

# RadarDistill: Boosting Radar-based Object Detection Performance via Knowledge Distillation from LiDAR Features

Geonho Bang<sup>1\*</sup> Kwangjin Choi<sup>1\*</sup> Jisong Kim<sup>1</sup> Dongsuk Kum<sup>2</sup> Jun Won Choi<sup>3†</sup>  
<sup>1</sup>Hanyang University, Korea <sup>2</sup>KAIST, Korea <sup>3</sup>Seoul National University, Korea

<sup>1</sup>{ghbang, kjchoi, jskim}@spa.hanyang.ac.kr <sup>2</sup>dsukum@kaist.ac.kr <sup>3</sup>junwchoi@snu.ac.kr

## Abstract

The inherent noisy and sparse characteristics of radar data pose challenges in finding effective representations for 3D object detection. In this paper, we propose RadarDistill, a novel knowledge distillation (KD) method, which can improve the representation of radar data by leveraging LiDAR data. RadarDistill successfully transfers desirable characteristics of LiDAR features into radar features using three key components: Cross-Modality Alignment (CMA), Activation-based Feature Distillation (AFD), and Proposal-based Feature Distillation (PFD). CMA enhances the density of radar features by employing multiple layers of dilation operations, effectively addressing the challenge of inefficient knowledge transfer from LiDAR to radar. AFD selectively transfers knowledge based on regions of the LiDAR features, with a specific focus on areas where activation intensity exceeds a predefined threshold. PFD similarly guides the radar network to selectively mimic features from the LiDAR network within the object proposals. Our comparative analyses conducted on the nuScenes datasets demonstrate that RadarDistill achieves state-of-the-art (SOTA) performance for radar-only object detection task, recording **20.5%** in mAP and **43.7%** in NDS. Also, RadarDistill significantly improves the performance of the camera-radar fusion model.

## 1. Introduction

While 3D perception based on camera and LiDAR sensors has been widely studied, radar sensors are now gaining attention due to their affordability and reliability in adverse weather conditions. Radar sensors can locate objects in a Bird's Eye View (BEV) and also measure their radial velocity using Doppler frequency analysis. However, when compared to LiDAR or camera sensors, radar's major limitations are its lower spatial resolution and a higher likelihood

\*Equal contributions

†Corresponding author

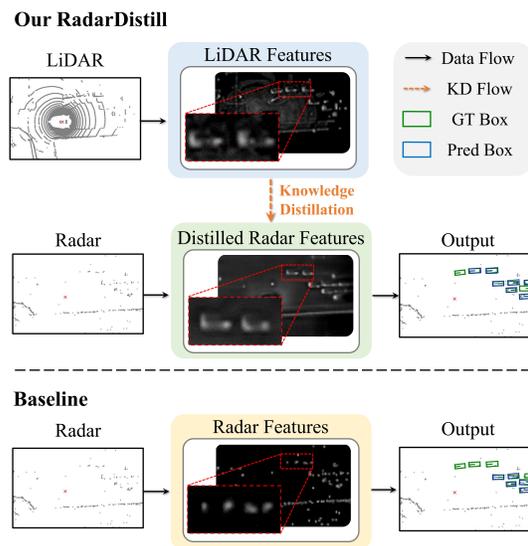


Figure 1. **Illustration of Proposed RadarDistill.** Our RadarDistill method facilitates knowledge transfer from LiDAR features to radar features, enhancing the quality of radar features for Bird's Eye View (BEV) object detection.

of false positives due to multi-path reflections. For decades, traditional object detection and tracking methods, based on manually crafted models, have been developed to overcome these limitations by many radar manufacturers. While deep neural networks (DNNs) have considerably improved 3D perception in camera and LiDAR sensors, similar advancements have not been mirrored in radar sensor-specific architectures. There are only a handful of studies that have applied deep neural networks to radar data. For instance, KPConvPillars [29] and Radar-PointGNN [27] leveraged KPConv [28] and graph neural networks, respectively, to detect objects using radar point clouds. However, these methods have not yet achieved the level of significant improvements realized with camera or LiDAR data. Recently, it was shown that radar can be effectively fused with camera or LiDAR data to enhance the robustness of 3D object

detection [13–15, 21, 32, 43].

This paper focuses on improving the performance of radar-based 3D object detection using deep neural networks. We note that the limited performance of radar is largely due to the challenges in finding effective representations, given the sparse and noisy nature of radar measurements. Inspired by the remarkable success of deep models that encode LiDAR point clouds, our goal is to transfer the knowledge extracted from a LiDAR-based model to enhance a radar-based model.

Recently, knowledge distillation (KD) techniques have shown success in transferring knowledge from one sensor modality to another, thereby improving the representation quality of the target model. To date, various KD methods have been introduced in the literature [3, 4, 8, 11, 16, 33, 41, 42]. BEVDistill [3] transforms both LiDAR and camera features into a BEV format, enabling the transfer of spatial knowledge from LiDAR features to camera features. DistillBEV [33] utilizes the prediction results from either LiDAR or LiDAR-camera fusion models to distinguish between foreground and background, guiding the student model to focus on KD in essential areas. S2M2-SSD [41] determines key areas based on the student model’s predictions and transfers information obtained from a LiDAR-camera fusion model in the key areas. Apart from these approaches, UniDistill [42] employs a universal cross-modality framework that enables knowledge transfer among diverse modalities. This framework is adaptable to different modality pairings, including camera-to-LiDAR, LiDAR-to-camera, and (camera+LiDAR)-to-camera settings.

In this paper, we propose RadarDistill, a novel KD framework designed to enhance the representation of radar data, leveraging LiDAR data. Our study shows that by employing a radar encoding network as a student network and a LiDAR encoding network as a teacher network, our KD framework effectively produces radar features akin to the dense and semantically rich LiDAR features. Although both LiDAR data and its encoding network are used to enhance radar features during the training phase, they are not required in the inference phase.

