RAFDet: Range View Augmented Fusion Network for Point-Based 3D Object Detection

Zhijie Zheng, Zhicong Huang, Jingwen Zhao, Kang Lin, Haifeng Hu, *Member, IEEE*, and Dihu Chen

***Abstract*—In recent years, point-based methods have achieved promising performance on 3D object detection task. Although effective, they still suffer from the inherent sparsity of point cloud, which makes it challenging to distinguish objects with backgrounds only relying on the view of raw point. To this end, we propose a straightforward yet effective multi-view fusion network termed RAFDet to alleviate this issue. The core idea of our method lies in combining the merits of raw point and its range view to enhance the representation learning for sparse point cloud, thus mitigating the sparsity problem and boosting the detection performance. In particular, we introduce a novel bidirectional attentive fusion module to equip sparse point with interacted fine-grained semantic clues during feature learning process. Then, we devise the range-view augmented fusion module to fully exploit the supplementary relationship between different perspectives with the aim of enhancing original point-view features. In the end, a single-stage detection head is utilized to predict final 3D bounding boxes based on the enhanced semantics. We have evaluated our method on the popular KITTI Dataset, DAIR-V2X Dataset and Waymo Open Dataset. Experimental results on the above three datasets demonstrate the effectiveness and robustness of our approach in terms of detection performance and model complexity.**

***Index Terms*—3D object detection, LiDAR, range view fusion, transformer.**

I. Introduction

With the surging demand for autonomous driving, 3D object detection task has played a key role in real-time scenario prediction and route planning. Thanks to the thriving development of sensor technology, multiple techniques are applied to perceive the surrounding environment and collect data for object estimation. Generally, LiDAR has been commonly used due to its intrinsic advantage of obtaining precise instance depth information. Nevertheless, it suffers from the problem of sparsity and non-uniformity, which hinders the further improvement of 3D detection performance.

To alleviate this phenomenon, recent studies [1], [2], [3], [4], [5], [6] have sought the supplementary traits between LiDAR and images to equip sparse points with dense texture information, which contributes to the boost of detection performance. These methods can be mainly categorized into **multi-view fusion** and **multi-modal fusion** methods. The multi-view fusion method termed MVF refers to the feature integration of diverse representations of point cloud, which can be generally classified into point view (PV), range view (RV) and bird's eye view (BEV). Point view means directly processing raw point cloud, and features obtained through the feature extractor are defined as point features. Range view refers to the 2D images scanned vertically and horizontally by LiDAR. Bird's eye view is the projected view onto x-y plane by conducting height compression to point cloud along the z axis.

For the first method, existing works [7], [8], [9] project point cloud into range view to obtain rich pixel features, which are then fused with BEV features at the deep level to improve the discrimination ability of detectors. However, the above approaches rely on the voxelization process of point cloud, thus inevitably leading to certain quantization loss. Furthermore, the utilization of BEV features, where the height information of object is compressed, negatively affects the accurate instance prediction.

The second method [10], [11], [12] aims to decorate sparse point cloud with textural features extracted from RGB images. Previous literature like [10] enhances point cloud with image features obtained from a pre-trained 2D network. VPFNet [13] employs an intermediate representation called virtual points to aggregate image features. Recent work FGFusion [12] demonstrates the importance of low-level image informatics and devises a multi-scale fusion scheme to integrate downsampled semantic features. Despite the effectiveness, multi-modal methods require accurate camera projection matrix, and the robustness of the model is largely influenced by perturbations of external parameters.

Therefore, to improve the robustness of network and the representation learning for point cloud, we propose a **R**ange View **A**ugmented **F**usion Network for Point-Based 3D Object **Det**ection named **RAFDet** to alleviate the sparsity problem. To demonstrate the effectiveness of multi-view fusion in our work, we make a comparison with a LiDAR-based single-view model IA-SSD [14] in detection results. As shown in Fig. 1(a), we can achieve favorable detection performance when only using raw point cloud based on its preserved geometry and depth information. Nonetheless, it brings about several false positives and missed positives due to the sparsity of point cloud. On the contrary, our method in Fig. 1(b) can provide supplementary context information to sparse points, which facilitates the model to

Received 27 December 2023; revised 2 July 2024 and 5 November 2024; accepted 9 November 2024. Date of publication 28 January 2025; date of current version 8 July 2025. This work was supported by the Key Area R&D Program of Guangdong Province under Grant 2022B0701180001. The associate editor coordinating the review of this article and approving it for publication was Dr Stefania Colonnese. *(Corresponding author: Dihu Chen.)*

Zhijie Zheng, Zhicong Huang, Jingwen Zhao, Kang Lin, and Haifeng Hu are with the School of Electronic and Information Technology, Sun Yat-Sen University, Guangzhou 511400, China.

Dihu Chen is with the School of Integrated Circuits, Sun Yat-Sen University, Guangzhou 518107, China (e-mail: stscdh@mail.sysu.edu.cn).

Digital Object Identifier 10.1109/TMM.2025.3535289

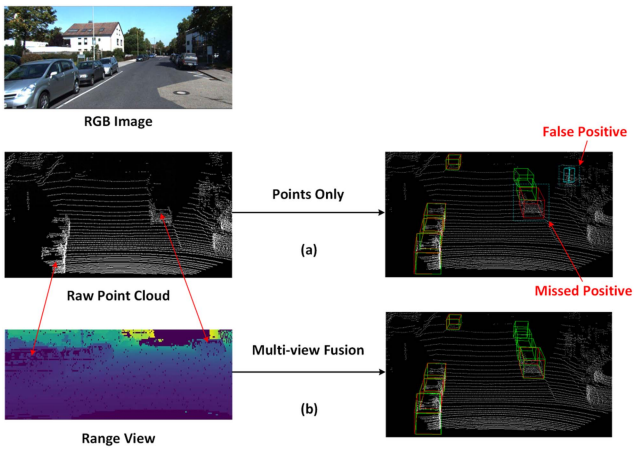

Fig. 1. An illustration of distinct detection results in (a) point only of IA-SSD and (b) multi-view fusion of our work. The red boxes refer to the ground truth. The green and blue boxes are the prediction of car and pedestrian, respectively.

detect more foreground objects and discard false positives. To be specific, the peculiar aspect of our method lies in the designed framework with two innovative branches and cross view fusion. Firstly, we transform the point cloud from cartesian coordinate to cylindrical coordinate. The corresponding feature extraction backbones are then used to obtain both high-level point features and image feature maps for two branches. After that, we project the downsampled key points into image plane and obtain lifted point-wise features through interpolation. In order to achieve a better fusion of the two perspectives, we design a novel range-view augmented fusion module, which takes the interpolated features as a supplement to enhance the original point-view features.

Simultaneously, we discover that most existing point-based detection pipelines lack adequate feature learning for point cloud, as indicated in the literature [15], [16], [17]. In this paper, we take an insightful look into this problem and discover that the simple concatenation adopted by [15] is incapable of exploiting fine-grained interaction between features from two channels. To solve this, a bidirectional attentive fusion module is devised to excavate in-depth fused features in a gate mechanism manner, so that we can equip point cloud with more delicate information. This in turn promotes the model to realize more precise object detection based on the enhanced clues.

Based on the above designs and analysis, our method has the strength of improving the representation learning ability of network for sparse point cloud through incorporating the supplementary information between point and range view. Our design can obtain competitive detection performance on several datasets and achieve the tradeoff between accuracy and model complexity, showing the effectiveness and robustness of the proposed model.

To summarize, our contributions of this paper can be listed as:

- We propose a novel one-stage point-based multi-view fusion network RAFDet, which takes advantage of different perspectives to solve the problem of insufficient representation capability of sparse point cloud.
- A bidirectional attentive fusion module is devised to exploit the in-depth interaction of valuable features from two channels, thus equipping sparse points with more distinct contextual information.
- A range-view augmented fusion module is presented with the purpose of enhancing point features through the exploration of supplementary properties cross perspectives, which improves the final detection performance.
- Extensive experimental results on KITTI Dataset [18], DAIR-V2X Dataset [19] and Waymo Open Dataset [20] demonstrate that our approach can achieve superior performance on multiple classes with low model complexity, which validates the effectiveness and robustness of our network.

II. Related Work

With the rapid development of deep learning, the 3D object detection task has witnessed tremendous progress. Generally, the existing methods can be sorted into four sections:

A. Point-based Methods for 3D Object Detection

Given the intrinsic advantages of point cloud, e.g., the reserved object geometry information, existing point-based methods directly process raw point cloud through hierarchical downsampling and feature extraction like PointNet++ [21]. Shi et al. [22] propose a two-stage detection network, which leverages RoI pooling to achieve elaborate refinement on initial bounding boxes at the second stage. 3DSSD [23] introduces a voting module [24] to predict accurate instance centers. Besides, a feature-aware sampling scheme is designed and integrated with traditional farthest point sampling to retain more foreground points, thus boosting the detection performance. Pan et al. [25] devise the local and global transformer block to take the place of PointNet++ [21], which utilizes the attention mechanism of transformer [26] to extract fine-grained contextual clues between objects. In order to improve the insufficient representation capacity of PointNet++ [21], SASA [27] presents a semantic-augmented abstraction module to equip point cloud with more semantic features. Similarly, IA-SSD [14] proposes a lightweight detection framework with centroid-aware sampling strategy to achieve the tradeoff between performance and efficiency.

Despite the effectiveness of point-based method, massive computational resources and limited inference speed are the main bottleneck that impedes its application, especially for scenes with large point cloud.

