotes the BEV feature channels, and K is the number of the reference points in deformable cross-attention.

Specifically, as shown in Figure 5, given camera and radar BEV features denoted by $F_c \in \mathbb{R}^{C_c \times H \times W}$, $F_r \in \mathbb{R}^{C_r \times H \times W}$, respectively, we first add learnable position embeddings to F_c and F_r. Then, F_r is transformed to queries z_{q_r} and reference points p_{q_r}, and F_c is viewed as keys and values. Next, we calculate the multi-head deformable cross-attention [53] by:

$$
\begin{aligned}
&\text{DeformAttn}\left(z_{q_r}, p_{q_r}, F_c\right) = \\
&\sum_{h=1}^{H} \mathrm{W}_h \left[\sum_{k=1}^{K} A_{hqk} \cdot \mathrm{W}'_h F_c \left(p_{q_r} + \Delta p_{hqk}\right) \right],
\end{aligned}
\tag{9}
$$

where h indexes the attention head, k indexes the sampled keys, K indicates the total sampled key number, Δp_{hqk} denotes the sampling offset, A_{hqk} represents attention weight calculated by z_{q_r} and F_c, W_h means output weight value to fuse multi-head attention, and W'_h is the value projection matrix at h^{th} head.

Similarly, we exchange F_r and F_c and conduct another deformable cross-attention to update F_r. Finally, the deformable cross-attention module in CAMF can be formulated as follows:

$$
\begin{cases}
F_c \leftarrow \text{DeformAttn}\left(z_{q_r}, p_{q_r}, F_c\right), \\
F_r \leftarrow \text{DeformAttn}\left(z_{q_c}, p_{q_c}, F_r\right).
\end{cases}
\tag{10}
$$

3.2.2 Channel and Spatial Fusion

After aligning the radar and camera BEV feature by cross-attention, we propose channel and spatial fusion layers to aggregate multi-modal BEV features, as illustrated in Figure 5. Specifically, we first concatenate two BEV features as $F_{multi} = [F_c, F_r]$. Then, F_{multi} is sent to a CBR block with a residual connection to obtain the fused feature. The CBR block is successively composed of a Conv 3×3, a Batch Normalization, and a ReLU activate function. After that, three CBR blocks are applied to further fuse the multi-modal features.

4 RCBEVDET++: RADAR-CAMERA SPARSE FUSION FOR 3D PERCEPTION

As illustrated in Figure 6, to unleash the full potential of RCBEVDet, we extend the CAMF module to accommodate sparse fusion with query-based multi-view camera perception models, which achieve higher accuracy than BEV-based methods. Besides, we apply RCBEVDet to more perception tasks, including 3D object detection, BEV semantic segmentation, and 3D multi-object tracking. To distinguish this updated version of RCBEVDet from the original one, we specially named it RCBEVDet++.

4.1 Sparse Fusion with CAMF

As shown in Figure 7, we adopt sparse fusion with CAMF to fuse dense radar BEV features and image sparse features. Specifically, we first replace the original image BEV features with image sparse features. Then, we perform a project-and-sample process to associate each image sparse feature with a radar feature using 3D absolute position. More specifically, we project the 3D absolute position into BEV and sample corresponding radar features with bilinear interpolation to obtain sparse radar features. Next, we utilize the positional encoding network composed of MLP to transform the 3D absolute position into 3D positional embedding and add them to the multi-modal queries. After that, to align the multi-modal mismatch, we adopt deformable cross-attention for sparse image features and dense radar BEV features, and simple cross-attention for sparse radar features and sparse image features as follows:

$$
\begin{cases}
F_c^{sparse} \leftarrow \text{DeformAttn}\left(z_{q_r}, p_{q_r}, F_c^{sparse}\right), \\
F_r^{sparse} \leftarrow \text{CrossAttn}\left(F_c^{sparse}, F_r\right).
\end{cases}
\tag{11}
$$

where F^{sparse} denotes the sparse feature for radar or image. Finally, we adopt a simple linear layer to fuse the sparse multi-modal features.

4.2 Downstream 3D Perception Tasks

Our RCBEVDet++ can generate high-quality multi-modal features, which can be leveraged for various 3D perception tasks, including 3D object detection, 3D multi-object tracking, and BEV semantic segmentation. To predict the 3D bounding boxes for 3D object detection, we adopt a query-based transformer decoder [33] and apply our sparse fusion with the CAMF module in every transformer decoder layer.

After that, we employ the tracking-by-detection framework for the 3D multi-object tracking task. Specifically, we perform velocity-based greedy distance matching; that is, we calculate the center distance of each object in multiple frames with the predicted velocity compensation and assign the same ID for objects with the minimum center distance in a greedy way.

For BEV semantic segmentation, we transform multi-modal features into dense BEV features since this requires a dense BEV map with categories. We follow the decoder architecture in CVT [35] to effectively decode dense BEV features to different maps with semantic representation. Additionally, we employ multiple heads to perform different types of BEV semantic segmentation tasks. Each head deals with one task, *e.g.,* vehicle segmentation. Finally, focal loss [54] with a sigmoid layer is used as the supervision for training.

5 EXPERIMENTS

In this section, we evaluate RCBEVDet and RCBEVDet++ through extensive experiments. In Section 5.1, we detail the experimental setup. In Section 5.2, we compare our method with state-of-the-art methods in three tasks, *i.e.,* 3D object detection, BEV semantic segmentation, and 3D multi-object tracking. In Section 5.3, we conduct an extensive ablation study to investigate individual components of RCBEVDet and RCBEVDet++. In Section 5.4, we discuss the task trade-off in BEV semantic segmentation of RCBEVDet++. In Section 5.5, we show the robustness of RCBEVDet. In Section 5.6, we demonstrate the model generalization of our method.

5.1 Implementation Details

5.1.1 Datasets and Evaluation Metrics

We conduct experiments on a popular large-scale autonomous diving benchmark, nuScenes [26], which includes 1000 driving scenes collected in Boston and Singapore. It comprises 850 scenes for training and validation and 150 scenes for testing. We report the results on the validation and test sets to compare with state-of-the-art methods and evaluate ablation results on the validation set.