Our proposed RadarDistill is designed based on three main ideas: 1) Cross-Modality Alignment (CMA), 2) Activation-based Feature Distillation (AFD), and 3) Proposal-based Feature Distillation (PFD). Our study indicates that transferring knowledge from LiDAR to radar features is difficult due to the inherent sparsity of radar data, which complicates the alignment with the more densely distributed LiDAR features. To address this problem, CMA boosts the student network’s capacity and simultaneously increases the ratio of active radar features by implementing multiple layers of dilation operations.

The proposed AFD and PFD minimize the distribution gap between the intermediate features produced by the radar

and LiDAR encoding networks. Initially, AFD conducts Activation-aware Feature Matching on low-level features. Specifically, it divides both radar and LiDAR features into active and inactive regions according to activation intensity of each features and constructs KD losses for each region separately. By assigning greater importance to the KD loss linked with sparsely distributed active regions, AFD enables the network to concentrate on transferring knowledge for significant features.

Next, PFD implements Proposal-level Selective Feature Matching, aimed at narrowing the differences from features associated with the proposals generated by the radar detection head. PFD directs the radar network to generate object features that are similar in shape to the high-level LiDAR features for accurately detected proposals. Conversely, for misdetected proposals such as false positives, the model is guided to suppress falsely activated features, reflecting low activation of the LiDAR features.

Combining all these ideas, our RadarDistill achieves **+15.6%** gain in mAP and a **+29.8%** gain in NDS over the current state-of-the-art (SOTA) performance of radar-only object detection methods on nuScenes benchmark [1]. We also show that when the radar features enhanced by RadarDistill are integrated into a radar-camera fusion model, significant performance improvement is achieved.

The key contributions of our work are as follows:

- Our study is the first to demonstrate that radar object detection can be substantially improved using LiDAR data during the training process. Our qualitative results in Fig. 1 highlight that the radar features acquired through RadarDistill successfully mimic those of LiDAR, leading to enhanced object detection and localization.
- Our findings reveal that the CMA is a crucial element of RadarDistill. In its absence, we observed a significant drop in performance enhancement. According to our ablation study, CMA plays a pivotal role in resolving inefficient knowledge transfer caused by the different densities of radar and LiDAR point clouds.
- We propose two novel KD methods, AFD and PFD. These methods bridge the discrepancy between radar and LiDAR features, operating at two separate feature levels and utilizing KD losses specifically designed for each level.
- RadarDistill achieves the state-of-the-art performance in the radar-only object detector category on the nuScenes benchmark. It also achieves a significant performance boost for camera-radar fusion scenarios.

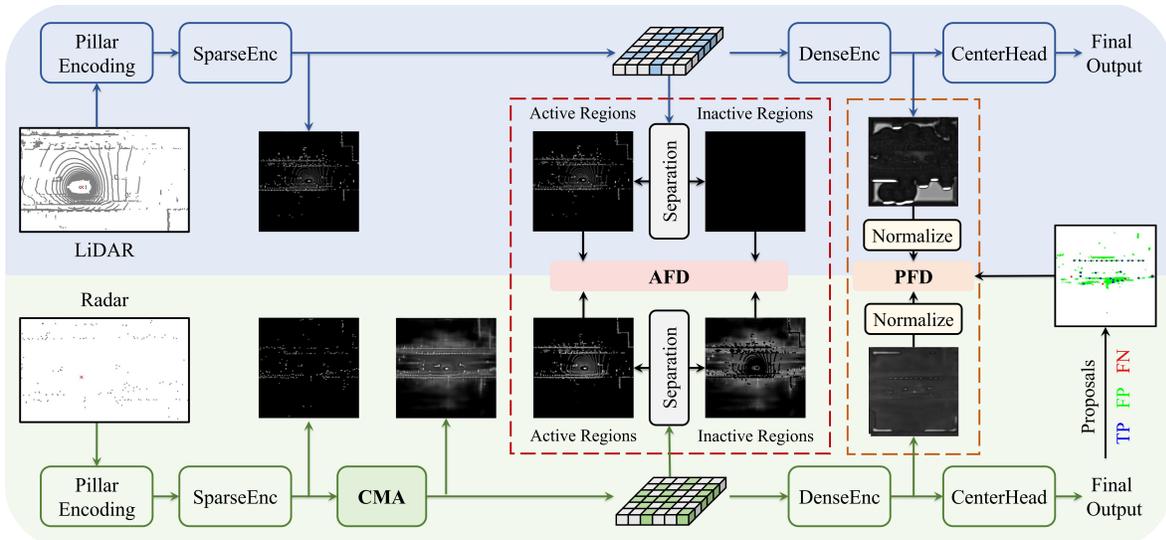


Figure 2. **Overall architecture of RadarDistill.** The input point clouds from each modality are independently processed through Pillar Encoding followed by SparseEnc to extract low-level BEV features. CMA is then employed to densify the low-level BEV features in the radar branch. AFD then identifies active and inactive regions based on both radar and LiDAR features and minimizes their associated distillation losses. Subsequently, PFD conducts knowledge distillation based on proposal-level features obtained from DenseEnc. Note that the LiDAR branch is solely utilized during the training phase to enhance the radar pipeline and is not required during inference.

## 2. Related Works

### 2.1. Radar-based 3D Object Detection

Radar-only 3D object detection models have employed backbone models adopted from various LiDAR-based detectors to suit the specific needs of radar data. Radar-PointGNN [27] utilized GNNs [24] to effectively extract features from sparse radar point clouds through a graph representation. KPConvPillars [29] introduced a hybrid architecture that combined grid-based and point-based approaches for radar-only 3D object detection. Recent studies have concentrated on detection techniques that combine radar with LiDAR or cameras [14, 15, 21, 32, 35, 43]. These methods aimed to complement the limitations of each sensor by using the data obtained from radars. RadarNet [35] employed a voxel-based early fusion and an attention-based late fusion approach to integrate radar and LiDAR data. RCM-Fusion [13] fused radar and camera data at both the feature level and instance level to fully utilize the potential of radar information. CRN [15] leveraged radar data to transform camera image features into a BEV view and then combined these transformed camera features with the radar BEV features using multi-modal deformable attention.