B. Voxel-based Methods for 3D Object Detection

Distinguishing from the above point-based methods, voxel-based methods generally discretize point cloud into uniform voxels or pillars. Then, stacked convolution neural networks are applied to extract high-level sparse features. The early pioneering work VoxelNet [28] represents point cloud as 3D voxels, which are encoded by voxel feature layer and passed into 3D convolution for final prediction. However, this puts a great computational burden on the detection network. Therefore, Yan et al. [29]

design sparse 3D convolution to neglect the impact of empty voxels, thus speeding up the detection pipeline by a large margin. To further save the computation overhead, PointPillars [30] uses pillar to simplify the data representation so that pseudo images can be generated and processed by 2D convolution. Deng et al. [31] introduce a novel two-stage framework with designed voxel RoI pooling to aggregate valuable features for coarse-to-fine box refinement. Simultaneously, Mao et al. [32] propose a sparse and submanifold voxel module to build a transformer-based backbone, which applies multiple attention on empty and non-empty voxels to obtain detailed features.

In general, current voxel-based method dominates in the field of 3D object detection due to its high efficiency and competitive performance. Nevertheless, well-designed training optimization strategies are needed for voxel-based methods [33] if to achieve satisfying detection performance.

C. 3D Object Detection with Multi-view Fusion

The multi-view fusion methods aim to incorporate different prominent features by combining perspectives like point view (PV), bird's eye view (BEV), and range view (RV). MVF [34] projects point cloud into BEV and range view, which are processed by convolution and concatenated to obtain high-level semantics. Similar to MVF [34], Pillar-OD [7] fuses point-wise features from BEV and range view through bilinear interpolation. Then, the intermediate features are mapped into BEV for object regression and classification. Deng et al. [8] make use of the hollow-3D property of point cloud and get the enhanced multi-view features through bilaterally guided fusion module. Besides, a hierarchical pooling strategy is adopted to obtain more fine-grained multi-scale features from different grid resolutions. Recently, ACDet [9] proposes an innovative fusion tactic on the basis of transformer to integrate deep BEV and range view features at late stage. Furthermore, inspired by [35], it presents a geometric-attention kernel to focus on discriminative spatial information between pixels.

Overall, most multi-view fusion methods put emphasis on the combination of BEV and range view. While the inherent drawbacks of the two views lie in the loss of height and depth information. Therefore, we opt for the point view, which has the advantage of structural information to achieve better combination with the range view.

D. 3D Object Detection with Multi-modal Fusion

Nowadays, more and more researchers seek to the solution of multi-modal fusion, particularly the fusion of LiDAR and camera owing to the complementary relationship. Qi et al. [36] propose to first detect objects in RGB images through 2D detectors, which are uplifted into 3D space for box estimation. EPNet [37] uses 2D convolution to encode multi-scale image features and conducts feature-level fusion to enhance point-wise features. PointPainting [38] relies on an additional segmentation network to get image segmented scores, which are added to point cloud to generate a more discriminable representation. Ma et al. [39] introduce a novel fusion scheme to strengthen the point cloud with extracted camera features. FusionPillars [40] leverages a dual-fusion backbone to augment both point and image features in a bidirectional manner as the interaction of two modalities leads to more robust performance. Yan et al. [41] emphasize the importance of low-level image features and design an attention module to achieve fine-grained multi-modal fusion. SGF3D [42] exploits the advantages of pseudo point cloud representation to improve the detection performance.

However, multi-modal fusion methods need to introduce pre-trained 2D model and accurate projection parameters, leading to much expensive consumption. Conversely, our model does not require extra pre-trained model or external parameters and achieves the end-to-end training of the whole detection pipeline.

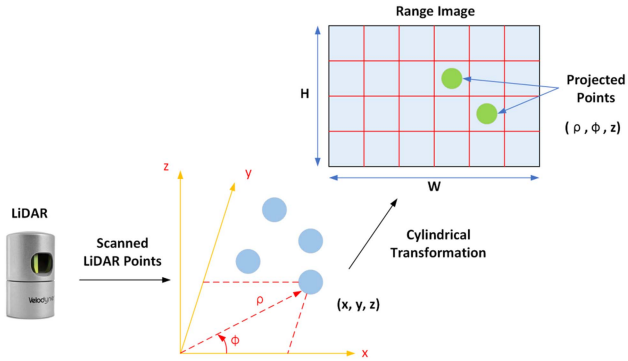

Fig. 2. The illustration of range image generation and cylindrical view projection.

III. Preliminaries

In this section, we present some background information about the generation of range image and initial features obtained from view projection.

Generally, a range image is produced from the t measurements of s-beam LiDAR in one scanning period, and the size of image is denoted as $s \times t$ [35]. For example, KITTI [18] uses 64-beam LiDAR to scan around the environment, which roughly generates up to 2000 points for each beam [43]. Therefore, we set the size of range image in KITTI to 64×2048. Moreover, in order to focus on the front view of potential foreground objects and reduce computation cost, we scale the size of range image to 48×512, following the common practice [9], [43].

After obtaining the range image, we conduct view transformation of point cloud to encode initial image features. The mainstream methods tend to utilize cylindrical or spherical view projection, while some research [7] discovers that spherical projection will lead to height distortion of objects, which is not beneficial for improving detection performance. Therefore, we opt for cylindrical projection in this paper, as shown in Fig. 2.

Specifically, given the Cartesian coordinate of point d_i, denoted as (x_i, y_i, z_i), cylindrical view projection is adopted to transform point into range image plane, which is formulated as:

$$\begin{aligned} \rho_i &= \sqrt{x_i^2 + y_i^2}, \\ \varphi_i &= \arctan \frac{y_i}{x_i}, \\ z_i &= z_i, \end{aligned} \tag{1}$$

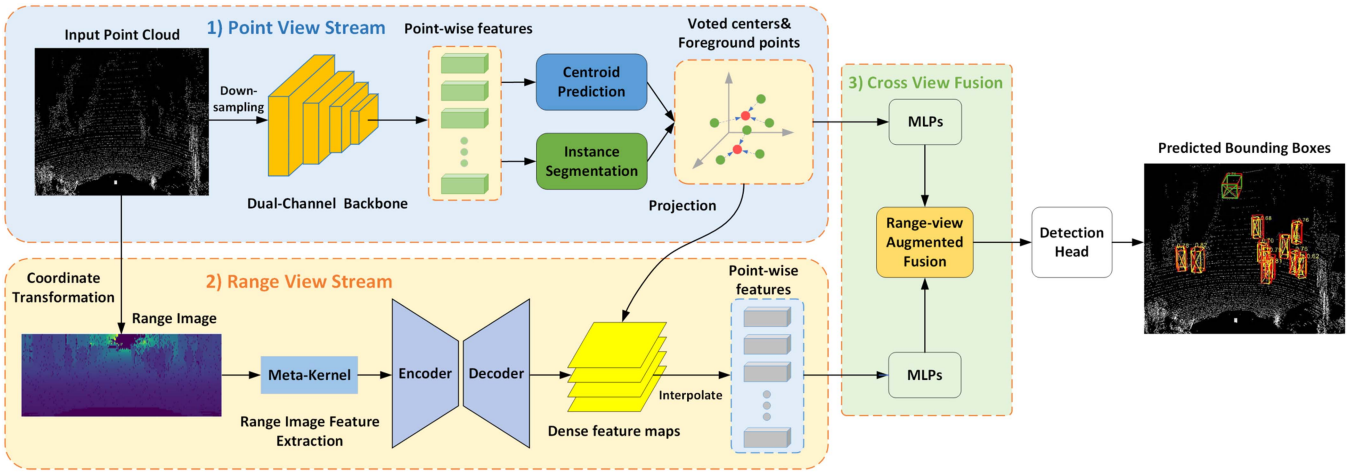

Fig. 3. The overall architecture of our proposed RAFDet. It is composed of point view stream, range view stream, cross view fusion, and detection head. 1) For point view stream, we feed raw point cloud into a dual-channel backbone, which consists of both self-attention channel and traditional convolution channel, to extract in-depth spatial features. These semantic clues are used for centroid prediction and instance segmentation through Multi-Layer Perceptrons. 2) For range view stream, the input point cloud is first projected into range image through cylindrical coordinate transformation to generate initial image features. The geometry-aware meta kernel and encoder-decoder 2D backbone are utilized to extract multi-scale dense image features. Afterwards, we cast downsampled points on range image feature maps and leverage bilinear interpolation to retrieve valuable point-wise features from 2D range image pixels. 3) For cross view fusion, the range-view augmented fusion module takes cross-perspective semantics from both point view and range view stream as input to enhance point-view features. Finally, a simple detection head is adopted for final bounding box estimation.

Then, we incorporate the original (x_i, y_i, z_i) of point d_i, the intensity r_i as well as the depth information ρ_i into image channel to form initial feature maps (48,512,5), which are subsequently sent to 2D backbone for feature extraction.

IV. Proposed Method

In this section, we introduce the architecture of our proposed RAFDet. Firstly, the overall framework is presented, then followed by the illustration of several significant elements.

A. Overall Framework

Most point-based methods suffer from the bottleneck of performance improvement due to the sparsity of points and the indistinguishable shape of foreground objects from the background. Therefore, inspired by the discovery that range image has dense representation [44], we propose RAFDet to enhance raw point features through the supplementary information of range view for sparse points. As shown in Fig. 3, our model is comprised of three main components: 1) point view stream with dual-channel backbone, 2) range view stream for range image feature extraction, 3) cross view fusion through range-view augmented fusion module.

To start with, we downsample the point cloud hierarchically to obtain a subset of points. Then, the dual-channel backbone is adopted to excavate useful features for potential objects. These distinct point-wise features are put into MLPs (Multi-Layer Perceptron) for instance segmentation [14] to maintain more foreground points. Besides, on the basis of obtained semantic clues, we further predict the centroid of latent objects by several MLPs to get more powerful representation. Meanwhile, we conduct cylindrical coordinate transformation on the input point cloud to get coarse range image feature maps, followed by the geometry-aware meta kernel and encoder-decoder 2D backbone [9] to obtain multi-scale image features. Subsequently, given the image feature maps as well as discriminative foreground points and object centroid, we perform interpolation to get point-wise features. A range-view augmented fusion module is devised to integrate complementary information of cross perspectives, thus enhancing original point-view features. We utilize a simple anchor-free 3D detection head to get final prediction results.