For 3D object detection, nuScenes provides a set of evaluation metrics, including mean Average Precision (mAP) and five true positive (TP) metrics: ATE, ASE, AOE, AVE, and AAE, which measure translation, scale, orientation, velocity, and attribute errors, respectively. The overall performance is measured by the nuScenes Detection Score (NDS) that consolidates all error types:

$$\text{NDS} = \frac{1}{10}[5 \times \text{mAP} + \sum_{\text{mTP} \in \text{TP}} (1 - \min(1, \text{mTP}))]. \quad (12)$$

For BEV semantic segmentation, we use mean Intersection over Union (mIoU) across all segmentation categories as the metric, following the settings of LSS [28].

For 3D multi-object tracking, we follow the official nuScenes metrics, which use Average Multi-Object Tracking Accuracy (AMOTA) and Average Multi-Object Tracking Precision (AMOTP) over various recall thresholds. Concretely, AMOTA is defined as follows:

$$AMOTA = \frac{1}{n-1} \sum_{r \in \{\frac{1}{n-1}, \frac{2}{n-1} \dots 1\}} MOTAR_r, \quad (13)$$

$$MOTAR_r = \\ \max\left(0, 1 - \frac{IDS_r + FP_r + FN_r - (1-r)P}{rP}\right), \quad (14)$$

where P refers to the number of true positives of the current class, r is the scalar factor, IDS represents the number of identity switches, FP and FN denote the number of false positives and false negatives, respectively, and n is set to 40. For AMOTP, it can be formulated as follows:

$$AMOTP = \frac{1}{n-1} \sum_{r \in \{\frac{1}{n-1}, \frac{2}{n-1}, \dots, 1\}} \frac{\sum_{i,t} d_{i,t}}{\sum_t TP_t}, \quad (15)$$

where $d_{i,t}$ represents the position error of tracked object i at time t and TP_t indicates the number of matches at time t.

5.1.2 Architecture and Training Details

We adopt BEVDepth [29] with BEVPoolv2 [55] and SparseBEV [33] as the camera stream for RCBEVDet and RCBEVDet++, respectively. For BEVDepth, we follow BEVDet4D [30] to accumulate the intermediate BEV feature from multiple frames and add an extra BEV encoder to aggregate these multi-frame BEV features. For radar, we accumulate multi-sweep radar points and use RCS and Doppler speed as input features in the same manner as GRIFNet [56] and CRN [13]. We set the number of stages S in the dual-stream radar backbone to 3.

For the 3D object detection head, we use the center head from CenterPoint [57] for RCBEVDet and the sparse head from Sprase-BEV [33] for RCBEVDet++. For the BEV semantic segmentation head, we adopt a separate segmentation head for each task. For 3D multi-object tracking, we follow CenterPoint to utilize estimated velocity to track object centers across multiple frames in a greedy manner.

Our models are trained in a two-stage manner. In the first stage, we train the camera-based model following the standard implementations [29], [33]. In the second stage, we train the radar-camera fusion model. The weights of the camera stream are inherited from the first stage, and the parameters of the camera stream are frozen during the second stage. All models are trained for 12 epochs with AdamW [58] optimizer. To prevent overfitting, we apply various data augmentations, including image rotation, cropping, resizing, and flipping, as well as radar horizontal flipping, horizontal rotation, and coordinate scaling.

5.2 Comparison with State-of-the-Art

We compare our methods with cutting-edge camera-based and radar-camera multi-modal methods in three tasks: 3D object detection, BEV semantic segmentation, and 3D multi-object tracking.

TABLE 1
Comparison of 3D object detection results on nuScenes `val` set. 'C' and 'R' represent camera and radar, respectively.

Method	Input	Backbone	Image Size	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
CenterFusion [14]	C+R	DLA34	448×800	45.3	33.2	0.649	**0.263**	0.535	0.540	**0.142**
CRAFT [12]	C+R	DLA34	448×800	51.7	41.1	0.494	0.276	0.454	0.486	0.176
RCBEVDet (Ours)	C+R	DLA34	448×800	**56.3**	**45.3**	**0.492**	0.269	**0.449**	**0.230**	0.188
RCBEV4d [59]	C+R	Swin-T	256×704	49.7	38.1	0.526	0.272	0.445	0.465	0.185
RCBEVDet (Ours)	C+R	Swin-T	256×704	**56.2**	**49.6**	**0.496**	**0.271**	**0.418**	**0.239**	**0.179**
CRN [13]	C+R	ResNet-18	256×704	54.3	**44.8**	0.518	**0.283**	0.552	0.279	0.180
RCBEVDet (Ours)	C+R	ResNet-18	256×704	**54.8**	42.9	0.502	0.291	0.432	0.210	0.178
BEVDet [16]	C	ResNet-50	256×704	39.2	31.2	0.691	0.272	0.523	0.909	0.247
BEVDepth [29]	C	ResNet-50	256×704	47.5	35.1	0.639	**0.267**	0.479	0.428	0.198
SOLOFusion [39]	C	ResNet-50	256×704	53.4	42.7	0.567	0.274	0.411	0.252	0.188
StreamPETR [34]	C	ResNet-50	256×704	54.0	43.2	0.581	0.272	0.413	0.295	0.195
SparseBEV [33]	C	ResNet-50	256×704	54.5	43.2	0.606	0.274	0.387	0.251	0.186
CRN [13]	C+R	ResNet-50	256×704	56.0	49.0	0.487	0.277	0.542	0.344	0.197
RCBEVDet (Ours)	C+R	ResNet-50	256×704	56.8	45.3	**0.486**	0.285	**0.404**	**0.220**	0.192
RCBEVDet++ (Ours)	C+R	ResNet-50	256×704	**60.4**	**51.9**	0.488	0.268	0.408	0.221	**0.177**

TABLE 2
Comparison of 3D object detection results on nuScenes `test` set. 'L', 'C', and 'R' represent LiDAR, camera, and radar, respectively. † uses future frames. We do not use test-time data augmentation or ensemble for submission.