### 2.2. Knowledge Distillation

Knowledge distillation, as a strategy for model compression, transfers the information from a larger teacher model with greater capacity to a smaller student model. The student model mimics the teacher’s intermediate features [22],

prediction logit [10], or activation boundary [9] to acquire knowledge from the teacher model. KD approaches have recently been extended from image classification to object detection [2, 7, 17, 30, 36, 38]. Defeat [7] utilized KD by decoupling features to foreground and background with ground truth. FGD [38] employed spatial and channel attention for 2D object detection, guiding the student model to focus on critical pixels and channels.

In 3D object detection, KD was applied to reduce model complexity in 2D detection [19, 37, 40], or to enhance detection performance through cross-modality knowledge distillation [3, 4, 8, 11, 16, 18, 33, 41, 42]. SparseKD [37] focused on KD for primary regions identified by the teacher model’s predictions, and the student model was initialized with the weights of the teacher to enhance feature extraction. MonoDistill [4] projected LiDAR point clouds onto the image plane and used a 2D CNN to develop an ‘image-version’ LiDAR-based model, effectively bridging the gap in feature representations between LiDAR and camera. BEVDistill [3] projected LiDAR and camera features into the BEV space, effectively transferring 3D spatial knowledge through dense feature distillation and sparse instance distillation. UniDistill [42] proposed a universal framework that was suitable for various teacher-student pairs. UniDistill projected both teacher and student features into BEVs and applied KD on the object features within each ground truth box.

As compared to the existing studies on cross-modality KD methods, our study focuses on developing KD specif-

ically tailored for radar object detection, considering the sparse and noisy nature of radar data.

### 3. Method

In this section, we present the details of the proposed KD framework, RadarDistill. Fig. 2 illustrates the overall architecture of RadarDistill. In Section 3.1, we provide a brief overview of the baseline model employed as both teacher and student models. In Section 3.2, we describe the details of Cross-Modality Alignment, a module for densifying radar features. In Section 3.3, we present the AFD, a novel Activation-aware Feature Matching approach. Finally Section 3.4 outlines the PFD, a Proposal-level Selective Feature Matching approach.

#### 3.1. Preliminary

We use PillarNet [23] as our baseline model for both LiDAR and radar detectors. PillarNet organizes both radar and LiDAR point clouds using a 2D pillar structure of the same size. It then generates two BEV pillar features  $F_{ldr}^{2D}$  and  $F_{rdr}^{2D}$ , utilizing separate pillar encoding networks in the LiDAR and radar branches, respectively. We define the low-level BEV features  $F_{ldr}^{(l)}$  and  $F_{rdr}^{(l)} \in \mathbb{R}^{C \times \frac{H}{8} \times \frac{W}{8}}$  obtained by encoding the pillar features  $F_{ldr}^{2D}$  and  $F_{rdr}^{2D}$  as

$$F_{mod}^{(l)} = SparseEnc(F_{mod}^{2D}), \quad (1)$$

where  $SparseEnc(\cdot)$  denotes a 2D sparse convolution-based encoder (SparseEnc) [6] and  $mod$  represents the detector modality, i.e.,  $mod \in \{ldr, rdr\}$ . Similarly, we describe the high-level BEV features  $F_{ldr}^{(h)}$  and  $F_{rdr}^{(h)} \in \mathbb{R}^{C \times \frac{H}{8} \times \frac{W}{8}}$  formed by encoding the low-level BEV features  $F_{ldr}^{(l)}$  and  $F_{rdr}^{(l)}$ , respectively as

$$F_{mod}^{(h)} = DenseEnc(F_{mod}^{(l)}), \quad (2)$$

where  $DenseEnc(\cdot)$  denotes a 2D dense convolution-based encoder (DenseEnc). The high-level BEV features are further processed through a CenterHead network [39] to generate the final prediction heatmaps for classification, regression, and IoU scoring

$$H_{mod}^{cls}, H_{mod}^{reg}, H_{mod}^{IoU} = CenterHead(F_{mod}^{(h)}). \quad (3)$$

Note that the pipeline in the LiDAR branch serves as the teacher model, while the one in the radar branch acts as the student model in our KD framework. These networks, serving as teacher and student, respectively, do not share weights.

#### 3.2. Cross-Modality Alignment

The objective of CMA is to enhance the density of radar BEV features, thereby facilitating the transfer of knowledge

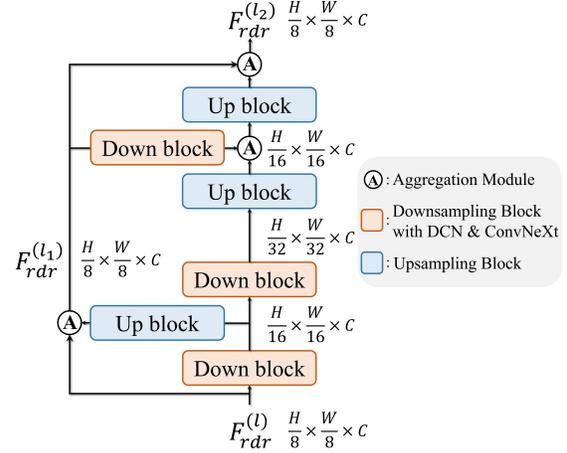


Figure 3. Detailed structure of the proposed CMA module.

from LiDAR features to radar features more effectively. We note that the average number of non-empty pillars in radar features  $F_{rdr}^{(l)}$  is only approximately 10% of the average number found in LiDAR features  $F_{ldr}^{(l)}$ . This considerable difference in the number of non-empty pillars poses challenges in aligning features between the two modalities. Specifically, transferring information from non-empty pillars in LiDAR data to corresponding empty pillars in radar data proves to be impractical. To tackle this challenge, CMA is employed to densify radar features. Our empirical study demonstrates that this densification process significantly contributes to the successful knowledge distillation from LiDAR to radar.