In the following section, we will give a detailed description of three parts, which are point view stream, range view stream, and cross view fusion. Lastly, the training losses used in our framework are presented.

B. Point View Stream

Previous study [15] has pointed out that merely relying on a single-channel feature coding network will lead to insufficient feature learning of sparse points. As a consequence, a dual-channel backbone which consists of PointNet++ [21] as well as transformer [26] is proposed to mitigate this problem. Despite its effectiveness, [15] only utilizes concatenation operation to fuse features from two channels, which fails to achieve effective in-depth feature integration and results in suboptimal performance of object detector. To this end, we design a novel bidirectional attentive fusion module termed **BAFM** to take the place of concatenation in order to better combine dual channel features. The illustration of dual-channel backbone is shown in Fig. 4.

To begin with, we adopt the commonly used downsampling strategy like furthest point sampling (FPS) to obtain a subset of key points for massive original point cloud. Next, the group operation is utilized to fetch the grouped coordinates and

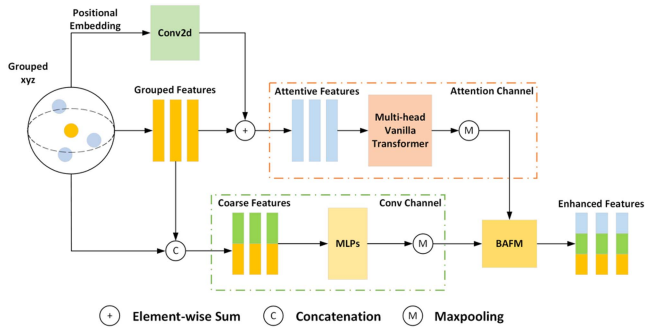

Fig. 4. The illustration of dual-channel backbone. We get both attentive and coarse features through group operation in different manners. Then these features are put into multi-head vanilla transformer and MLPs respectively to obtain long-range contextual as well as fine-grained information of objects. We fuse the two types of features through bidirectional attentive fusion module to exploit more in-depth representation.

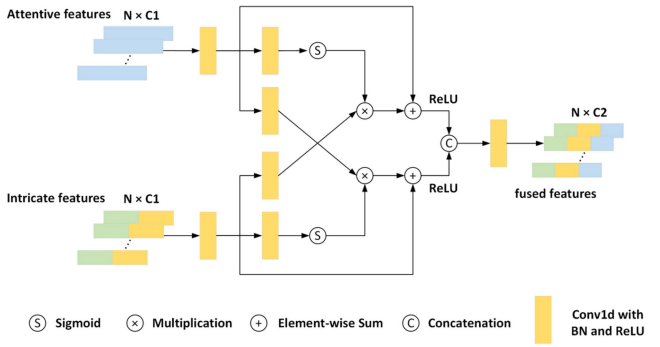

Fig. 5. The illustration of bidirectional attentive fusion module. The selective masks are generated via sigmoid function and multiplied with disparate features. Then we add them by residual skip and perform concatenation to get distinguishing fused features.

features of surrounding points based on specific key points, which are concatenated together to form the coarse features $\{f_n^c | n = 1, .., M\}$, where M is the number of points. Furthermore, we use multiple convolution layers to embed the positional information of grouped points and add it with grouped features to generate the attentive features $\{f_n^a | n = 1, .., M\}$. Then, the above two types of features are fed into corresponding stacked MLPs as well as multi-head vanilla transformer to excavate the fine-grained contextual information, which is formulated as follows:

$$\begin{aligned} \tilde{f}_n^a &= \mathrm{Max}\left(TransBlock\left(f_n^a\right)\right), \\ \tilde{f}_n^c &= \mathrm{Max}\left(MLP\left(f_n^c\right)\right), n = 1, \ldots, M, \end{aligned} \tag{2}$$

where $Max(\cdot)$ denotes the max-pooling operation.

Given the extracted features $\tilde{f}_n^a$ from self-attention mechanism and $\tilde{f}_n^c$ from MLPs, we intend to exploit an effective fusion tactics to equip point cloud with richer representation. Typically, a naive approach is to leverage addition or concatenation for feature fusion like [15]. Nevertheless, attentive features are built on capturing instance long-range dependencies while intricate features are focused on fine-grained local information of object. Therefore, it is essential to design a detailed fusion module to minimize the disparity of the two features.

Inspired by the fusion block in [8], which relies on gate mechanism to select significant features, we propose a bidirectional attentive fusion module, shown in Fig. 5. Specifically, the input features both come through one layer of convolution with BN and ReLU, where $Sigmoid$ function is conducted to generate selective masks. Hence, we can excavate more distinguishing and in-depth features from the two channels. This fusion procedure can be formulated as follows:

$$\tilde{f}_n^{a1}, \tilde{f}_n^{a2}, \tilde{f}_n^{c1}, \tilde{f}_n^{c2} = ReLU\left(BN\left(CONV\left(\tilde{f}_n^a, \tilde{f}_n^c\right)\right)\right),$$

$$mask_n^a, mask_n^c = Sigmoid\left(\left(CONV\left(\tilde{f}_n^{a1}, \tilde{f}_n^{c1}\right)\right)\right),$$

$$\tilde{f}_n^{a3}, \tilde{f}_n^{c3} = ReLU\left(mask_n^c \times \tilde{f}_n^{a2} + \tilde{f}_n^{a1}, mask_n^a \times \tilde{f}_n^{c2} + \tilde{f}_n^{c1}\right),$$

$$\tilde{f}_n^{fuse} = \phi\left(CONCAT\left(\tilde{f}_n^{a3}, \tilde{f}_n^{c3}\right)\right), \tag{3}$$

where $\tilde{f}_n^{fuse}$ is the aggregated fusion features. Therefore, we can get in-depth semantic clues through the proposed bidirectional feature interaction mechanism.

These clues are used for instance segmentation and centroid prediction in different stages with the purpose of retaining more foreground points and obtaining powerful object representation. To be specific, we use simple MLPs to calculate the scores of points belonging to each category and choose the highest k points with $TopK$ function. Besides, similar operations are performed to compute the offsets, which are added with original point coordinates to obtain the predicted centroid.

C. Range View Stream

In order to obtain the dense range image features as a supplement to LiDAR point features, we adopt the geometry-aware kernel [9] and an encoder-decoder structure [45], as shown in range view stream in Fig. 3.

Specifically, given the initial feature map $f_i \in R^{H \times W \times 5}$ from coordinate transformation according to (1), a geometry-aware kernel is used to mine spatial range image features. The geometry-aware kernel comprises multiple stacked fully connected layers with batch normalization (BN) and ReLU to aggregate neighboring features to occupied pixels, which outputs the enhanced feature map $f_t \in R^{H \times W \times 32}$ as follows:

$$\begin{aligned} f_i^1, f_i^2, \ldots, f_i^n &= ReLU\left(BN\left(FC\left(f_i\right)\right)\right), \\ f_t &= \phi\left(CONCAT\left(f_i^1, f_i^2, \ldots, f_i^n\right)\right), \end{aligned} \tag{4}$$

where n denotes the number of fully connected (FC) layers and $\phi(\cdot)$ is the aggregation layer, which is composed of a convolution layer with BN and ReLU.

Subsequently, we further put these enhanced feature maps into an encoder-decoder backbone to extract multi-scale valuable image features. Particularly, several image blocks are used to reduce the image size and capture high-dimensional range

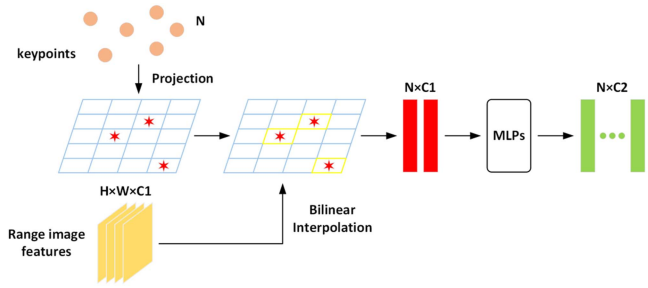

Fig. 6. The illustration of bilinear interpolation from range images to points. A series of shared MLPs are used to obtain high-dimensional vectors from interpolated point-wise features.

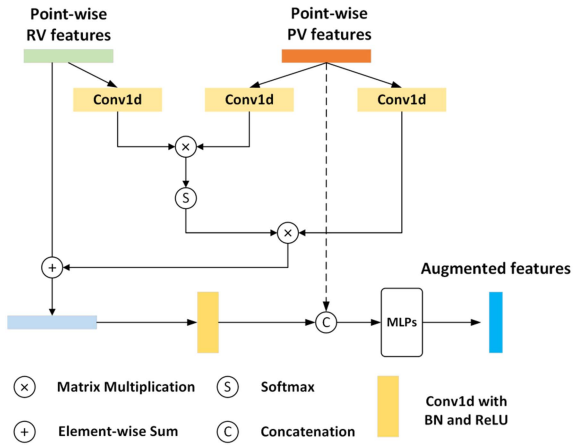

Fig. 7. The illustration of range-view augmented fusion module. We first enhance the range view features by mining potential valuable information from point view features. Next, the concatenation operation is adopted to combine multi-view features, which are sent to MLPs for generating final augmented point features.

image features, each of which contains three branches of 2D convolution with different receptive fields. Then, features from these branches are concatenated and aggregated by another 2D convolution. Therefore, we can get more meaningful and in-depth feature maps for possible objects.

