Cross-Modality Knowledge Distillation Network for Monocular 3D Object Detection

Yu Hong[1], Hang Dai[2]*, and Yong Ding[1]*

[1] Zhejiang University, Zhejiang, China
[2] MBZUAI, Abu Dhabi, UAE
yuhong_1999@zju.edu.cn hang.dai@mbzuai.ac.ae dingy@vlsi.zju.edu.cn
* Corresponding authors.

Abstract. Leveraging LiDAR-based detectors or real LiDAR point data to guide monocular 3D detection has brought significant improvement, e.g., Pseudo-LiDAR methods. However, the existing methods usually apply non-end-to-end training strategies and insufficiently leverage the LiDAR information, where the rich potential of the LiDAR data has not been well exploited. In this paper, we propose the **C**ross-**M**odality **K**nowledge **D**istillation (CMKD) network for monocular 3D detection to efficiently and directly transfer the knowledge from LiDAR modality to image modality on both features and responses. Moreover, we further extend CMKD as a semi-supervised training framework by distilling knowledge from large-scale unlabeled data and significantly boost the performance. Until submission, CMKD ranks 1^{st} among the monocular 3D detectors with publications on both KITTI *test* set and Waymo *val* set with significant performance gains compared to previous state-of-the-art methods. Our code will be released at https://github.com/Cc-Hy/CMKD.

1 Introduction

Detecting objects in 3D space is crucial to a wide range of applications, such as augmented reality, robotics and autonomous driving. The 3D detectors are to generate 3D bounding boxes with size, location, orientation and category parameters to localize and classify the detected objects, enabling the system to perceive and understand the surrounding environment. In autonomous driving [16, 12, 2], 3D object detectors can be categorized into LiDAR point cloud based [9, 52, 53], stereo image based [32, 26, 56], monocular image based [22, 54, 49, 39] and multi-modality based methods [23, 43] according to the input resources. Compared with LiDAR sensors, monocular cameras have many unique advantages such as low price, colored information and dense perception, and monocular 3D object detection has become an active research area. However, there exists a large performance gap between LiDAR-based 3D detectors and monocular 3D detectors due to the lack of precise 3D information in monocular images. Thus, monocular 3D object detection is an extremely challenging task.

Recently, leveraging LiDAR-based detectors or real LiDAR point data to guide monocular 3D detection has brought significant improvement. For example, Pseudo-LiDAR methods [58, 59, 41] transform the 2D images into 3D pseudo

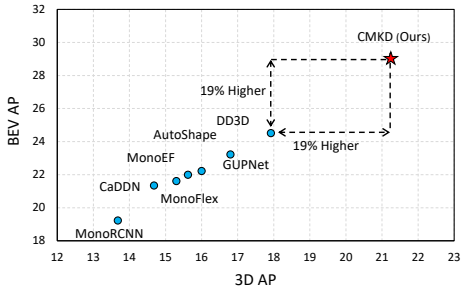

Fig. 1. Comparison between top-ranking monocular 3D detectors and CMKD (Ours) on KITTI leaderboard [16] for Car with $3D\ AP$ and $BEV\ AP$ metrics. Higher is better.

points via depth estimation networks [13, 27], and use a LiDAR-based detector [46, 24] to perform 3D detection. Many methods [58, 59, 41, 50, 44], including most of the Pseudo-LiDAR methods, use real LiDAR point data to provide accurate 3D supervision during training, e.g., projecting the LiDAR points onto the image plane for a sparse ground truth depth map for depth supervision.

However, there is still room for improvement in this pattern. These methods only mimic the LiDAR data representation and extract some plain information from the LiDAR data like depth maps, but do not consider further exploiting deeper information such as high-dimensional features. To transfer the useful knowledge from the LiDAR data more efficiently and directly, we propose a novel cross-modality knowledge distillation network to mitigate the gap between the image modality and the LiDAR modality on both features and responses. Specifically, we use a LiDAR-based detector as the teacher model to provide the Bird's-Eye-View (BEV) feature map which inherits accurate 3D information from LiDAR points as the feature guidance. And we use the predictions of the teacher model with the awareness of soft label quality as the response guidance. We then transform the knowledge from the LiDAR-based teacher model to the image-based student model in both feature and response level via distillation, thus more fully exploiting the beneficial information of the LiDAR data.