Fig. 3 illustrates the architecture of CMA. CMA comprises the *Down Block*, *Up Block*, and *Aggregation Module*. *Down Block* conducts down-sampling via deformable convolution [5], followed by the ConvNeXt V2 blocks [34]. The *Up Block* conducts up-sampling operations using a 2D transposed convolution. The *Aggregation Module* concatenates two input features and applies a  $1 \times 1$  convolution layer. After applying the *Down Block* and *Up Block* operations twice each, in addition to incorporating a side pathway through the *Aggregation Module*, CMA generates two low-level BEV features,  $F_{rdr}^{(l_1)}$  and  $F_{rdr}^{(l_2)}$ , at a  $1/8$  resolution. As a result of CMA’s dilation operation, these low-level features exhibit increased density compared to the input radar features. (Refer to Fig. 2 for a visual comparison of the features before and after CMA.) These intermediate features are subsequently utilized for knowledge distillation in the following AFD stage. For a detailed structure of the CMA, refer to the Supplementary Materials.

#### 3.3. Activation-based Feature Distillation

AFD conducts Activation-aware Feature Matching on the low-level features obtained from both radar and LiDAR

branches. AFD performs knowledge distillation from the LiDAR BEV features  $F_{ldr}^{(l)}$  to the first radar BEV features  $F_{rdr}^{(l_1)}$ . Simultaneously, it also performs knowledge distillation from  $F_{ldr}^{(l)}$  to the second radar BEV features  $F_{rdr}^{(l_2)}$ .

AFD conducts selective knowledge distillation based on regions using active masks. We determine the active masks  $M_{rdr}^{(l_1)}$ ,  $M_{rdr}^{(l_2)}$ , and  $M_{ldr}^{(l)}$  using the corresponding densified radar features  $F_{rdr}^{(l_1)}$ ,  $F_{rdr}^{(l_2)}$ , and the low-level LiDAR features  $F_{ldr}^{(l)}$ , respectively, as

$$F_{mod,i,j}^{(l')} = \sum_{c=1}^C F_{mod,c,i,j}^{(l')}, \quad (4)$$

$$M_{mod,i,j}^{(l')} = \begin{cases} 1, & \text{if } F_{mod,i,j}^{(l')} > 0, \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

where  $F \in \mathbb{R}^{C \times \frac{H}{8} \times \frac{W}{8}}$  represents BEV features,  $i, j$ , and  $c$  correspond to height, width, and channel indices, respectively,  $mod \in \{ldr, rdr\}$ , and  $l' \in \{l_1, l_2, l\}$ .

Using the active masks obtained from (4) and (5), we can identify the active regions (AR) where both radar and LiDAR features are active, and the inactive regions (IR) where radar features are active and LiDAR features are inactive, i.e.,

$$AR^{(l_n)} = (M_{rdr}^{(l_n)} = 1) \& (M_{ldr}^{(l)} = 1), \quad (6)$$

$$IR^{(l_n)} = (M_{rdr}^{(l_n)} = 1) \& (M_{ldr}^{(l)} = 0), \quad (7)$$

where  $n \in [1, 2]$ . AFD minimizes the difference between radar and LiDAR features in both the AR and IR. AFD is specifically trained to transform radar features to resemble LiDAR features within AR, while simultaneously learning to suppress radar features within IR.

Due to the sparsity of LiDAR data, the AR area is typically much smaller than the IR area. To prevent the model from overly focusing on knowledge distillation within IR, we scale the loss terms associated with AR and IR based on their relative area ratios. We devise adaptive loss weights  $W_{sep}^{(l_n)}$  to balance the loss terms within AR and IR according to their respective pixel counts, i.e.,

$$W_{sep,i,j}^{(l_n)} = \begin{cases} \alpha, & \text{if } (i, j) \in AR^{(l_n)}, \\ \rho^{(l_n)} \times \beta, & \text{if } (i, j) \in IR^{(l_n)}, \\ 0, & \text{otherwise,} \end{cases} \quad (8)$$

where  $\rho^{(l_n)} = \frac{N_{AR^{(l_n)}}}{N_{IR^{(l_n)}}}$  represents the relative importance of the  $IR^{(l_n)}$  over  $AR^{(l_n)}$ ,  $\alpha$  and  $\beta$  are the intrinsic balancing parameters, and  $N_{AR^{(l_n)}}$  and  $N_{IR^{(l_n)}}$  are the number of pixels in  $AR^{(l_n)}$  and  $IR^{(l_n)}$ , respectively. The weight associated with IR is determined proportionally by  $\rho$ , representing the ratio of pixel count for AR to that for IR.

The distillation loss  $L_{low}^{(n)}$  for the  $n$ -th activation features is given by

$$L_{low}^{(n)} = \sum_{c=1}^C \sum_{i=1}^H \sum_{j=1}^W W_{sep,i,j}^{(l_n)} (F_{ldr,c,i,j}^{(l)} - F_{rdr,c,i,j}^{(l_n)})^2. \quad (9)$$

The final AFD loss  $L_{AFD}$  is obtained by averaging the distillation loss functions associated with two densified radar features  $F_{rdr}^{(l_1)}$  and  $F_{rdr}^{(l_2)}$

$$L_{AFD} = \frac{1}{2} \sum_{n=1}^2 L_{low}^{(n)}. \quad (10)$$

### 3.4. Proposal-based Feature Distillation

PFD employs Proposal-level Feature Matching to transfer knowledge from the high-level features of LiDAR to those of radar. This guidance helps the radar network generate object features that closely mimic the high-level features of LiDAR, within object proposals.

DenseEnc is applied to produce two high-level features  $F_{ldr}^{(h_1)}$  and  $F_{ldr}^{(h_2)}$  from the low-level features  $F_{ldr}^{(l)}$  in the LiDAR branch. Similarly, DenseEnc also produces two high-level features  $F_{rdr}^{(h_1)}$  and  $F_{rdr}^{(h_2)}$  from  $F_{rdr}^{(l_2)}$  in the radar branch. DenseEnc generates the first high-level features  $F_{mod}^{(h_1)}$  using the Conv2D-BN-ReLU block followed by a 2D convolution block and a 2D transposed convolution. Here, the Conv2D-BN-ReLU block refers to a sequence of operations comprising 2D Convolution (Conv2D), Batch Normalization (BN), and Rectified Linear Unit (ReLU), while the 2D convolution block consists of six layers of Conv2D-BN-ReLU. For LiDAR branch, DenseEnc produces the second high-level LiDAR features  $F_{ldr}^{(h_2)}$  by concatenate  $F_{ldr}^{(h_1)}$  with  $F_{ldr}^{(l)}$  and applying the 2D Convolution block again. DenseEnc also generates the second high-level radar features  $F_{rdr}^{(h_2)}$  by concatenate  $F_{rdr}^{(h_1)}$  with  $F_{rdr}^{(l_2)}$  and applying the 2D Convolution block in radar branch. Finally, the CenterHead is applied to both  $F_{ldr}^{(h_2)}$  and  $F_{rdr}^{(h_2)}$  to produce the classification heatmaps  $H_{ldr}^{cls}$  and  $H_{rdr}^{cls}$ , respectively.