After the encoder stage, decoding operation is performed to propagate the learned high-dimensional features back to the original images. Specifically, the pixelshuffle layer [45] is first used to scale the shape of feature maps and lift feature dimension. Next, similar to the former image blocks, upsampling blocks are leveraged to generate the upsampled image features. The illustration of the aforementioned procedure is as follows:

$$f_i^{down} = ResBlock[i]\left(f_t\right), \\ f_i^{rv} = UpBlock[i]\left(f_i^{down}\right), i = 1, \ldots, k, \tag{5}$$

where k indicates how many blocks we use in the network and $ResBlock$ or $UpBlock$ corresponds to the above structure we have presented.

Finally, we can get distinct range image feature maps $\{f_i^{rv}\}$ of different channels and sizes, which later could be used to supplement and enhance the point-view features through devised fusion module.

After the previous steps, we now get both LiDAR point features $\tilde{f}_n^{fuse}$ and range image features f_i^{rv}. To better conduct multi-view fusion, we need to interpolate range image features to point-wise features based on downsampled key points, which is elaborated in Fig. 6.

Specifically, given the coordinates of points in cartesian coordinate system, we conduct projection to get the corresponding indices and masks in 2D range image plane. The masks are multiplied with projected indices to make sure that indices out of image plane are discarded. Afterwards, common bilinear interpolation is performed to lift image features to spatial point-wise features according to valid indices and features f_i^{rv}. Then, the interpolated point-wise features are put into a series of MLPs to produce features matching the original point feature dimension, which serves as supplementary to augment point features through the following fusion module.

Note that, the range image features of different scales focus on semantic information of various objects. Therefore, to better exploit potential supplements for objects in different categories, we interpolate multi-scale upsampled image features with certain strides instead of only the final output, which can be formulated as follows:

$$f_n^t = B\left(T, f_i^{rv}, S\right), S = 1, 2, \\ \tilde{f}_n^t = MLP\left(f_n^t\right), \tag{6}$$

where B denotes the bilinear interpolation, T is the projected indices and S is the downsampled stride for range images.

D. Cross View Fusion

Up to now, we have obtained interpolated point-wise features from range images while how to effectively carry out feature fusion with original point-view features still remains a challenge. Common practices like addition or concatenation are not capable of providing appropriate supplementary information because the discrepancy of views is neglected [8]. Besides, several previous works [8], [9] directly fuse multi-view features based on high-dimensional semantics at the final output of the network. Although effective, as stated in [12], they overlook the fact that fine-grained feature interaction is lacked during the downsampling process and low-dimensional fusion contributes to the accuracy improvement of the model. Therefore, in consideration of the aforementioned issues, we propose an innovative range-view augmented fusion module **RAFM** and adopt an early-stage fusion scheme to augment point-wise PV features with interpolated RV features.

Specifically, the structure of the proposed RAFM is shown in Fig. 7, where point-wise RV features $\tilde{f}_n^t$ and PV features $\tilde{f}_n^{fuse}$ are two inputs to this module. Firstly, the two types of features are processed by one-dimensional convolution to generate initial references $\tilde{f}_n^T$, $\tilde{f}_n^K$, $\tilde{f}_n^B$. Then, we use matrix multiplication to explore the correlation between $\tilde{f}_n^T$ and $\tilde{f}_n^K$, followed by $Softmax$ function to obtain attentive masks. These masks are multiplied with $\tilde{f}_n^B$ to tap into valuable information, which is later added to point-wise RV features. Finally, the enhanced RV features are fed into one layer of convolution for smoothness

and concatenated with initial PV features to form the augmented fusion features.

Based on the above analysis, we can seek the supplementary information from RV features to PV features through RAFM, thus promoting the representation ability of original point-view features. It should be noted that we only utilize RAFM for fusion at the first two layers of downsampling in order to save computational overhead and avoid dimensional gap between multi-view features.

E. Training Losses

The proposed RAFDet is optimized end-to-end and the overall loss for our framework can be summarized as foreground segmentation loss L_{seg}, centroid estimation loss L_{ctr}, classification loss L_{cls} and box regression loss L_{reg}, shown as:

$$L_{total} = L_{seg} + L_{ctr} + L_{cls} + L_{reg}, \tag{7}$$

For foreground segmentation and box classification, we apply cross-entropy loss to distinguish foreground points from the background and assign positive labels to objects. Besides, the centroid estimation loss in [14] is adopted to obtain more accurate predicted centers with the supervision of ground truth. Finally, we utilize common smooth$-l_1$ loss with regard to box regression.

V. Experiment

In this section, we present the comparable experimental results on popular datasets and detailed implementations. Besides, we conduct extensive ablative studies to verify the effectiveness of each proposed component. Some qualitative visualization results are also provided to demonstrate the superior detection performance of our model.

A. Datasets and Evaluation Metrics

1) KITTI Dataset: The KITTI Dataset [18] is an outdoor detection benchmark with annotations of multiple classes, where car, pedestrian, and cyclist make up the majority of the dataset. For each category, three different difficult levels are labeled as easy, moderate, and hard depending on the degree of occlusion and truncation. It contains 7481 training samples and 7518 testing samples for both point cloud and image data. We partition the 7481 training samples into 3712 samples for train set and 3769 samples for val set to make fair comparisons with other works. We use both train and val set to train our model and report its performance on the official KITTI test server.

2) DAIR-V2X Dataset: The DAIR-V2X Dataset [19] is a newly published multi-modality dataset including 71254 LiDAR samples and 71254 camera samples. It captures raw data in real scenarios from both vehicle and infrastructure, thus creating three subsets of the whole dataset named DAIR-V2X-I, DAIR-V2X-V, and DAIR-V2X-C. Following the work [46], we use DAIR-V2X-I to evaluate the performance of our model on three classes (vehicle, pedestrian, and cyclist). The DAIR-V2X-I contains approximately ten thousand point cloud frames captured by infrastructure sensors and we divide them to train, validation, and test set as 5:2:3 in consistency with [46]. We report the detection results on the validation set.

3) Waymo Open Dataset: The Waymo Open Dataset [20] is a large scale dataset for autonomous driving and contains 798 training sequences and 202 validation sequences. It has three main categories named Vehicle, Pedestrian and Cyclist with two types of difficulty: LEVEL_1 and LEVEL_2. Specifically, LEVEL_1 denotes objects that have over 5 points and LEVEL_2 means objects with at least 1 point. Following the common practice [8], we sample 20% of Waymo training and validation sequences randomly for evaluation.

4) Evaluation Metrics: We adopt the standard 3D average precision with 40 recall positions AP_{3D} to evaluate RAFDet for KITTI test and val set, which is defined as follows:

$$P = \frac{TP}{TP+FP}, R = \frac{TP}{TP+FN},$$
$$AP_{3D} = \frac{1}{40}\sum_{r\in R}\max\left\{P(r)\right\}, R = \left[\frac{1}{40}, \frac{2}{40}, \ldots, 1\right], \tag{8}$$

where TP denotes the predicted box as true positive if the calculated 3D intersection over union (IoU_{3D}) with ground truth is above a predefined threshold, FP and FN are false positive and false negative, respectively. The 3D intersection over union can be formulated as follows:

$$IoU_{3D} = \frac{box(A) \cap box(GT)}{box(A) \cup box(GT)}, \tag{9}$$

where the IoU_{3D} thresholds are set as 0.7, 0.5, 0.5 for car, pedestrian, and cyclist in KITTI dataset, respectively.

For DAIR-V2X Dataset, the used evaluation criterion is 3D average precision with 40 recall positions same as KITTI [18]. We adopt the IoU threshold of 0.5, 0.25, 0.25 to the corresponding classes following [19].

For Waymo Open Dataset, we utilize average precision (AP) and average precision weighted by heading (APH) following [20] with LEVEL_1 difficulty, which are defined as:

$$AP = 100\int_0^1 \max\left\{P\left(r'\right)|r' >= r\right\} dr,$$
$$APH = 100\int_0^1 \max\left\{H\left(r'\right)|r' >= r\right\} dr, \tag{10}$$

where $H(r')$ is weighted by heading accuracy denoted as $(|\theta - \bar{\theta}|, 2\pi - |\theta - \bar{\theta}|)/\pi$. θ and $\bar{\theta}$ are predicted heading and ground truth heading within $[-\pi, \pi]$, respectively.

B. Implementation Details

1) Input Data: For the KITTI dataset, we feed the raw point data with four initial characteristics including space coordinates of x, y, z, and reflected intensity r into the network. To generate range view feature maps, we set the size of range image as 48×512 and conduct cylindrical coordinate transformation from points to image plane, where the range of azimuth is $[-45°, 45°]$. As for the DAIR-V2X dataset, the input is the same

as that of the KITTI dataset since the DAIR-V2X dataset can be easily converted to KITTI format.

2) Data Augmentation: In order to increase the diversity of data samples and avoid the overfitting of the model, we apply several frequently used data augmentation strategies as [14]. Specifically, the first is random world flip along the x axis. Then, we use random world rotation with the rot angle of $[-\frac{\pi}{4}, \frac{\pi}{4}]$. Besides, we adjust the size of objects based on random world scaling, where the scale range is set as $[0.95, 1.05]$. Finally, the data shuffle operation is performed at the training stage to obtain better convergence and robustness of the model.

3) Network Detail: For the point view stream, we adopt the 3D dual-channel backbone of DTSSD [15] to extract point-wise features, which is made up of set abstraction layers and vanilla transformer blocks. The set abstraction layer contains four parts: downsampling, grouping, MLPs, and feature aggregation. Firstly, we limit the raw point cloud scene to 16384 points as input to our backbone. Then, the set abstraction layer and transformer block are used for hierarchical feature extraction. We set the number of downsampled points to $[4096, 2048, 1024]$ and the dimension level for aggregated features is set as $[64, 128, 256]$. For the transformer block, we adopt two heads and one layer without dropout. The input features are $[64, 128]$ and hidden features are $[128, 256]$. Note that, the transformer block is utilized only at the first two stages to save the computational cost. After that, the features from two channels are fed into bidirectional attentive fusion module for in-depth interaction with the input channel dimension $[96, 256]$.