Additionally, the unlabeled data, e.g., raw images and LiDAR points without ground truth 3D labels, is widely used by monocular 3D detectors [58, 62, 41, 63, 44], but only for a sub-task like depth pre-training, and the potential of the unlabeled data has not been well exploited for the main detection task. To this end, we further extend CMKD as a semi-supervised training framework to technically better leverage the unlabeled data. Given a relatively small number of labeled samples to train the LiDAR-based teacher model, we can directly train CMKD on unlabeled data with the teacher model extracting beneficial information and transferring it to the student model. Unlike the existing methods who only use the unlabeled data for depth pre-training, CMKD can directly perform the multi-task training with unlabeled data in an end-to-end manner. Meanwhile, our semi-supervised training pipeline generalizes the application of CMKD in real-world scenes, where we only need to label a small portion of

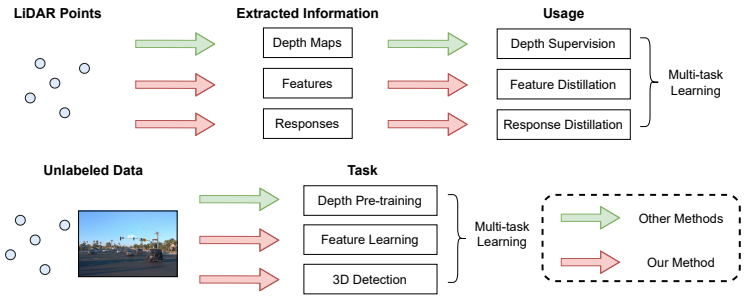

Fig. 2. Comparison between other methods and CMKD (Ours). For the LiDAR points, CMKD performs knowledge distillation by extracting features and responses from them, not only the depth maps. For the unlabeled data, CMKD can directly use it for multi-task training including feature learning and 3D detection, not only the depth pre-training sub-task.

the data and can use the whole set for training, thus significantly reducing the annotation cost. We show the major difference between CMKD and the existing methods using LiDAR point information and unlabeled data in Fig. 2.

We summarize our contributions in three-fold: **i**) We propose a novel cross-modality knowledge distillation network to directly and efficiently transfer the knowledge from LiDAR modality to image modality on both features and responses, digging deeper in cross-modality knowledge transfer and significantly improving monocular 3D detection accuracy (Fig. 1). **ii**) We propose to distill the unlabeled data with our CMKD framework in a semi-supervised manner. With a relatively small amount of annotated data, CMKD can be trained end-to-end on the unlabeled data, which enables it to be trained with state-of-the-art performance while significantly reducing annotation cost. **iii**) CMKD ranks 1^{st} among the monocular 3D detectors with publications on KITTI *test* set [16] and Waymo *val* set [12] with remarkable performance gains.

2 Related Works

LiDAR-based 3D Detection LiDAR-based 3D detection [52, 53, 47, 66, 30, 28, 29, 31] has been developing rapidly in recent years. LiDAR sensors capture precise 3D measurement information from the surroundings in the form of unordered 3D points $(x, y, z, \cdots)$, where x, y, z are the absolute 3D coordinates of each point and the others could be additional information such as reflection intensity. Point-based methods, e.g., PointNet [47], PointNet++ [48] take the raw point clouds as input, and extract point-wise features through structures like multi-layer perceptron for 3D object detection. Voxel-based methods, e.g., VoxelNet [66], SECOND [61] extend the representation of 2D image as pixels into 3D space by dividing 3D space into voxels. Thanks to the precise 3D information provided by point clouds, LiDAR-based methods have achieved relatively high accuracy on different 3D object detection benchmarks [16, 12, 2].

Pseudo-LiDAR based 3D Detection Pseudo-LiDAR based 3D detectors [58, 63, 41, 59, 6] benefit from both mimicking the LiDAR data representation and the accurate 3D information provided by the LiDAR data. These methods first transform the 2D images into intermediate 3D representations like pseudo point clouds via depth estimators [13, 27], and then perform LiDAR-based methods on them. In this work, we take advantage of the LiDAR data by extracting features and responses, thus further exploiting the potential of the LiDAR data.

Leveraging Unlabeled Data Leveraging large-scale unlabeled data has been very popular among monocular 3D detectors especially for depth estimation pre-training. Pseudo-LiDAR [58] and many extension works [41, 55, 62] use an off-the-shelf depth estimator like DORN [13] that is well-trained on the unlabeled KITTI Raw for depth estimation. DD3D [44] leverages extra super-large scale unlabeled data DDAD15M for depth pre-training which leads to significant performance improvements for monocular 3D detection. A major improvement is that CMKD can directly use the unlabeled data to perform multi-task training in an end-to-end manner, not only the depth pre-training sub-task.