Method	Input	Backbone	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
CenterPoint [57]	L	VoxelNet	67.3	60.3	0.262	0.239	0.361	0.288	0.136
TransFusion-L [60]	L	VoxelNet	70.2	65.5	0.256	0.240	0.351	0.278	0.129
KPConvPillars [61]	R	Pillars	13.9	4.9	0.823	0.428	0.607	2.081	1.000
CenterFusion [14]	C+R	DLA34	44.9	32.6	0.631	0.261	0.516	0.614	0.115
RCBEV [59]	C+R	Swin-T	48.6	40.6	0.484	0.257	0.587	0.702	0.140
MVFusion [44]	C+R	V2-99	51.7	45.3	0.569	0.246	0.379	0.781	0.128
CRAFT [12]	C+R	DLA34	52.3	41.1	0.467	0.268	0.456	0.519	0.114
BEVFormer [17]	C	V2-99	56.9	48.1	0.582	0.256	0.375	0.378	0.126
PETRv2 [40]	C	V2-99	58.2	49.0	0.561	0.243	0.361	0.343	0.120
BEVDepth [29]	C	V2-99	60.5	51.5	0.446	0.242	0.377	0.324	0.135
BEVDepth [29]	C	ConvNeXt-B	60.9	52.0	0.445	0.243	0.352	0.347	0.127
BEVStereo [62]	C	V2-99	61.0	52.5	0.431	0.246	0.358	0.357	0.138
SOLOFusion [39]	C	ConvNeXt-B	61.9	54.0	0.453	0.257	0.376	0.276	0.148
CRN [13]	C+R	ConvNeXt-B	62.4	57.5	0.416	0.264	0.456	0.365	0.130
StreamPETR [34]	C	V2-99	63.6	55.0	0.493	0.241	0.343	0.243	0.123
SparseBEV [33]	C	V2-99	63.6	55.6	0.485	0.244	0.332	0.246	0.117
SparseBEV† [33]	C	V2-99	67.5	60.3	0.425	0.239	0.311	0.172	0.116
RCBEVDet (Ours)	C+R	V2-99	63.9	55.0	**0.390**	**0.234**	0.362	0.259	**0.113**
RCBEVDet++ (Ours)	C+R	V2-99	**68.7**	**62.6**	0.437	0.252	**0.181**	**0.203**	0.191
StreamPETR [34]	C	ViT-L	67.6	62.0	0.470	0.241	0.258	0.236	0.134
Far3D [63]	C	ViT-L	68.7	63.5	0.432	0.237	0.278	0.227	**0.130**
SparseBEV† [34]	C	ViT-L	70.2	-	-	-	-	-	-
RCBEVDet++† (Ours)	C+R	ViT-L	**72.7**	**67.3**	**0.341**	**0.234**	**0.241**	**0.147**	**0.130**

5.2.1 3D Object Detection

We provide 3D object detection results on `val` and `test` set in Tables 1 and 2, respectively.

As shown in Table 1, RCBEVDet outperforms previous radar-camera multi-modal 3D object detection methods across various backbones. Besides, based on SparseBEV, RCBEVDet++ surpasses CRN by 4.4 NDS, showing the effectiveness of our fusion method. Furthermore, RCBEVDet and RCBEVDet++ reduce the velocity error by 14.6% compared with the previous best method, demonstrating the efficiency of our method in utilizing radar information.

On `test` sets, with the V2-99 backbone, RCBEVDet++ improves the SparseBEV baseline by 5.1 NDS and 7.0 mAP and surpasses its offline version with future frames. Notably, RCBEVDet++ with the smaller V2-99 backbone achieves competitive performance compared to StreamPETR and Far3D with the larger backbone ViT-L. Moreover, RCBEVDet++ with the larger ViT-L backbone achieves 72.7 NDS and 67.3 mAP without test-time data augmentation, setting new state-of-the-art radar-camera 3D object detection results on nuScenes.

5.2.2 BEV Semantic Segmentation

We compare our method with state-of-the-art BEV semantic segmentation methods on `val` set in Tables 3. With the ResNet-101 backbone, RCBEVDet++ shows a favorable performance increase over CRN for the "Drivable Area" class (+0.6 IoU) and over BEVGuide for the "Lane" class (+6.3 IoU), respectively. In the combined evaluation for all tasks, RCBEVDet++ achieves

TABLE 3
Comparison of BEV semantic segmentation on nuScenes `val` set. 'C' and 'R' represent camera and radar, respectively. 'Driv. Area' denotes drivable area. 'mIoU' is the average IoU over Vehicle, Drivable Area, and Lane. † is a Simple-BEV [45] customized by BEVCar [37].

Methods	Input	Backbone	mIoU↑	Vehicle↑	Driv. Area↑	Lane↑
CVT [35]	C	EfficientNet	46.6	36.0	74.3	29.4
BEVFormer-S [17]	C	ResNet-101	48.4	43.2	80.7	21.3
CRN [13]	C+R	ResNet-50	-	58.8	82.1	-
Simple-BEV++† [37]	C+R	ResNet-101	55.4	52.7	77.7	35.8
BEVGuide [36]	C+R	EfficientNet	60.0	**59.2**	76.7	44.2
BEVCar [37]	C+R	ResNet-101	61.0	57.3	81.8	43.8
RCBEVDet++ (Ours)	C+R	ResNet-101	**62.8**	55.3	**82.7**	**50.5**

TABLE 4
Comparison of 3D multi-object tracking on nuScenes `test` set. 'L', 'C', and 'R' represent LiDAR, camera, and radar, respectively.