For the range view stream, we take the 2D backbone from [45] and make several modifications to get a lightweight network. Specifically, the backbone includes a geometry-aware kernel, three resblocks, and two upsampling blocks. The resblock has two convolution blocks and one shortcut block with input channel $[32, 32, 128]$ and output channel $[32, 128, 256]$ of different strides $[1, 2, 2]$. The upsampling block contains four blocks of 2D convolution with BN and ReLU to obtain aggregated multi-scale range image features. Finally, we can get two scales of feature maps denoted as $[64, 24, 256]$ and $[32, 48, 512]$. Regarding the range-view augmented fusion module, we only leverage it at the first two stages in accordance with the transformer block. The input to range-view augmented fusion module is set as $[64, 128]$.

Meanwhile, to preserve more foreground points, we employ two layers of MLPs with dimension $[128, 256]$ as semantic segmentation to three classes at the second and third stages. Afterwards, we downsample the point cloud to 512 points and adopt a voting layer composed of two shared MLPs from $256 \rightarrow 128 \rightarrow 3$ to predict corresponding instance centroids. These centroids are fed into another set abstraction layer to get aggregated features of 512 dimensions, which is used for final prediction. The detection head is composed of simple linear layers with BN and ReLU for object classification and regression.

4) Training & Inference: We construct RAFDet based on the OpenPCDet [52] toolbox. The network is trained on 4 NVIDIA GTX 1080Ti GPUs with a total batch size of 8 for 80 epochs with regard to KITTI and DAIR-V2X Datasets. For Waymo Open Dataset, we train our model on 4 NVIDIA GTX 3090Ti GPUs with a total batch size of 8 for 30 epochs. Our model is optimized by Adam optimizer [53] with one-cycle strategy [54]. The initial learning rate is set as 0.003 for KITTI, 0.001 for DAIR-V2X and 0.003 for Waymo Open Dataset. At the inference stage, we use 1 NVIDIA GTX 1080Ti with a batch size of 4 to evaluate our model. We set the score threshold of 0.1 to maintain potential foreground objects and adopt 3D NMS with 0.01 threshold to filter redundant proposals.

C. Comparison With State-of-the-Arts

1) Evaluation on KITTI Dataset: Table I shows the detection performance of our method on the KITTI test set with 3D average precision. Typically, we classify the existing methods into one-stage and two-stage detectors with different view representations of point cloud. As can be seen from Table I, our method can surpass most one-stage pipelines based on a single view like BEV, PV, or RV in terms of 3D mean average precision. Besides, for the comparison with one-stage multi-modal fusion methods, the proposed RAFDet outperforms [40] by (1.63%, 5.93%, 2.99%) on cyclist class and [39] by (2.12%, 4.44%, 7.81%) on car class in terms of easy, moderate and hard difficulty, respectively. This demonstrates the effectiveness of our method by combining the merits of diverse perspectives in point cloud without introducing extra RGB image. Moreover, we compare our method with several two-stage detectors and still obtain competitive detection performance regarding to three classes, which further validates the superiority of our model. However, we observe that our method achieves inferior performance on the detection of car and pedestrian compared to several methods [37], [40], [51]. We assume there are two reasons for this phenomenon: (1) the images contain much more dense semantic clues than range feature maps projected by sparse point cloud, which helps to equip sparse objects with richer information. (2) the pillar-based methods pay close attention to the shape of pedestrian during the feature extraction process, thus boosting its detection performance.

Apart from the KITTI test set, we present our model performance conducted on the KITTI val set, as shown in Table II. It is obvious that our model achieves prominent improvement in detection performance, especially for the category of small objects, surpassing [41] by (0.39%, 3.13%, 3.88%) on pedestrian class. We assume that it is because the supplementary information of range view in point cloud provides rich semantic features for small objects with sparse points, thus improving the detection accuracy. While for the car category, our method falls behind the distinct performance with other works [8], [42]. We attribute this to the occlusion problem between vehicles and merely relying on the supplementary information of range view can not effectively guide the model to detect occluded cars. Furthermore, the two-stage approaches have an inherent advantage of higher recall than one-stage methods due to the second stage proposal refinement mechanism, thus achieving superior detection performance.

2) Evaluation on DAIR-V2X Dataset: We further verify the effectiveness of our model on the DAIR-V2X dataset and the comparison of RAFDet with other methods is reported in

TABLE I
QUANTITATIVE COMPARISON ON THE KITTI TEST SET IN TERMS OF 3D AVERAGE PRECISION WITH RECALL 40 POSITIONS FOR CAR, PEDESTRIAN, AND CYCLIST

Method	Type	View	3D Car (IoU=0.7)			3D Ped. (IoU=0.5)			3D Cyc. (IoU=0.5)			3D mAP
			Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard	Mod.
VoxelNet [28]	one-stage	BEV	77.47	65.11	57.73	39.48	33.69	31.51	61.22	48.36	44.37	49.05
PointPillars [30]	one-stage	BEV	82.58	74.31	68.99	51.45	41.92	38.89	77.10	58.65	51.92	58.29
DVFENet [47]	one-stage	BEV	86.20	79.18	74.58	43.55	37.50	35.33	78.73	62.00	55.18	59.56
S-AT GCN [48]	one-stage	BEV	83.20	76.04	71.17	44.63	37.37	34.92	75.24	61.70	55.32	58.37
GD-MAE [49]	one-stage	BEV	88.14	79.03	73.55	-	-	-	-	-	-	-
DFAF3D [17]	one-stage	BEV	88.59	79.37	72.21	47.58	40.99	37.65	82.09	**65.86**	**59.02**	62.07
IA-SSD [14]	one-stage	PV	88.34	**80.13**	75.04	46.51	39.03	35.60	78.35	61.94	55.70	60.36
RangeDet [35]	one-stage	RV	85.41	77.36	72.60	-	-	-	-	-	-	-
PPIF [39]	one-stage	PV+IM	86.12	75.37	67.25	-	-	-	-	-	-	-
EOTL [50]	one-stage	PV+IM	79.97	69.13	58.57	48.65	40.11	35.99	75.20	58.96	50.41	56.06
Faraway-Frustum [16]	one-stage	PV+IM	87.45	79.05	**76.14**	46.33	38.58	35.71	77.36	62.00	55.40	59.87
FusionPillars [40]	one-stage	PV+IM+BEV	86.96	75.74	73.03	**55.87**	**48.42**	**45.42**	80.62	59.43	55.76	61.19
PointRCNN [22]	two-stage	PV	86.96	75.64	70.70	47.98	39.37	36.01	74.96	58.82	52.53	57.94
Pointformer [25]	two-stage	PV	87.13	77.06	69.25	50.67	42.43	39.60	75.01	59.80	53.99	59.76
EPNet [37]	two-stage	PV+IM	**89.81**	79.28	74.59	-	-	-	-	-	-	-
PointPainting [51]	two-stage	PV+IM	82.11	71.70	67.08	50.32	40.97	37.87	77.63	63.78	55.89	58.81
APIDFF-Net [41]	two-stage	PV+IM	87.86	78.66	72.51	-	-	-	-	-	-	-
RAFDet (ours)	one-stage	PV+RV	88.24	79.81	75.06	48.95	41.89	38.66	**82.25**	65.36	58.75	**62.35**

The best and second best performance are marked in bold and underline, respectively. BEV: bird's eye view. PV: point view. RV: range View. IM: image view.

TABLE II
QUANTITATIVE COMPARISON ON THE KITTI VAL SET IN TERMS OF 3D AVERAGE PRECISION WITH RECALL 40 POSITIONS FOR CAR, PEDESTRIAN, AND CYCLIST

Method	Type	View	3D Car (IoU=0.7)			3D Ped. (IoU=0.5)			3D Cyc. (IoU=0.5)		
			Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
VoTr-SSD [32]	one-stage	BEV	88.01	78.91	75.88	-	-	-	-	-	-
VoxSeT [55]	one-stage	BEV	91.02	82.01	79.04	59.81	54.23	49.19	89.69	70.37	65.89
IA-SSD [14]	one-stage	PV	91.78	83.37	80.34	61.56	56.85	51.72	89.72	71.41	68.14
DTSSD [15]	one-stage	PV	89.66	82.60	79.63	63.68	57.66	51.95	93.18	74.92	70.30
TANet [56]	one-stage	PV	87.52	76.64	73.86	67.30	60.77	54.45	84.53	61.64	57.44
ACDet [9]	one-stage	BEV+RV	88.91	77.51	74.56	68.47	61.94	57.16	89.51	69.79	65.54
AVFP-MVX [57]	one-stage	BEV+IM	91.24	80.45	76.91	68.12	61.92	57.27	82.62	62.08	58.22
PointRCNN [22]	two-stage	PV	91.81	80.65	78.16	64.05	55.35	48.57	92.64	71.68	67.19
$H^2$3D R-CNN [8]	two-stage	BEV+RV	92.43	83.50	82.84	65.74	57.65	52.83	92.51	73.13	68.62
APIDFF-Net [41]	two-stage	PV+IM	92.42	82.98	80.42	68.99	60.6	54.3	-	-	-
SGF3D [42]	two-stage	PV+IM	**95.48**	**88.40**	**85.94**	-	-	-	-	-	-
EPNet [37]	two-stage	PV+IM	88.76	78.65	78.32	66.74	59.29	54.82	83.88	65.60	62.70
Baseline [14]	one-stage	PV	91.78	83.37	80.34	61.56	56.85	51.72	89.72	71.41	68.14
RAFDet (ours)	one-stage	PV+RV	90.43	82.11	79.24	**69.38**	**63.73**	**58.18**	**93.69**	**76.14**	**72.97**

The best and second best performance are marked in bold and underline, respectively. BEV: bird's eye view. PV: point view. RV: range View. IM: image view.