PFD conducts knowledge distillation solely within the key target regions of high-level BEV features. PFD identifies key target regions based on the predicted radar heatmap  $H_{rdr}^{cls}$  and the ground truth heatmap  $H_{GT}^{cls}$ . The ground truth heatmap is generated by projecting the 3D centers of ground truth boxes onto the BEV space and applying a Gaussian kernel to represent object locations [44]. The identified key target regions are labeled as true positives (TP), false positives (FP), or false negatives (FN) according to

$$TP = (H_{GT}^{cls} > \sigma) \& (H_{rdr}^{cls} > \sigma), \quad (11)$$

$$FP = (H_{GT}^{cls} < \sigma) \& (H_{rdr}^{cls} > \sigma), \quad (12)$$

$$FN = (H_{GT}^{cls} > \sigma) \& (H_{rdr}^{cls} < \sigma), \quad (13)$$

where  $\sigma$  represents the threshold parameter set to 0.1 in our setup. TP and FN regions encompass areas containing real objects, while FP regions correspond to mis-detected regions. Consequently, PFD concentrates on feature alignment in both TP and FN regions while suppressing radar features in FP regions. Due to the inherent imbalance in the proportions of these three regions, distinct weights are applied to the distillation loss terms associated with TP, FN, and FP regions. The proposal-dependent loss weights  $W_{proposal}$  are determined depending on the area of each region

$$W_{proposal,i,j} = \begin{cases} \frac{\lambda_1}{N_{TP}+N_{FN}}, & \text{if } (i,j) \in (TP \cup FN), \\ \frac{\lambda_2}{N_{FP}}, & \text{if } (i,j) \in FP, \\ 0, & \text{otherwise,} \end{cases} \quad (14)$$

where  $\lambda_1$  and  $\lambda_2$  represent the balancing parameters for the respective regions, and  $N_{TP}$ ,  $N_{FN}$  and  $N_{FP}$  denote the number of pixels within TP, FN, and FP regions, respectively.

Finally, the distillation loss defined for the  $m$ th high-level features is weighted by the loss weights  $W_{proposal}$  as

$$L_{high}^{(m)} = \sum_{c=1}^C \sum_{i=1}^H \sum_{j=1}^W W_{proposal,i,j} \left| S_{ldr,c,i,j}^{(h_m)} - S_{rdr,c,i,j}^{(h_m)} \right|, \quad (15)$$

where

$$S_{mod,c,i,j}^{(h_m)} = \frac{\exp(F_{mod,c,i,j}^{(h_m)})}{\sum_{k=1}^C \exp(F_{mod,k,i,j}^{(h_m)})}, \quad (16)$$

where  $mod \in \{rdr, ldr\}$ . Both the radar and LiDAR features  $F_{rdr}^{(h_m)}$  and  $F_{ldr}^{(h_m)}$  go through normalization in the channel dimension, resulting in  $S_{rdr}^{(h_m)}$  and  $S_{ldr}^{(h_m)}$ , respectively. This normalization step aims to align the magnitude scale of the high-level features between the radar and LiDAR branches. The final PFD loss  $L_{PFD}$  is computed by averaging the distillation loss over  $m = 1, 2$ , i.e.,

$$L_{PFD} = \frac{1}{2} \sum_{m=1}^2 L_{high}^{(m)}. \quad (17)$$

### 3.5. Loss Function

The total loss function used to train RadarDistill is obtained from

$$L_{total} = L_{det} + \gamma L_{AFD} + \delta L_{PFD}, \quad (18)$$

where  $\gamma$  and  $\delta$  are the regularization parameters for weighting the loss terms  $L_{AFD}$  and  $L_{PFD}$ , respectively.

## 4. Experiments

### 4.1. Experimental Setup

**Datasets and metrics.** We conduct the experiments on the nuScenes [1] dataset. This dataset contains 700, 150, 150 driving scenes for training, validation, and testing, respectively. The nuScenes dataset gathers radar data using three radar sensors at the vehicle’s front and corners, and two at the rear, providing 360-degree coverage. These sensors collectively capture data at a frequency of 13Hz, ensuring broad coverage up to 250 meters and offering velocity measurement accuracy within  $\pm 0.1km/h$ . Our evaluation utilizes the official metrics from nuScenes, which include mean average precision (mAP) and the nuScenes detection score (NDS).

**Implementation details.** As our baseline model, we employed PillarNet-18, which corresponds to PillarNet [23] with a ResNet-18 backbone. We utilized the Adam optimizer with a learning rate of 0.001, implementing a one-cycle learning rate policy. The weight decay was set to 0.01 with the momentum scheduled from 0.85 to 0.95. We trained the PillarNet-18 model for 20 epochs with a batch size of 16. We used the *Class-Balanced Grouping and Sampling* (CBGS) strategy [45] to mitigate class imbalance issue. Data augmentation techniques were applied, including random scene flipping along the  $X$  and  $Y$  axes, random rotation, scaling, translation, and ground-truth box sampling. Our detection range was set to  $[-54m, 54m]$  for both the  $X$  and  $Y$  axes, and  $[-5m, 3m]$  for the  $Z$  axis. We used pillars of dimensions  $(0.075m, 0.075m)$ .