Methods	Input	AMOTA↑	AMOTP↓
CenterPoint [57]	L	63.8	0.555
UVTR [64]	C	51.9	1.125
Sparse4D [41]	C	51.9	1.078
ByteTrackV2 [65]	C	56.4	1.005
StreamPETR [34]	C	56.6	0.975
CRN [13]	C+R	56.9	0.809
RCBEVDet++ (Ours)	C+R	**59.6**	**0.713**

TABLE 5
Ablation of the main components of RCBEVDet. We successively add components to BEVDepth [29] to compose RCBEVDet. Each component improves the 3D detection performance consistently.

Model Configuration	Input	NDS↑	mAP↑
BEVDepth [29]	C	47.5	35.1
+ Temporal [30]	C	51.9 ↑4.4	40.5 ↑5.4
+ PointPillar+BEVFusion	C+R	53.6 ↑1.7	42.3 ↑1.8
+ RadarBEVNet	C+R	55.7 ↑2.1	45.3 ↑3.0
+ CAMF	C+R	56.4 ↑0.7	**45.6** ↑0.3
+ Temporal Supervision	C+R	**56.8** ↑0.4	45.3 ↓0.3

TABLE 6
Ablation of RadarBEVNet. The dual-stream radar backbone obtains marginal performance gains without the Injection and Extraction module.

Radar Backbone	NDS↑	mAP↑
PointPillar	54.3	42.6
+ RCS-aware BEV encoder	55.7 ↑1.4	44.5 ↑1.9
+ Transformer Backbone	55.8 ↑0.1	44.8 ↑0.3
+ Injection and Extraction module	**56.4** ↑0.6	**45.6** ↑0.8

state-of-the-art performance with 62.8 mIoU, outperforming previous best results by 1.8 mIoU. These results demonstrate the effectiveness of our method in dealing with the BEV semantic segmentation task.

5.2.3 3D Multi-Object Tracking

In Table 4, we summarize the 3D multi-object tracking results on nuScenes `test` set. Due to the high accuracy of our method in estimating objects' locations and velocities, RCBEVDet++ achieves the best AMOTA and AMOTP simultaneously compared with state-of-the-art methods.

5.3 Ablation Studies

We ablate various design choices for our proposed method. For simplicity, we conduct ablation on the 3D detection task. All results in the ablation are on the nuScenes `val` set with ResNet-50 backbone, 256×704 image input size, and 128×128 BEV size if not specified.

5.3.1 Main Components

In this study, we conduct experiments to evaluate the effectiveness of the main components in Section 3, including RadarBEVNet and CAMF. Specifically, as shown in Table 5, we successively add components to the baseline BEVDepth to compose RCBEVDet. Firstly, based on the camera-only model, we build a radar-camera 3D object detection baseline with BEVFusion [4] by adopting PointPillar as the radar backbone following CRN [13]. The baseline radar-camera detector achieves 53.6 NDS and 42.3 mAP, improving the camera-only detector by 1.7 NDS and 1.8 mAP. Next, substituting PointPillar with the proposed RadarBEVNet yields 2.1 NDS and 3.0 mAP improvement, demonstrating Radar-BEVNet's strong radar feature representation capability. Furthermore, integrating CAMF boosts 3D detection performance from 55.7 NDS to 56.4 NDS. Additionally, we follow Hop [66] to

incorporate extra multi-frame losses, termed Temporal Supervision, resulting in a 0.4 NDS improvement alongside a 0.3 mAP reduction.

Overall, we observe that each component consistently improves 3D object detection performance. Meanwhile, the results demonstrate that multi-module fusion can significantly boost detection performance.

5.3.2 RadarBEVNet

Experimental results related to the design of RadarBEVNet, including the Dual-stream radar backbone and RCS-aware BEV encoder, are presented in Table 6. Specifically, the baseline model, which uses PointPillar as the radar backbone, achieves 54.3 NDS and 42.6 mAP. Integrating the RCS-aware BEV encoder enhances 3D object detection performance by 1.4 NDS and 1.9 mAP, demonstrating the effectiveness of the proposed RCS-aware BEV feature reconstruction. Additionally, we find that the direct integration of the Transformer-based Backbone leads to marginal performance improvement. This is attributed to the individual processing of radar points by the Point-based and Transformer-based backbones, which lack the effective interaction of their distinct radar feature representations. To alleviate this issue, we introduce the Injection and Extraction module, resulting in a performance gain of 0.6 NDS and 0.8 mAP.

Furthermore, we compare the proposed RadarBEVNet with PointPillar on different input modalities. As shown in Table 7,

TABLE 7
Comparison between PointPillar and RadarBEVNet on different modal inputs. We evaluate the effectiveness of RadarBEVNet with two modal inputs.

Radar Backbone	Input	NDS↑	mAP↑
PointPillar	R	20.3	7.3
RadarBEVNet		**22.0** ↑1.7	**10.8** ↑3.5
PointPillar	C+R	54.3	42.6
RadarBEVNet		**56.4** ↑2.1	**45.6** ↑3.0

TABLE 8
Ablation of CAMF.

Fusion Method	NDS↑	mAP↑
BEVFusion [4]	55.7	45.3
+ Deformable Cross-Attention	56.1 ↑0.4	45.5 ↑0.2
+ Channel and Spatial Fusion	**56.4** ↑0.3	**45.6** ↑0.1

RadarBEVNet shows superior detection performance compared to PointPillar, with improvements of 1.7 NDS and 2.1 NDS for radar and radar-camera inputs, respectively.

5.3.3 Cross-attention Multi-layer Fusion (CAMF)

In this study, we conducted ablation experiments on the CAMF module, which includes the deformable cross-attention mechanism for aligning multi-modal features and the channel and spatial fusion module, as shown in Table 8. Specifically, the baseline model with the fusion module from BEVfusion [4] achieves 55.7 NDS and 45.3 mAP. When the Deformable Cross Attention is incorporated for multi-modal BEV feature alignment, the 3D detection performance improves from 55.7 NDS and 45.3 mAP to 56.1 NDS and 45.5 mAP. This highlights the effectiveness of the cross-attention mechanism in aligning cross-modal features. Additionally, we notice that introducing the channel and spatial fusion module for BEV feature fusion obtains a performance increase of 0.3 NDS and 0.1 mAP compared to the one-layer fusion in BEVFusion [4]. This indicates that channel and spatial multi-layer fusion provides better multi-modal BEV features.