Table III. It can be seen that our work has a favorable advantage over other BEV or IM methods in detection performance of three classes. Specifically, our method exceeds all baselines by over 7.50% regarding to the detection of pedestrian, which illustrates the effectiveness and generalization of our model. We notice that our model encounters a slight performance drop of vehicle compared with [61]. We argue that this is because the proposed depth prediction and refinement module from [61] facilitates the precise detection of long-distance vehicle. Moreover, the limited number of center points in our method impedes the further improvement of vehicle detection performance.

3) Evaluation on Waymo Open Dataset: To illustrate the efficacy of our RAFDet on large scale dataset, we conduct experiments on Waymo open dataset to evaluate the performance of our model. As can be seen from Table IV, our method outperforms the baseline [14] on three classes by a large margin, which shows that our design can improve the representation learning ability of network for sparse point cloud, thus boosting the detection performance. Besides, we make comparisons with some competitive BEV-based baselines like [62] and obtain promising results on the category of vehicle and cyclist. This manifests the feasibility and effectiveness of our model. Nevertheless, our method lags behind [63] and [62] on the performance of vehicle and pedestrian, respectively. We analyze that it is the two-stage pipeline of [63] and voxel-to-object scheme in [62] prompting better detection performance on the corresponding category.

D. Ablation Studies

We conduct extensive ablation experiments on the KITTI validation set to demonstrate the efficacy of different components and design details in our network. It is noted that all ablative studies are trained on the KITTI train set and evaluated on the KITTI val set by 3D average precision with recall 11 positions.

TABLE III
QUANTITATIVE COMPARISON ON THE DAIR-V2X VAL SET IN TERMS OF 3D AVERAGE PRECISION WITH RECALL 40 POSITIONS FOR CAR, PEDESTRIAN, AND CYCLIST

Method	View	Vehicle (IoU=0.5)			Ped. (IoU=0.25)			Cyc. (IoU=0.25)		
		Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
SECOND [29]	BEV	71.47	53.99	54.00	55.16	52.49	52.52	54.68	31.05	31.19
PointPillars [30]	BEV	63.07	54.00	54.01	38.53	37.20	37.28	38.46	22.60	22.49
MVXNet [58]	BEV+IM	71.04	53.71	53.76	55.83	54.45	54.40	54.05	30.79	31.06
ImvoxelNet [59]	IM	44.78	37.58	37.55	6.81	6.74	6.73	21.06	13.57	13.17
BEVFormer [60]	IM	61.37	50.73	50.73	16.89	15.82	15.95	22.16	22.13	22.06
BEVDepth [61]	IM	**75.50**	**63.58**	**63.67**	34.95	33.42	33.27	55.67	**55.47**	**55.34**
RAFDet (ours)	PV+RV	69.16	56.77	56.80	**66.75**	**62.33**	**62.34**	**64.45**	45.67	46.04

The best and second best performance are marked in bold and underline, respectively. BEV: bird's eye view. PV: point view. RV: range View. IM: image view.

TABLE IV
QUANTITATIVE COMPARISON ON THE WAYMO VAL SET WITH 20% TRAINING SAMPLES (∼ 32K SAMPLES) FOR VEHICLE, PEDESTRIAN, AND CYCLIST

Method	L1 mAP/mAPH	Vehicle (L1) AP/APH	Pedestrian (L1) AP/APH	Cyclist (L1) AP/APH
SECOND [29]	63.47/60.06	70.96/70.34	65.23/54.24	57.13/55.62
PointPillars [30]	63.96/55.96	70.43/69.83	66.21/46.32	55.26/51.75
CenterPoint [64]	70.70/67.88	71.33/70.76	72.09/65.49	68.68/67.39
VoxelNeXt* [62]	72.99/**69.50**	74.07/73.56	**76.76/68.18**	68.14/66.78
IA-SSD [14]	69.19/64.48	70.53/69.67	69.38/58.47	67.67/65.30
LiDAR-RCNN [65]	71.10/66.20	73.50/73.00	71.20/58.70	68.60/66.90
Part-A2 [63]	70.96/67.18	**74.66/74.12**	71.71/62.24	66.53/65.18
Baseline [14]	69.19/64.48	70.53/69.67	69.38/58.47	67.67/65.30
RAFDet (ours)	**73.39**/69.35	74.34/73.67	74.41/64.83	**71.42/69.56**

The best and second best performance are marked in bold and underline, respectively.
* denotes the reproduced result based on the official released codes.

TABLE V
THE ABLATION EXPERIMENTS OF DIFFERENT COMPONENTS IN THE NETWORK

	DCB	BAFM	RAFM	Car Mod (IoU=0.7)	Ped. Mod (IoU=0.5)	Cyc. Mod (IoU=0.5)
1	-	-	-	79.15	55.56	71.41
2	✓	-	-	**79.39**	58.64	72.77
3	-	-	✓	78.91	60.78	74.44
4	✓	✓	-	78.50	60.57	72.11
5	✓	-	✓	78.24	62.10	73.40
6	✓	✓	✓	78.87	**62.79**	**74.80**

The best performance are marked in bold.

TABLE VI
THE ABLATION EXPERIMENTS OF BIDIRECTIONAL ATTENTIVE FUSION MODULE AND RANGE-VIEW AUGMENTED FUSION MODULE WITH 3D MEAN AVERAGE PRECISION

Method		Fusion Type				3D mAP
		Add.	Concat.	RAFM	BAFM	Mod
w/o DCB	1	✓	-	-	-	70.13
	2	-	✓	-	-	69.57
	3	-	-	✓	-	**71.37**
w/ DCB	4	✓	-	-	-	70.31
	5	-	✓	-	-	70.47
	6	-	-	-	✓	**70.58**

The best performance is marked in bold.

1) Effect of Different Components: Table V illustrates the effectiveness of each component adopted in our framework. Note that, DCB, BAFM, and RAFM denote the dual-channel backbone, bidirectional attentive fusion module, and range-view augmented fusion module, respectively. Firstly, we discard the above three components and adopt IA-SSD [14] as a baseline for comparison. When we only add DCB or RAFM into the network, the detection performance for small objects boosts dramatically. This shows that the introduction of dual-channel backbone can extract fine-grained contextual semantics for sparse objects and the proposed range-view augmented fusion module can serve as a supplement to enhance the point-view features. Later, both DCB and BAFM are appended to the network and we observe 1.93% improvement on pedestrian compared with DCB only, which verifies that our proposed BAFM can exploit in-depth interaction of dual channel features. Meanwhile, the combination of DCB and RAFM contributes to the boost of detection performance by 6.54% in terms of pedestrian. It demonstrates that our proposed framework successfully makes use of multi-view features and self-attention mechanism to enhance the feature learning of sparse point cloud, thus boosting the detection performance. Finally, three modules are integrated into the network and we obtain the ultimate best performance.

2) Effect of Bidirectional Fusion and Range-view Fusion: We further conduct experiments to investigate the effect of our proposed bidirectional attentive fusion module and range-view augmented fusion module, respectively. As shown in Table VI, we first remove the dual-channel backbone and discuss the impact of RAFM separately. When we only use addition or concatenation operation to achieve the multi-view fusion, the model performance is suboptimal compared to the usage of RAFM. This indicates that our proposed range-view augmented fusion module can better exploit the supplementary information between multiple views and benefit the detection performance. Next, we discuss the effect of BAFM within DCB by getting rid of RAFM. It can be seen that our model obtains the best performance when equipped with BAFM in comparison with addition or concatenation operation. Therefore, it shows that our proposed bidirectional attentive fusion module is capable of mining distinguishable representation of sparse points from dual channels.

3) The Number of Centers for Prediction: We conduct ablative studies to find out how the number of centers for prediction N affects our model performance. Specifically, we have four different settings of centers, as can be seen in Table VII. It is observed that if N is set too small, e.g., 128 points on the first row, the overall detection performance is unsatisfactory. When

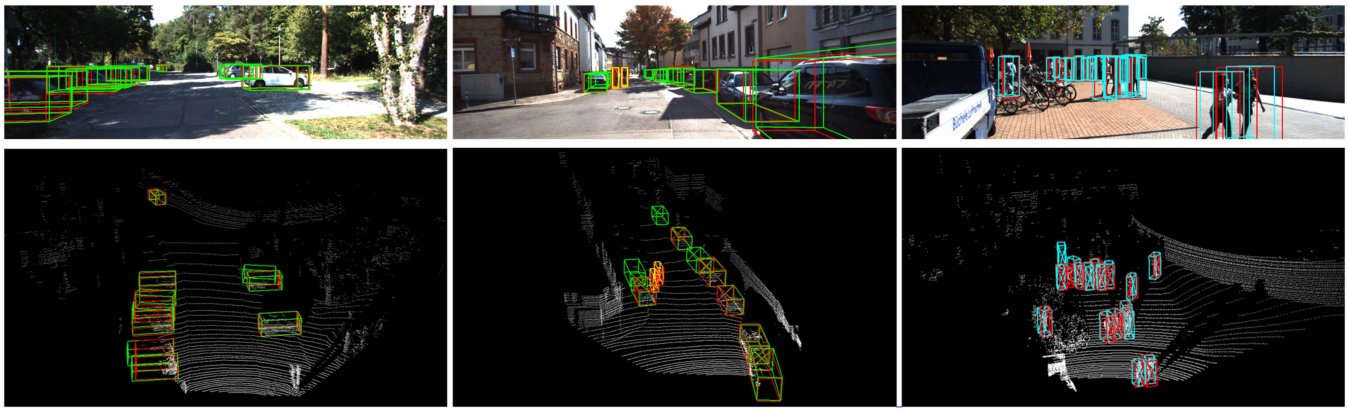

Fig. 8. Qualitative results on the KITTI validation split. We visualize the detection performance in both RGB and point cloud scenes. The ground truth boxes are marked in red and the predicted boxes are marked in green, cyan, and yellow for car, pedestrian, and cyclist, respectively.