Our proposed RadarDistill model was trained for 40 epochs. The rest of training setup follows the same training setup as the baseline model. We initialized the radar backbone network using the pre-trained LiDAR backbone network, following the *Inheriting Strategy* presented in [12]. The hyperparameters  $\alpha$  and  $\beta$  in (8) were set to  $3 \times 10^{-4}$  and  $5 \times 10^{-5}$ , respectively. Additionally, the hyperparameters  $\lambda_1$  and  $\lambda_2$  in (14) were set to 5 and 1, respectively. The parameters  $\gamma$  and  $\delta$  in (18) were set to 5 and 25, respectively. The entire training was conducted on 4 NVIDIA RTX 3090 GPUs.

### 4.2. Performance Comparison

Table 1 presents the performance of RadarDistill on *nuScenes testset*. Our RadarDistill achieves significant performance improvements compared to other radar-based object detectors. Specifically, RadarDistill achieves a mAP of 20.5% and a NDS of 43.7%, outperforming the previous state-of-the-art (SOTA) method, KPConvPillars [29], by a considerable margin of +15.6% in mAP and +29.8% in NDS. Table 2 displays the AP performance of RadarDistill for each class. We compare our RadarDistill with other camera-only and camera-radar fusion-based object detec-

Method	Input	KD	mAP $\uparrow$	NDS $\uparrow$	mATE $\downarrow$	mASE $\downarrow$	mAOE $\downarrow$	mAVE $\downarrow$	mAAE $\downarrow$
Radar-PointGNN [27]	R	-	0.5	3.4	1.024	0.859	0.897	1.020	0.931
KPConvPillars [29]	R	-	4.9	13.9	0.823	0.428	0.607	2.081	1.000
PillarNet* [23] (Our baseline)	R	-	8.6	34.7	0.532	0.283	0.615	0.438	0.092
<b>RadarDistill</b>	R	$\checkmark$	<b>20.5</b>	<b>43.7</b>	<b>0.461</b>	<b>0.263</b>	<b>0.525</b>	<b>0.336</b>	<b>0.072</b>

Table 1. **Performance evaluation on nuScenes testset.** The best performed metrics in the radar-only model are marked in bold. “\*” denotes models we have reproduced without applying test time augmentation. Our model achieves the state-of-the-art performance in all metrics.

Method	Input	KD	Car	Truck	bus	Trailer	C.V	Ped.	Motor.	Bicycle	T.C	Barrier
CenterFusion [21]	C, R	-	50.9	25.8	23.4	23.5	7.7	37.0	31.4	20.1	57.5	48.4
FCOS3D [31]	C	-	52.4	<b>27.0</b>	<b>27.7</b>	25.5	<b>11.7</b>	<b>39.7</b>	<b>34.5</b>	<b>29.8</b>	55.7	<b>53.8</b>
MonoDIS [25]	C	-	47.8	22.0	18.8	17.6	7.4	37.0	29.0	24.5	48.7	51.1
CenterNet [44]	C	-	53.6	<b>27.0</b>	24.8	25.1	8.6	37.5	29.1	20.7	<b>58.3</b>	53.3
PillarNet* [23] (Our baseline)	R	-	41.8	11.6	8.4	6.5	0.0	7.4	1.0	0.0	1.1	8.1
<b>RadarDistill</b>	R	$\checkmark$	<b>54.0</b>	15.3	11.3	<b>29.5</b>	5.5	9.2	15.3	0.9	21.7	42.3

Table 2. **Performance evaluation per class on nuScenes testset.** The best performed metrics are marked in bold. ‘C.V’, ‘Ped.’, ‘Motor.’, and ‘T.C.’ represent construction vehicle, pedestrian, motorcycle, and traffic cone, respectively.

tors. RadarDistill exhibits superior performance compared to the baseline, PillarNet-18 [23] across all classes. Notably, it achieves performance gains of +12.2% and +23% for the Car and Trailer classes, respectively. Furthermore, RadarDistill outperforms other camera-only and sensor fusion models in the Car class, which is surprising given the limited resolution of radar data compared to camera data.

### 4.3. Ablation Studies

We conducted ablation studies on the nuScenes validation set to assess the impact of each proposed idea. Our model was trained using 1/7 of the training set and evaluated on the entire validation set.

**Component Analysis.** Table 3 illustrates the performance enhancements achieved by each component of our proposed model. Integrating CMA into the PillarNet-18 baseline results in a 2% improvement in NDS performance. With both CMA and AFD enabled, the NDS performance further improves by 4.4%. Activating PFD yields a 1.0% additional increase in NDS performance. We also tried disabling CMA while other components are enabled in RadarDistill. In this case, the NDS performance drops by 4%, highlighting the critical role of CMA in our RadarDistill model.

**Proposed Distillation Method.** We compare the proposed Distillation method with other well known KD methods. We consider the following methods

- *Baseline\**: This baseline is constructed by integrating CMA into the PillarNet-18 baseline.
- *Baseline\*\**: This baseline is constructed by applying

CMA and AFD to the PillarNet-18 baseline.

- *Complete* [22]: This method applies the fixed weight across entire areas of the low-level BEV features.
- *Gaussian* [26]: This approach generates a 2D Gaussian mask based on the centers of GT boxes and assigns higher weighted distillation loss to areas closer to the foreground.
- *FG/BG* [7]: This strategy separates the foreground and background using GT boxes, applying different weighted distillation losses for each.

Table 4 evaluates the performance of AFD in comparison with other KD methods. We applied AFD and these methods to the Baseline\*, where the low-level BEV features are produced by CMA. We observe that the proposed AFD achieves significantly better performance than other KD methods. Particularly, it achieves the performance gains of 3.7% in AP in car category, 1.3% in mAP, and 2.0% in NDS over the FG/BG method.

Table 5 compares PFD with other KD methods when applied to high-level BEV features. We applied these methods to Baseline\*\* where the high-level features are produced by CMA and AFD. We also confirm that the proposed PFD achieves higher detection performance gains than other KD methods. Specifically, it yields the performance improvements of 0.5% in AP for the car category, 0.2% in mAP, and 1.0% in NDS over the FG/BG method.