5.3.4 Sparse Fusion with CAMF

Table 9 presents the ablation for sparse fusion with CAMF. The first row in Table 9 denotes the SparseBEV baseline. When only adopting deformable attention to align radar BEV features with image sparse features obtains performance gains of 1.2 NDS and 2.3 mAP. After adding the radar query sample for multi-modal feature alignment further boosts detection performance by 2.4 NDS and 4.2 mAP. Besides, we observe that replacing learnable positional encoding with non-parametric encoding (*i.e.*, sine positional encoding) improves results by 1.9 NDS and 1.9 mAP. Lastly, unlike CAMF in RCBEVDet, linear fusion outperforms multi-layer fusion (MLP in Table 9). The reason is that the BEV features are 2D dense features, which require spatial and channel fusion. In contrast, sparse query features are 1D features; thus, a linear fusion layer is adequate for practice.

5.4 Task Trade-off in BEV semantic segmentation

BEV semantic segmentation in nuScenes needs to complete three tasks, including vehicle, drivable area, and lane segmentation. To achieve an optimal balance among these tasks, we adjust the loss weights of the three tasks and present the results in Table 10. We
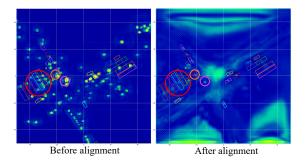

Fig. 8. **Visualization of radar misalignment.** The boxes are the ground-truth boxes. CAMF can correct the radar misalignment.

observe that applying equal loss weights for each task obtains 57.7 mIoU. By progressively increasing the loss weights of vehicle and lane classes while decreasing the loss weights for the drivable area, segmentation performance first improves from 57.7 mIoU to 59.5 mIoU, reaching its peak, and then declines to 58.9 mIoU. The best task trade-off is achieved with loss weights of 400, 80, and 200 for vehicle, drivable area, and lane, respectively; further increasing loss weights for vehicle and lane classes can hurt the segmentation performance of all three tasks.

5.5 Analysis of Robustness

5.5.1 Sensor Failure

To analyze the robustness in sensor failure scenarios, we randomly drop either images or radar inputs for evaluation. In this experiment, we adopt the dropout training strategy as data augmentation to train RCBEVDet and report *Car* class mAP following CRN [13]. Specifically, RCBEVDet consistently outperforms CRN and BEVFusion, showing higher mAP for the *Car* class in all sensor failure cases. Notably, the performance of CRN decreases by 4.5, 11.8, and 25.0 mAP in three radar sensor failure cases, while RCBEVDet only experiences drops of 0.9, 6.4, and 10.4 mAP.

These results highlight that the proposed cross-attention module enhances the robustness of BEV features through dynamic alignment.

5.5.2 Modal Alignment

To further demonstrate the effects of radar-camera alignment with CAMF, we randomly perturb the x- and y-axis coordinates of radar inputs. Concretely, we uniformly sample noise from $[-1, 1]$ for the x-axis and y-axis coordinates of each radar point, respectively. As shown in Table 12, RCBEVDet only drops 1.3 NDS and 1.5 mAP with noise radar inputs, while CRN decreases by 2.3 NDS and 5.1 mAP. Besides, we visualize how CAMF addresses radar misalignment in Figure 8. As illustrated, many radar features exhibit positional offsets from ground-truth boxes. With CAMF, these radar features are realigned within the ground-truth boxes, effectively correcting the radar misalignment.

5.5.3 Comparison with CRN

CRN [13] also utilizes deformable cross-attention to address the radar-camera mismatch issue. The results in Tables 11 and 12 demonstrate that our CAMF is more robust than the Multi-modal Deformable cross-attention module (MDCA) proposed in CRN. To further distinguish our method from CRN, we present the

TABLE 9
Ablation of sparse fusion with CAMF.

DeformAttn-only	Radar Query Sample	Positional Encoding	Fusion Layer	NDS↑	mAP↑
×	×	-	-	54.5	43.2
✓	×	Learnable	MLP	55.7↑1.2	45.5↑2.3
✓	✓	Learnable	MLP	58.1↑2.4	49.7↑4.2
✓	✓	Sine	MLP	60.0↑1.9	51.6↑1.9
✓	✓	Sine	Linear	60.4↑0.4	51.9↑0.3

TABLE 10
Task trade-off for BEV semantic segmentation. The best task trade-off is obtained when the loss weights for the three tasks are 400, 80, and 200.

Loss Weights			IoU			mIoU↑
Vehicle	Driv. Area	Lane	Vehicle↑	Driv. Area↑	Lane↑	
100	100	100	48.7	79.2	45.3	57.7
200	50	150	51.2	78.8	**46.2**	58.7
400	80	200	**52.8**	**79.8**	46.0	**59.5**
500	75	175	**52.8**	78.7	44.6	58.7
600	70	300	52.7	78.1	45.9	58.9

TABLE 11
Analysis of robustness. We report *Car* class mAP in the same manner as CRN [13].

	Input	Drop	# of view drops			
			0	1	3	All
BEVDepth [29]	C	C	49.4	41.1	24.2	0
CenterPoint [57]	R	R	30.6	25.3	14.9	0
BEVFusion [4]	C+R	C	63.9	58.5	45.7	14.3
		R		59.9	50.9	34.4
CRN [13]	C+R	C	68.8	62.4	48.9	12.8
		R		64.3	57.0	43.8
RCBEVDet (Ours)	C+R	C	**72.5**	**66.9**	**53.5**	**16.5**
		R		**71.6**	**66.1**	**62.1**

TABLE 12
Radar-camera multi-modal alignment with noise radar inputs.