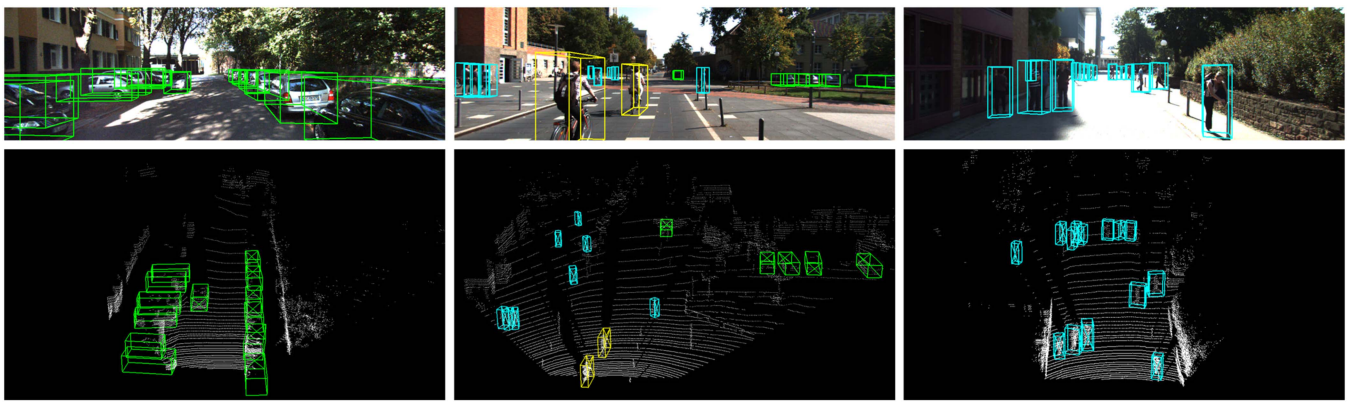

Fig. 9. Qualitative results on the KITTI test split. No ground truth labels are provided on the test split and we directly present the visualization results in RGB and point cloud scenes. The predicted boxes for different classes are set the same as that of the KITTI validation split.

TABLE VII
ANALYSIS ON THE EFFECT OF DIFFERENT NUMBER OF CENTERS FOR PREDICTION

	Number of Centers N	Car Mod (IoU=0.7)	Ped. Mod (IoU=0.5)	Cyc. Mod (IoU=0.5)
1	128	78.71	60.31	71.83
2	256	**79.14**	61.82	73.01
3	512	78.87	**62.79**	**74.80**
4	1024	78.74	62.21	73.66

The best performance are marked in bold.

N is set too large, such as 1024 points on the last row, the performance will not be further improved. Besides, it will occupy more computational resources, which is not beneficial for the training of network. Hence, in consideration of the tradeoff between model performance and efficiency, we finally choose 512 centers in our model for prediction.

4) Tradeoff between Model Complexity and Performance: In Fig. 10(a), we present the comparison of our work with other methods on model parameters and average 3D mean average precision. It can be seen that our model obtains the best performance with acceptable model parameters on average 3D mAP compared with former works. In addition, we present the effectiveness of our model by comparing the precision versus the recall with other algorithms. As shown in Fig. 10(b), our method reaches both high average 3D mAP and recall over other methods. These results demonstrate that our model can achieve the tradeoff between model complexity and performance, thus clarifying the superiority of our network.

5) Effectiveness and limitation of our method: From the above analysis, it is indicated that our proposed multi-view fusion method is effective when integrating the advantages of point cloud and its range view. The designed bidirectional attentive fusion module can excavate in-depth semantic clues during feature extraction for point cloud. Moreover, the proposed range-view augmented fusion module is capable of exploring supplementary information across two different views. Therefore, our model can enhance the representation learning of sparse points to alleviate the sparsity problem of point cloud, thus obtaining the best 3D mAP performance reaching 62.35 over other methods on three object classes. Nevertheless, our method has limitations on the performance of specific categories due to the occlusion problem between various objects, which hinders the further improvement of our network. In the future research, we will

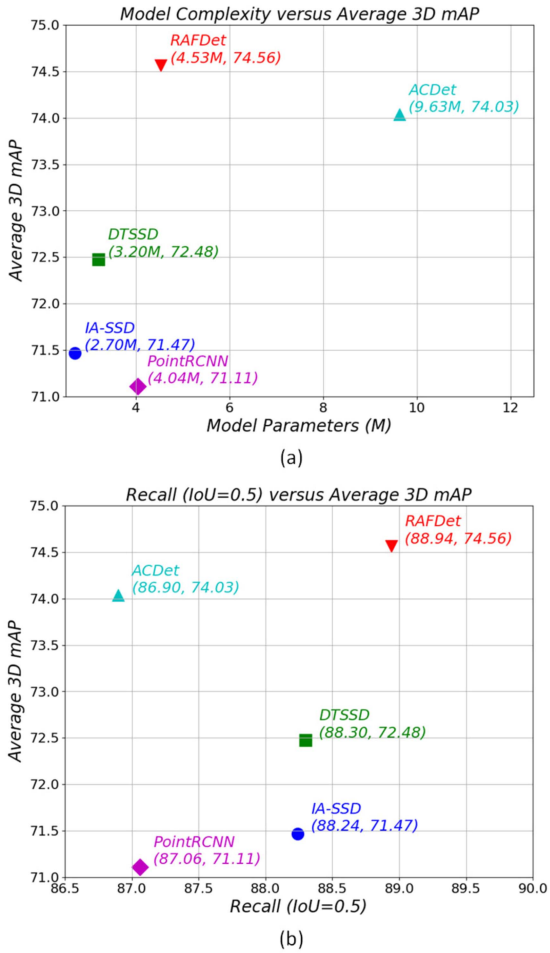

Fig. 10. (a) Tradeoff between model complexity and performance with average 3D mean average precision. (b) Comparison of average 3D mean average precision of different methods with recall (IoU = 0.5).

thoroughly take this issue into account and utilize BEV information to mitigate this problem.

E. Visualization Results

In this section, some qualitative results of RAFDet are exhibited on the KITTI validation and test set, as shown in Figs. 8 and 9. In Fig. 8, we can see that our model has the capability of accurately detecting different types of objects in various scenarios. Furthermore, the detection results in Fig. 9 show the extensibility and effectiveness of our model when referring to the detection of unseen complex scenes.

VI. Conclusion

In this work, we propose a novel range view augmented fusion network RAFDet, which takes advantage of range view as a supplement to alleviate the sparsity of point cloud for 3D object detection. To enhance the feature learning of point cloud, we adopt the dual-channel backbone and devise a bidirectional attentive fusion module to achieve fine-grained interaction of dual channel features. Moreover, we design a range-view augmented fusion module by exploiting the supplementary information between range view and point view, thus equipping sparse points with dense pixel features. In this way, our model has a more rich representation of point cloud and is capable of distinguishing foreground objects from the background more accurately. Extensive experiments on three popular benchmarks show the effectiveness and robustness of our method with the tradeoff between model complexity and performance.

References

[1] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. 2017 IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 6526–6534.
[2] T. Xie et al., "FARP-Net: Local-global feature aggregation and relation-aware proposals for 3D object detection," *IEEE Trans. Multimedia*, vol. 26, pp. 1027–1040, 2024.
[3] Y. Zhang and H. Wu, "3D object detection based on multi-view adaptive fusion," in *Proc. 2022 IEEE Asia-Pacific Conf. Image Process. Electron. Comput.*, 2022, pp. 743–748.
[4] Z. Liu, J. Cheng, J. Fan, S. Lin, Y. Wang, and X. Zhao, "Multi-modal fusion based on depth adaptive mechanism for 3D object detection," *IEEE Trans. Multimedia*, pp. 1–11, 2023, doi: 10.1109/TMM.2023.3270638.
[5] L. Xie, G. Xu, D. Cai, and X. He, "X-view: Non-egocentric multi-view 3D object detector," *IEEE Trans. Image Process.*, vol. 32, pp. 1488–1497, 2021.
[6] Y. Wu et al., "Self-supervised intra-modal and cross-modal contrastive learning for point cloud understanding," *IEEE Trans. Multimedia*, vol. 26, pp. 1626–1638, 2024.
[7] Y. Wang et al., "Pillar-based object detection for autonomous driving," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., Springer, 2020, pp. 18–34.
[8] J. Deng, W. Zhou, Y. Zhang, and H. Li, "From multi-view to hollow-3D: Hallucinated hollow-3D R-CNN for 3D object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4722–4734, Dec. 2021.
[9] J. Xu, G. Wang, X. Zhang, and G. Wan, "ACDet: Attentive cross-view fusion for LiDAR-based 3D object detection," in *Proc. 2022 Int. Conf. 3D Vis.*, 2022, pp. 74–83.
[10] C. Wang, C. Ma, M. Zhu, and X. Yang, "Pointaugmenting: Cross-modal augmentation for 3D object detection," in *Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11789–11798.
[11] Z. Liu et al., "EPNet++: Cascade bi-directional fusion for multi-modal 3D object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 8324–8341, Jul. 2023.
[12] Z. Yin, H. Sun, N. Liu, H. Zhou, and J. Shen, "FGFusion: Fine-grained LiDAR-camera fusion for 3D object detection," in *Proc. Chin. Conf. Pattern Recognit. Comput. Vis.*, 2023, pp. 505–517.
[13] H. Zhu et al., "VPFNet: Improving 3D object detection with virtual point based LiDAR and stereo data fusion," *IEEE Trans. Multimedia*, vol. 25, pp. 5291–5304, 2023.
[14] Y. Zhang et al., "Not all points are equal: Learning highly efficient point-based detectors for 3D LiDAR point clouds," in *Proc. 2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 18931–18940.
[15] Z. Zheng, Z. Huang, J. Zhao, H. Hu, and D. Chen, "DTSSD: Dual-channel transformer-based network for point-based 3D object detection," *IEEE Signal Process. Lett.*, vol. 30, pp. 798–802, 2023.
[16] H. Zhang, D. Yang, E. Yurtsever, K. A. Redmill, and U. Ozguner, "Faraway-frustum: Dealing with LiDAR sparsity for 3D object detection using fusion," in *Proc. 2021 IEEE Int. Intell. Transp. Syst. Conf.*, 2021, pp. 2646–2652.
[17] Q. Tang, X. Bai, J. Guo, B. Pan, and W. Jiang, "DFAF3D: A dual-feature-aware anchor-free single-stage 3D detector for point clouds," *Image Vis. Comput.*, vol. 129, 2022, Art. no. 104594.
[18] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The kitti vision benchmark suite," in *Proc. 2012 IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.
[19] H. Yu et al., "DAIR-V2X: A large-scale dataset for vehicle-infrastructure cooperative 3D object detection," in *Proc. 2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 21329–21338.