**Impact of Scale Normalization in PFD.** Next, we investigate the impact of scale normalization in PFD. Table 6

<i>CMA</i>	<i>AFD</i>	<i>PFD</i>	mAP↑	NDS↑
			5.4	27.3
✓			6.4	29.3
✓	✓		10.9	33.7
✓	✓	✓	<b>11.2</b>	<b>34.7</b>
	✓	✓	7.0	30.7

Table 3. Ablation study for evaluating the contribution from each component of RadarDistill on *nuScenes* validation set.

Region	<i>AFD</i>		
	Car(AP)↑	mAP↑	NDS↑
Baseline*	34.8	6.4	29.3
Complete [22]	39.0	8.6	32.2
Gaussian [26]	40.4	8.8	29.5
FG/BG [7]	42.1	9.6	31.7
Our AFD	<b>45.8</b>	<b>10.9</b>	<b>33.7</b>

Table 4. Comparison of AFD with different KD methods evaluated on *nuScenes* validation set.

presents a comparison of performances with and without scale normalization. We observe that our scale normalization results in a significant improvement of 6.2% in AP (CAR category), 2.7% in mAP, and 2.9% in NDS. This underscores the substantial difference in magnitude scale between radar and LiDAR high-level features, and demonstrates that narrowing this gap significantly boosts the effect of knowledge distillation.

#### Comparison when applied to Radar-Camera Fusion.

Next, we investigate whether the improved features generated by RadarDistill can also lead to performance enhancements in radar-camera fusion methods. Table 7 presents the detection performance when the proposed RadarDistill is intergrated into the radar-camera fusion method, BEVFusion [20]. Because BEVFusion was originally designed for LiDAR-camera fusion, we adapted its design for radar-camera fusion following [15]. Then, RadarDistill was simply incorporated to the radar encoding network in BEVFusion and the entire model was trained end to end. We note that RadarDistill yields performance improvements of 1.8% in Car AP, 1.3% in mAP, and 1.1% in NDS over the baseline BEVFusion model. We believe that further enhancements in performance could be achieved by applying more sophisticated fusion strategies, a direction we leave for future research.

## 5. Conclusions

In this study, we introduced RadarDistill, a novel radar-based 3D object detection method aimed at enhancing radar data features by leveraging information from LiDAR data

Region	<i>PFD</i>		
	Car(AP)↑	mAP↑	NDS↑
Baseline**	45.8	10.9	33.7
Complete [22]	45.2	10.8	33.8
Gaussian [26]	45.9	10.8	33.8
FG/BG [7]	45.6	11.0	33.7
Our PFD	<b>46.1</b>	<b>11.2</b>	<b>34.7</b>

Table 5. Comparison of PFD with different KD methods evaluated on *nuScenes* validation set.

Method	Car(AP)↑	mAP↑	NDS↑
PFD w/o norm	39.9	8.5	31.8
PFD	<b>46.1</b>	<b>11.2</b>	<b>34.7</b>

Table 6. Ablation study for evaluating the impact of scale normalization used in PFD on *nuScenes* validation set. *PFD w/o norm*: PFD without scale normalization. *PFD*: PFD with scale normalization.

Method	Input	KD	Car(AP)↑	mAP↑	NDS↑
BEVFusion* [20]	C,R		65.9	38.3	45.3
RadarDistill-CR	C,R	✓	<b>67.7</b>	<b>39.6</b>	<b>46.4</b>

Table 7. Ablation study for evaluating the impact of RadarDistill on radar-camera fusion models. “\*\*” denotes models we have reproduced without applying test time augmentation.

through knowledge distillation. We proposed three effective techniques—CMA, AFD, and PFD—to make the challenging task of transferring knowledge from LiDAR to radar successful. Our approach emphasizes guiding the radar encoding network to generate features that closely resemble the semantically rich features of LiDAR. CMA densifies radar features, aiding the radar encoding network in learning the complex distribution of LiDAR features better. AFD and PFD target the reduction of discrepancies between radar and LiDAR features in both low-level and high-level BEV features. Our evaluation demonstrated that RadarDistill can yield significant performance improvements over both radar-based and radar-camera fusion-based baselines, establishing state-of-the-art performance in radar-only object detection.

## 6. Acknowledgement

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [No.2021-0-01343-004, Artificial Intelligence Graduate School Program (Seoul National University)] and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) [No.2020R1A2C2012146].