Radar Input	RCBEVDet		CRN	
	NDS↑	mAP↑	NDS↑	mAP↑
Original	56.8	45.3	56.0	49.0
Noise	55.5 ↓1.3	43.7 ↓1.5	53.7 ↓2.3	43.9 ↓5.1

Fig. 9. **Comparison between alignment modules in RCBEVDet and CRN.** RCBEVDet employs an alignment-then-fusion pipeline and obtains better robustness.

conduct experiments with various backbone and detector designs in the 3D object detection framework.

5.6.1 Generalization for Backbone Architectures

To demonstrate the RCBEVDet's model generalization for backbone architecture, we conduct experiments on BEVDepth with various backbone architectures, including CNN-based and Transformer-based backbones. As shown in Table 13, our method can improve baseline performance by over 3.8~4.9 NDS and 4.8~10.2 mAP across different backbones. Furthermore, for the same type of backbone architectures with varying sizes (*e.g.*, ResNet-18 and ResNet-50), RCBEVDet can achieve consistent performance gains of 4.9 NDS.

5.6.2 Generalization for 3D Detector Architecture

We evaluate the detector architecture generalization of our method by plugging it into various mainstream multi-view camera-based 3D object detectors, including LSS-based (*e.g.*, BEVDet and BEVDepth) and transformer-based methods (*e.g.*, StreamPETR and SparseBEV). These methods represent a range of detector designs. As shown in Table 14, our method boosts the performance of all popular multi-view camera-based 3D object detectors by fusing radar features. Specifically, for the LSS-based method, RCBEVDet improves 5.6 NDS and 4.9 NDS for BEVDet and

architecture details of the fusion module in Figure 9. Specifically, CAMF in RCBEVDet first aligns features from the camera and radar separately and then fuses aligned features using several convolutional blocks. In contrast, MDCA in CRN first fuses features from two modalities. Then, it uses the fused features, image features, and radar features as the query, key, and value, respectively, and applies deformable attention to adjust the fused features. Thus, the fusion-then-alignment pipeline of MDCA in CRN has limited alignment precision because it still has feature mismatches in the camera and radar fusion process. In comparison, our CAMF, which employs an alignment-then-fusion pipeline, fully leverages the alignment capabilities of deformable attention in the first step and achieves better feature fusion robustness.

5.6 Model Generalization

RCBEVDet employs a dual-branch architecture to fuse radar and multi-view cameras and can be incorporated with any existing multi-view camera-based methods to enhance feature representation. To demonstrate the model generalization of RCBEVDet, we
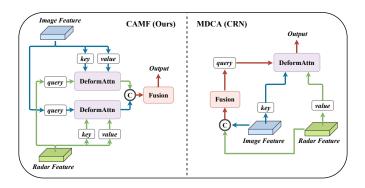
TABLE 13
Model generalization capability of RCBEVDet over various special backbones.

Method	Input	Backbone	Image Size	NDS↑	mAP↑
BEVDepth + Temporal	C	DLA34	448×800	52.5	40.2
RCBEVDet (Ours)	C+R	DLA34	448×800	56.3↑3.8	45.3↑5.1
BEVDepth + Temporal	C	Swin-T	256×704	51.8	39.4
RCBEVDet (Ours)	C+R	Swin-T	256×704	56.2↑4.4	49.6↑10.2
BEVDepth + Temporal	C	ResNet-18	256×704	49.9	35.0
RCBEVDet (Ours)	C+R	ResNet-18	256×704	54.8↑4.9	42.9↑7.9
BEVDepth + Temporal	C	ResNet-50	256×704	51.9	40.5
RCBEVDet (Ours)	C+R	ResNet-50	256×704	56.8↑4.9	45.3↑4.8

TABLE 14
Model generalization capability over various multi-view camera-based 3D object detectors.

Method	Input	NDS↑	mAP↑
BEVDet + Temporal	C	48.7	35.4
RCBEVDet (Ours)	C+R	54.3↑5.6	41.3↑5.9
BEVDepth + Temporal	C	51.9	40.5
RCBEVDet (Ours)	C+R	56.8↑4.9	45.3↑4.8
StreamPETR	C	54.0	43.2
RCBEVDet++ (Ours)	C+R	59.6↑5.6	51.5↑8.3
SparseBEV	C	54.5	43.2
RCBEVDet++ (Ours)	C+R	60.4↑5.9	51.9↑8.7

BEVDepth, respectively. For the transformer-based detectors, RCBEVDet++ obtains similar performance gains in NDS, *i.e.*, 5.6 NDS and 5.9 NDS improvement for StreamPETR and SparseBEV, respectively. Notably, we observe more mAP improvements in transformer-based methods compared to LSS-based methods. The reason is that LSS-based methods typically use depth supervision from LiDAR points for more accurate 3D position prediction, while transformer-based methods learn 3D position implicitly. Consequently, the transformer-based methods can derive more benefits from radar features, which provide additional depth information. Overall, these results demonstrate the detector architecture generalization of our method across various 3D object detectors.

6 CONCLUSION

In this paper, we first introduce RCBEVDet, a radar-camera fusion 3D detector. It comprises an existing camera-based 3D detection model, a specially designed radar feature extractor, and the CAMF module for aligning and fusing radar-camera multi-modal features. RCBEVDet improves the performance of various camera-based 3D object detectors across several backbones and shows well robustness capability against sensor failure cases on the nuScenes dataset.

To unleash the full potential of RCBEVDet, we propose RCBEVDet++, which extends the CAMF module to support query-based multi-view camera perception models with sparse fusion and adapts to more perception tasks, including 3D object detection, BEV semantic segmentation, and 3D multi-object tracking. Extensive experiments on the nuScenes dataset show that RCBEVDet++ further boosts the performance of camera-based perception models and achieves new state-of-the-art radar-camera multi-modal results on these three perception tasks. Notably, without using test-time augmentation or model ensemble, RCBEVDet++ obtains 72.73 NDS and 67.34 mAP for 3D object detection with ViT-L as the image backbone.