[20] P. Sun et al., "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2443–2451.
[21] C. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet : Deep hierarchical feature learning on point sets in a metric space," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5105–5114.
[22] S. Shi, X. Wang, and H. Li, "PointRCNN: 3D object proposal generation and detection from point cloud," in *Proc. 2019 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 770–779.
[23] Z. Yang, Y. Sun, S. Liu, and J. Jia, "3DSSD: Point-based 3D single stage object detector," in *Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11037–11045.
[24] C. Qi, O. Litany, K. He, and L. J. Guibas, "Deep hough voting for 3D object detection in point clouds," in *2019 IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9276–9285.
[25] X. Pan, Z. Xia, S. Song, L. E. Li, and G. Huang, "3D object detection with pointformer," in *Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7459–7468.
[26] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[27] C. Chen, Z. Chen, J. Zhang, and D. Tao, "SASA: Semantics-augmented set abstraction for point-based 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 221–229.
[28] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4490–4499.
[29] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, 2018, Art. no. 3337.
[30] A. H. Lang et al., "Pointpillars: Fast encoders for object detection from point clouds," in *Proc. 2019 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 12689–12697.
[31] J. Deng et al., "Voxel R-CNN: Towards high performance voxel-based 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 1201–1209.
[32] J. Mao et al., "Voxel transformer for 3D object detection," in *Proc. 2021 IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 3144–3153.
[33] J.-Y. Li, C. Luo, and X. Yang, "Pillarnext: Rethinking network designs for 3D object detection in LiDAR point clouds," in *Proc. 2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 17567–17576.
[34] Y. Zhou et al., "End-to-end multi-view fusion for 3d object detection in LiDAR point clouds," in *Proc. Conf. Robot Learn.*, 2020, pp. 923–932.
[35] L. Fan, X. Xiong, F. Wang, N. Wang, and Z. Zhang, "Rangedet: In defense of range view for LiDAR-based 3D object detection," in *Proc. 2021 IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 2898–2907.
[36] C. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum PointNets for 3D object detection from RGB-D data," in *Proc. 2018 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 918–927.
[37] T. Huang, Z. Liu, X. Chen, and X. Bai, "EPNET: Enhancing point features with image semantics for 3D object detection," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., Springer, 2020, pp. 35–52.
[38] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3D object detection," in *Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4603–4611.
[39] J. Ma, X. Wang, H. Duan, and R. Wang, "3D object detection based on the fusion of projected point cloud and image features," in *Proc. 6th Int. Conf. Electron. Inf. Technol. Comput. Eng.*, 2022, pp. 1473–1478.
[40] J. Zhang, D. Xu, Y. Li, L. Zhao, and R. Su, "Fusionpillars: A 3D object detection network with cross-fusion and self-fusion," *Remote. Sens.*, vol. 15, 2023, Art. no. 2692.
[41] W. Yan et al., "Adaptive learning point cloud and image diversity feature fusion network for 3D object detection," *Complex Intell. Syst.*, vol. 10, pp. 2825–2837, 2023.
[42] C. Li, G. Wang, Q. Long, and Z. Zhou, "SGF3D: Similarity-guided fusion network for 3D object detection," *Image Vis. Comput.*, vol. 142, 2024, Art. no. 104895.
[43] Z. Liang, Z. Zhang, M. Zhang, X. Zhao, and S. Pu, "RangeioUDet: Range image based real-time 3D object detector optimized by intersection over union," in *Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7136–7145.
[44] J. Mao, S. Shi, X. Wang, and H. Li, "3D object detection for autonomous driving: A comprehensive survey," *Int. J. Comput. Vis.*, vol. 131, no. 8, pp. 1909–1963, 2023.
[45] T. Cortinhal, G. S. Tzelepis, and E. E. Aksoy, "Salsanext: Fast, uncertainty-aware semantic segmentation of LiDAR point clouds," in *Proc. Int. Symp. Vis. Comput.*, 2020, pp. 1473–1478.
[46] L. Yang et al., "BEVHeight: A robust framework for vision-based roadside 3D object detection," in *Proc. 2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 21611–21620.
[47] Y. He et al., "DVFENet: Dual-branch voxel feature extraction network for 3D object detection," *Neurocomputing*, vol. 459, pp. 201–211, 2021.
[48] L. Wang, C. Wang, X. Zhang, T. Lan, and J. Li, "S-AT GCN: Spatial-attention graph convolution network based feature enhancement for 3D object detection," *CoRR*, vol. abs/2103.08439, 2021, *arXiv:2103.08439*.
[49] H. Yang et al., "GD-MAE: Generative decoder for MAE pre-training on LiDAR point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 9403–9414.
[50] R. Yang, Z. Yan, T. Yang, Y. Wang, and Y. Ruichek, "Efficient online transfer learning for road participants detection in autonomous driving," *IEEE Sensors J.*, vol. 23, no. 19, pp. 23522–23535, Oct. 2023.
[51] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3D object detection," in *Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4603–4611.
[52] O. D. Team, "OpenPCDet: An open-source toolbox for 3D object detection from point clouds," 2020. [Online]. Available: https://github.com/open-mmlab/OpenPCDet
[53] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representations*, Y. Bengio and Y. LeCun, Eds., May 2015, *arXiv:abs/1412.6980*.
[54] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1 - Learning rate, batch size, momentum, and weight decay," 2018, *arXiv:1803.09820*.
[55] C. He, R. Li, S. Li, and L. Zhang, "Voxel set transformer: A set-to-set approach to 3D object detection from point clouds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8417–8427.
[56] Z. Liu et al., "Tanet: Robust 3D object detection from point clouds with triple attention," in *Proc. Conf. Assoc. Advance. Artif. Intell.*, 2020, pp. 11677–11684.
[57] X. Wang, J. Lan, B. Wang, C. Chen, and S. Chen, "AVFP-MVX: Multimodal voxelnet with attention mechanism and voxel feature pyramid," *IEEE Sensors J.*, vol. 23, no. 6, pp. 6139–6149, Mar. 2023.
[58] V. A. Sindagi, Y. Zhou, and O. Tuzel, "MVX-Net: Multimodal VoxelNet for 3D object detection," in *Proc. 2019 Int. Conf. Robot. Automat.*, 2019, pp. 7276–7282.
[59] D. Rukhovich, A. Vorontsova, and A. Konushin, "ImVoxelNet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection," in *Proc. 2022 IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2022, pp. 1265–1274.
[60] Z. Li et al., "Bevformer: Learning Bird's-Eye-View representation from multi-camera images via spatiotemporal transformers," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2022, pp. 1–18.
[61] Y. Li et al., "Bevdepth: Acquisition of reliable depth for multi-view 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 1477–1485.
[62] Y. Chen, J. Liu, X. Zhang, X. Qi, and J. Jia, "VoxelNext: Fully sparse VoxelNet for 3D object detection and tracking," in *Proc. 2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 21674–21683.
[63] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 8, pp. 2647–2664, Aug. 2021.
[64] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3D object detection and tracking," in *Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11779–11788.
[65] Z. Li, F. Wang, and N. Wang, "LiDAR R-CNN: An efficient and universal 3D object detector," in *Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7542–7551.

Zhijie Zheng received the B.S. degree in 2022 from Sun Yat-sen University, Guangzhou, China, where he is currently working toward the M.S. degree with the School of Electronics and Information Engineering. His research interests mainly include computer vision and deep learning, especially for 3D object detection, multi-dataset fusion, and multi-label image classification.

Zhicong Huang received the B.S. and M.S. degrees in 2019 and 2021, respectively, from Sun Yat-sen University, Guangzhou, China, where he is currently working toward the Ph.D. degree with the School of Electronics and Information Technology. His research interests include artificial intelligence and hardware accelerator, and especially for autonomous driving.

Jingwen Zhao received the M.E. degree from Northwest University, Xi'an, China, in 2021. He is currently working toward the Ph.D. degree with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China. His research interests mainly include computer vision, pattern recognition, and deep learning, and in particular focusing on multi-sensor 3D object detection and BEV-Aware detection.

Kang Lin received the B.E. degree from the North University of China, Taiyuan, China, in 2022. He is currently working toward the M.E. degree with the School of Electronics and Information Engineering, Sun Yat-sen University, Guangzhou, China. His research interests include machine vision and image processing, deep learning, and particularly in the field of weakly-supervised temporal action localization.

Haifeng Hu(Member, IEEE) received the Ph.D. degree from Sun Yat-sen University, Guangzhou, China, in 2004. He is currently a Professor with the School of Electronics and Information Technology, Sun Yat-sen University. Since 2000, he has authored or coauthored about 260 papers. His research interests include computer vision, pattern recognition, natural language processing, image processing, and neural computation.

Dihu Chen received the B.S. and M.S. degrees in semiconductor physics from Sichuan University, Chengdu, China, in 1986 and 1989, respectively, and the Ph.D. degree in solid-state electron from the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, in 2000. Since 1989, he has been with Sun Yat-sen University, Guangzhou, China. He is currently working on electronic devices, integrated circuit design, and design methodology.