## References

- [1] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multi-modal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 2, 6
- [2] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 30, 2017. 3
- [3] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qin-hong Jiang, and Feng Zhao. Bevdistill: Cross-modal bev distillation for multi-view 3d object detection. *arXiv preprint arXiv:2211.09386*, 2022. 2, 3
- [4] Zhiyu Chong, Xinzhu Ma, Hong Zhang, Yuxin Yue, Haojie Li, Zhihui Wang, and Wanli Ouyang. Monodistill: Learning spatial features for monocular 3d object detection. *arXiv preprint arXiv:2201.10830*, 2022. 2, 3
- [5] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 4
- [6] Benjamin Graham and Laurens Van der Maaten. Sub-manifold sparse convolutional networks. *arXiv preprint arXiv:1706.01307*, 2017. 4
- [7] Jianyuan Guo, Kai Han, Yunhe Wang, Han Wu, Xinghao Chen, Chunjing Xu, and Chang Xu. Distilling object detectors via decoupled features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2154–2164, 2021. 3, 7, 8
- [8] Xiaoyang Guo, Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Liga-stereo: Learning lidar geometry aware representations for stereo-based 3d detector. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3153–3163, 2021. 2, 3
- [9] Byeongho Heo, Minsik Lee, Sangdoon Yun, and Jin Young Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3779–3787, 2019. 3
- [10] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3
- [11] Yu Hong, Hang Dai, and Yong Ding. Cross-modality knowledge distillation network for monocular 3d object detection. In *European Conference on Computer Vision*, pages 87–104. Springer, 2022. 2, 3
- [12] Zijian Kang, Peizhen Zhang, Xiangyu Zhang, Jian Sun, and Nanning Zheng. Instance-conditional knowledge distillation for object detection. *Advances in Neural Information Processing Systems*, 34:16468–16480, 2021. 6
- [13] Jisong Kim, Minjae Seong, Geonho Bang, Dongsuk Kum, and Jun Won Choi. Rcm-fusion: Radar-camera multi-level fusion for 3d object detection. *arXiv preprint arXiv:2307.10249*, 2023. 2, 3
- [14] Youngseok Kim, Sanmin Kim, Jun Won Choi, and Dongsuk Kum. Craft: Camera-radar 3d object detection with spatio-contextual fusion transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1160–1168, 2023. 2, 3
- [15] Youngseok Kim, Juyeb Shin, Sanmin Kim, In-Jae Lee, Jun Won Choi, and Dongsuk Kum. Crn: Camera radar net for accurate, robust, efficient 3d perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17615–17626, 2023. 2, 3, 8
- [16] Marvin Klingner, Shubhankar Borse, Varun Ravi Kumar, Behnaz Rezaei, Venkatraman Narayanan, Senthil Yogamani, and Fatih Porikli. X3kd: Knowledge distillation across modalities, tasks and stages for multi-camera 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13343–13353, 2023. 2, 3
- [17] Quanquan Li, Shengying Jin, and Junjie Yan. Mimicking very efficient network for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6356–6364, 2017. 3
- [18] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. *Advances in Neural Information Processing Systems*, 35:18442–18455, 2022. 3
- [19] Yanjing Li, Sheng Xu, Mingbao Lin, Jihao Yin, Baochang Zhang, and Xianbin Cao. Representation disparity-aware distillation for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6715–6724, 2023. 3
- [20] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird’s-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2774–2781. IEEE, 2023. 8
- [21] Ramin Nabati and Hairong Qi. Centerfusion: Center-based radar and camera fusion for 3d object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1527–1536, 2021. 2, 3, 7
- [22] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014. 3, 7, 8
- [23] Guangsheng Shi, Ruifeng Li, and Chao Ma. Pillarnet: Real-time and high-performance pillar-based 3d object detection. In *European Conference on Computer Vision*, pages 35–52. Springer, 2022. 4, 6, 7
- [24] Weijing Shi and Raj Rajkumar. Point-gnn: Graph neural network for 3d object detection in a point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1711–1719, 2020. 3
- [25] Andrea Simonelli, Samuel Rota Buló, Lorenzo Porzi, Manuel López-Antequera, and Peter Kotschieder. Disentangling monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1991–1999, 2019. 7

- [26] Ruoyu Sun, Fuhui Tang, Xiaopeng Zhang, Hongkai Xiong, and Qi Tian. Distilling object detectors with task adaptive regularization. *arXiv preprint arXiv:2006.13108*, 2020. 7, 8
- [27] Peter Svenningsson, Francesco Fioranelli, and Alexander Yarovoy. Radar-pointgnn: Graph based object recognition for unstructured radar point-cloud data. In *2021 IEEE Radar Conference (RadarConf21)*, pages 1–6. IEEE, 2021. 1, 3, 7
- [28] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019. 1
- [29] Michael Ulrich, Sascha Braun, Daniel Köhler, Daniel Niederlöhner, Florian Faion, Claudius Gläser, and Holger Blume. Improved orientation estimation and detection with hybrid object detection networks for automotive radar. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 111–117. IEEE, 2022. 1, 3, 6, 7
- [30] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4933–4942, 2019. 3
- [31] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 913–922, 2021. 7
- [32] Yingjie Wang, Jiajun Deng, Yao Li, Jinshui Hu, Cong Liu, Yu Zhang, Jianmin Ji, Wanli Ouyang, and Yanyong Zhang. Bi-lrfusion: Bi-directional lidar-radar fusion for 3d dynamic object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13394–13403, 2023. 2, 3
- [33] Zeyu Wang, Dingwen Li, Chenxu Luo, Cihang Xie, and Xiaodong Yang. Distillbev: Boosting multi-camera 3d object detection with cross-modal knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8637–8646, 2023. 2, 3
- [34] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16133–16142, 2023. 4
- [35] Bin Yang, Runsheng Guo, Ming Liang, Sergio Casas, and Raquel Urtasun. Radarnet: Exploiting radar for robust perception of dynamic objects. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 496–512. Springer, 2020. 3
- [36] Chenhongyi Yang, Mateusz Ochal, Amos Storkey, and Elliot J Crowley. Prediction-guided distillation for dense object detection. In *European Conference on Computer Vision*, pages 123–138. Springer, 2022. 3
- [37] Jihan Yang, Shaoshuai Shi, Runyu Ding, Zhe Wang, and Xiaojuan Qi. Towards efficient 3d object detection with knowledge distillation. *Advances in Neural Information Processing Systems*, 35:21300–21313, 2022. 3
- [38] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4643–4652, 2022. 3
- [39] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021. 4
- [40] Linfeng Zhang, Runpei Dong, Hung-Shuo Tai, and Kaisheng Ma. Pointdistiller: Structured knowledge distillation towards efficient and compact 3d detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21791–21801, 2023. 3
- [41] Wu Zheng, Mingxuan Hong, Li Jiang, and Chi-Wing Fu. Boosting 3d object detection by simulating multimodality on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13638–13647, 2022. 2, 3
- [42] Shengchao Zhou, Weizhou Liu, Chen Hu, Shuchang Zhou, and Chao Ma. Unidistill: A universal cross-modality knowledge distillation framework for 3d object detection in bird’s-eye view. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5116–5125, 2023. 2, 3
- [43] Taohua Zhou, Junjie Chen, Yining Shi, Kun Jiang, Mengmeng Yang, and Diange Yang. Bridging the view disparity between radar and camera features for multi-modal fusion 3d object detection. *IEEE Transactions on Intelligent Vehicles*, 8(2):1523–1535, 2023. 2, 3
- [44] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 5, 7
- [45] Benjin Zhu, Zhengkai Jiang, Xiangxin Zhou, Zeming Li, and Gang Yu. Class-balanced grouping and sampling for point cloud 3d object detection. *arXiv preprint arXiv:1908.09492*, 2019. 6