REFERENCES

[1] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. (document)

[2] X. Ma, W. Ouyang, A. Simonelli, and E. Ricci, "3d object detection from images for autonomous driving: a survey," *IEEE Transactions on Pattern Recognition and Machine Intelligence (PAMI)*, 2023. (document)

[3] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *Journal of field robotics*, 2020. (document)

[4] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. L. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *International Conference on Robotics and Automation (ICRA)*, 2023. (document), 5.3.1, 8, 5.3.3, 11

[5] G. L. Charvat, *Small and short-range radar systems*. CRC Press, 2014. (document)

[6] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue, "Towards deep radar perception for autonomous driving: Datasets, methods, and challenges," *Sensors*, 2022. (document)

[7] P. Li, P. Wang, K. Berntorp, and H. Liu, "Exploiting temporal relations on radar perception for autonomous driving," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. (document)

[8] M. Bijelic, T. Gruber, F. Mannan, F. Kraus, W. Ritter, K. Dietmayer, and F. Heide, "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. (document)

[9] L. Zheng, S. Li, B. Tan, L. Yang, S. Chen, L. Huang, J. Bai, X. Zhu, and Z. Ma, "Rcfusion: Fusing 4d radar and camera with bird's-eye view features for 3d object detection," *IEEE Transactions on Instrumentation and Measurement*, 2023. (document), 2.2

[10] W. Xiong, J. Liu, T. Huang, Q.-L. Han, Y. Xia, and B. Zhu, "Lxl: Lidar exclusive lean 3d object detection with 4d imaging radar and camera fusion," *arXiv preprint arXiv:2307.00724*, 2023. (document)

[11] T. Liang, H. Xie, K. Yu, Z. Xia, Z. Lin, Y. Wang, T. Tang, B. Wang, and Z. Tang, "Bevfusion: A simple and robust lidar-camera fusion framework," in *Neural Information Processing Systems (NeurIPS)*, 2022. (document)

[12] Y. Kim, S. Kim, J. W. Choi, and D. Kum, "Craft: Camera-radar 3d object detection with spatio-contextual fusion transformer," in *Association for the Advancement of Artificial Intelligence (AAAI)*, 2023. (document), 2.2, 1, 2

[13] Y. Kim, J. Shin, S. Kim, I.-J. Lee, J. W. Choi, and D. Kum, "Crn: Camera radar net for accurate, robust, efficient 3d perception," in *IEEE International Conference on Computer Vision (ICCV)*, 2023. (document), 2.2, 5.1.2, 1, 2, 3, 4, 5.3.1, 5.5.1, 5.5.3, 11

[14] R. Nabati and H. Qi, "Centerfusion: Center-based radar and camera fusion for 3d object detection," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021. (document), 2.2, 1, 2

[15] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, and F. Zhao, "Graph-detr3d: rethinking overlapping regions for multi-view 3d object detection," in *ACM International Conference on Multimedia (ACM MM)*, 2022. 2.1

[16] J. Huang, G. Huang, Z. Zhu, Y. Ye, and D. Du, "Bevdet: High-performance multi-camera 3d object detection in bird-eye-view," *arXiv preprint arXiv:2112.11790*, 2021. 2.1, 2.1.1, 1

[17] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *European Conference on Computer Vision (ECCV)*, 2022. 2.1, 2.1.2, 2, 3

[18] S. Shi, Z. Wang, J. Shi, X. Wang, and H. Li, "From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network," *IEEE Transactions on Pattern Recognition and Machine Intelligence (PAMI)*, 2020. 2.1

[19] X. Weng and K. Kitani, "Monocular 3d object detection with pseudo-lidar point cloud," in *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2019. 2.1

[20] Y. Chen, L. Tai, K. Sun, and M. Li, "Monopair: Monocular 3d object detection using pairwise spatial relationships," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2.1

[21] X. Xu, Z. Chen, and F. Yin, "Monocular depth estimation with multi-scale feature fusion," *IEEE Signal Processing Letters*, 2021. 2.1

[22] Y. Lu, X. Ma, L. Yang, T. Zhang, Y. Liu, Q. Chu, J. Yan, and W. Ouyang, "Geometry uncertainty projection network for monocular 3d object detection," in *IEEE International Conference on Computer Vision (ICCV)*, 2021. 2.1

[23] A. Simonelli, S. R. Bulo, L. Porzi, E. Ricci, and P. Kontschieder, "Towards generalization across depth for monocular 3d object detection," in *European Conference on Computer Vision (ECCV)*, 2020. 2.1

[24] Z. Miao, J. Chen, H. Pan, R. Zhang, K. Liu, P. Hao, J. Zhu, Y. Wang, and X. Zhan, "Pvgnet: A bottom-up one-stage 3d object detector with integrated multi-level features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2.1

[25] F. Liu and X. Liu, "Voxel-based 3d detection and reconstruction of multiple objects from a single image," 2021. 2.1

[26] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2.1, 5.1.1

[27] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2.1

[28] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *European Conference on Computer Vision (ECCV)*, 2020. 2.1, 2.1.1, 5.1.1

[29] 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 *Association for the Advancement of Artificial Intelligence (AAAI)*, 2023. 2.1, 2.1.1, 5.1.2, 1, 2, 5, 11

[30] J. Huang and G. Huang, "Bevdet4d: Exploit temporal cues in multi-camera 3d object detection," *arXiv preprint arXiv:2203.17054*, 2022. 2.1, 2.1.1, 5.1.2, 5

[31] Y. Wang, V. Guizilini, T. Zhang, Y. Wang, H. Zhao, , and J. M. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *Conference on Robot Learning (CoRL)*, 2021. 2.1, 2.1.2

[32] Y. Liu, T. Wang, X. Zhang, and J. Sun, "Petr: Position embedding transformation for multi-view 3d object detection," in *European Conference on Computer Vision (ECCV)*, 2022. 2.1, 2.1.2

[33] H. Liu, Y. Teng, T. Lu, H. Wang, and L. Wang, "Sparsebev: High-performance sparse 3d object detection from multi-camera videos," in *IEEE International Conference on Computer Vision (ICCV)*, 2023. 2.1, 2.1.2, 4.2, 5.1.2, 1, 2

[34] S. Wang, Y. Liu, T. Wang, Y. Li, and X. Zhang, "Exploring object-centric temporal modeling for efficient multi-view 3d object detection," in *IEEE International Conference on Computer Vision (ICCV)*, 2023. 2.1, 2.1.2, 1, 2, 4

[35] B. Zhou and P. Krähenbühl, "Cross-view transformers for real-time map-view semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2.1, 2.1.2, 2.2, 4.2, 3

[36] Y. Man, L.-Y. Gui, and Y.-X. Wang, "Bev-guided multi-modality fusion for driving perception," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2.1, 2.2, 3

[37] J. Schramm, N. Vödisch, K. Petek, B. R. Kiran, S. Yogamani, W. Burgard, and A. Valada, "Bevcar: Camera-radar fusion for bev map and object segmentation," *arXiv preprint arXiv:2403.11761*, 2024. 2.1, 2.2, 3

[38] A. Hu, Z. Murez, N. Mohan, S. Dudas, J. Hawke, V. Badrinarayanan, R. Cipolla, and A. Kendall, "FIERY: Future instance segmentation in bird's-eye view from surround monocular cameras," in *IEEE International Conference on Computer Vision (ICCV)*, 2021. 2.1.1

[39] J. Park, C. Xu, S. Yang, K. Keutzer, K. M. Kitani, M. Tomizuka, and W. Zhan, "Time will tell: New outlooks and A baseline for temporal multi-view 3d object detection," in *International Conference on Learning Representations (ICLR)*, 2023. 2.1.1, 1, 2

[40] 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 *IEEE International Conference on Computer Vision (ICCV)*, 2023. 2.1.2, 2

[41] X. Lin, T. Lin, Z. Pei, L. Huang, and Z. Su, "Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion," *arXiv preprint arXiv:2211.10581*, 2022. 2.1.2, 4

[42] ——, "Sparse4d v2: Recurrent temporal fusion with sparse model," *arXiv preprint arXiv:2305.14018*, 2023. 2.1.2

[43] B. Yang, R. Guo, M. Liang, S. Casas, and R. Urtasun, "Radarnet: Exploiting radar for robust perception of dynamic objects," in *European Conference on Computer Vision (ECCV)*, 2020. 2.2

[44] Z. Wu, G. Chen, Y. Gan, L. Wang, and J. Pu, "Mvfusion: Multi-view 3d object detection with semantic-aligned radar and camera fusion," in *International Conference on Robotics and Automation (ICRA)*, 2023. 2.2, 2

[45] A. W. Harley, Z. Fang, J. Li, R. Ambrus, and K. Fragkiadaki, "Simple-bev: What really matters for multi-sensor bev perception?" in *International Conference on Robotics and Automation (ICRA)*, 2023. 2.2, 3

[46] Y. Long, A. Kumar, D. D. Morris, X. Liu, M. Castro, and P. Chakravarty, "RADIANT: radar-image association network for 3d object detection," in *Association for the Advancement of Artificial Intelligence (AAAI)*, 2023. 2.2

[47] G. Shi, R. Li, and C. Ma, "Pillarnet: Real-time and high-performance pillar-based 3d object detection," in *European Conference on Computer Vision (ECCV)*, 2022. 2.2

[48] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3.1

[49] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3.1.1

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Neural Information Processing Systems (NeurIPS)*, 2017. 3.1.1, 3.1.1

[51] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, 2020. 3.1.1

[52] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, 2018. 3.1.2

[53] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: deformable transformers for end-to-end object detection," in *International Conference on Learning Representations (ICLR)*, 2021. 3.2.1, 3.2.1

[54] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *IEEE International Conference on Computer Vision (ICCV)*, 2017. 4.2

[55] J. Huang and G. Huang, "Bevpoolv2: A cutting-edge implementation of bevdet toward deployment," *arXiv preprint arXiv:2211.17111*, 2022. 5.1.2

[56] Y. Kim, J. W. Choi, and D. Kum, "Grif net: Gated region of interest fusion network for robust 3d object detection from radar point cloud and monocular image," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2020. 5.1.2

[57] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3d object detection and tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5.1.2, 2, 4, 11

[58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015. 5.1.2

[59] T. Zhou, J. Chen, Y. Shi, K. Jiang, M. Yang, and D. Yang, "Bridging the view disparity between radar and camera features for multi-modal fusion 3d object detection," *IEEE Transactions on Intelligent Vehicles*, 2023. 1, 2

[60] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[61] M. Ulrich, S. Braun, D. Köhler, D. Niederlöhner, F. Faion, C. Gläser, and H. Blume, "Improved orientation estimation and detection with hybrid object detection networks for automotive radar," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2022. 2

[62] P. Li, X. Chen, and S. Shen, "Stereo r-cnn based 3d object detection for autonomous driving," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[63] X. Jiang, S. Li, Y. Liu, S. Wang, F. Jia, T. Wang, L. Han, and X. Zhang, "Far3d: Expanding the horizon for surround-view 3d object detection," in *Association for the Advancement of Artificial Intelligence (AAAI)*, 2024. 2

[64] Y. Li, Y. Chen, X. Qi, Z. Li, J. Sun, and J. Jia, "Unifying voxel-based representation with transformer for 3d object detection," 2022. 4

[65] D. Stadler and J. Beyerer, "Bytev2: Associating more detection boxes under occlusion for improved multi-person tracking," in *International Conference on Pattern Recognition (ICPR)*, 2022. 4

[66] Z. Zong, D. Jiang, G. Song, Z. Xue, J. Su, H. Li, and Y. Liu, "Temporal enhanced training of multi-view 3d object detector via historical object prediction," in *IEEE International Conference on Computer Vision (ICCV)*, 2023. 5.3.1