Knowledge Distillation The standard knowledge distillation [37, 20, 21, 14, 60, 8] is performed between different models on the same modality. Usually, a well-trained heavy teacher model is applied on the input to obtain informative representations and then supervise the features or the output logits of a simple student model, compressing the model yet maintaining high accuracy. In this work, we use the cross-modality knowledge distillation between the LiDAR modality and monocular image modality for monocular 3D detection.

Difference between CMKD and Similar Methods The general idea of knowledge distillation has been explored by some existing works, and we explain the difference. LIGA-Stereo [18] focuses on the feature distillation only, and it is proposed for the stereo 3D detection task. MonoDistill [7] converts the representation of LiDAR modality to image modality, while CMKD converts the representation of image modality to LiDAR modality. LPCG [45] uses a LiDAR-based detector to generate pseudo labels without considering the intermediate high-dimensional features. Moreover, LPCG applies a one-size-fits-all method to use the soft labels, while we further take the soft label quality into account and use the quality-aware confidence scores to adaptively penalize the contribution of each soft label. DA-3d [62] applies non-end-to-end training strategies with fixed 2D detector and depth estimator, and only the trainable feature extractor is optimized for the feature distillation. But the monocular detector in CMKD is fully differentiable and can be trained end-to-end with all components jointly optimized. Overall, CMKD jointly uses feature and response distillation for the monocular 3D detection task in an end-to-end manner. With the novel design of using totally soft guidance, CMKD can further handle large-scale unlabeled data which is easy to collect for autonomous driving cars, extending its application in real-world scenarios and boosting the performance. Apart from the general idea of knowledge distillation, CMKD is also different in the way to perform distillation with novel explorations in each distillation module, achieving new state-of-the-art performance on KITTI and Waymo benchmarks.

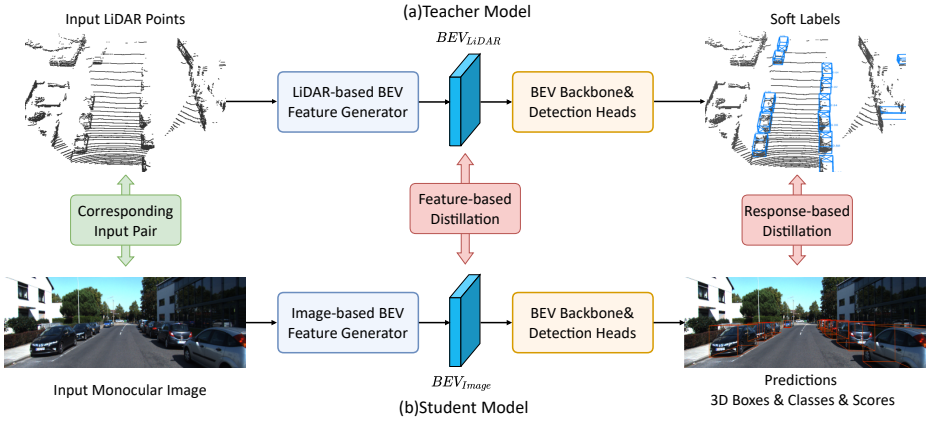

Fig. 3. Overview of the cross-modality knowledge distillation (CMKD) network for monocular 3D detection. (a) A pre-trained LiDAR-based 3D detector as the teacher model that extracts beneficial information from the LiDAR point data as soft guidance. (b) A trainable monocular 3D detector as the student model with the feature-based and response-based knowledge distillation.

3 Method

3.1 Framework Overview

Fig. 3 illustrates the overview of the cross-modality knowledge distillation network for monocular 3D object detection. The general idea is simple and straightforward. The key is to extract the same type of feature and response representations from both input LiDAR points and input monocular images, and perform knowledge distillation between the two modalities. Our framework includes a pre-trained LiDAR-based 3D detector as the teacher model, which extracts information from LiDAR points as soft guidance in the training stage, a trainable monocular 3D detector as the student model, and the cross-modality knowledge distillation on both features and responses.

Training. In the training stage, we take the monocular image and the corresponding LiDAR points as the input pair. The pre-trained teacher model is inferred only from input LiDAR points to provide the BEV feature maps that inherit accurate 3D information from LiDAR points as the feature guidance, and the predictions with 3D bounding boxes, object classes and their corresponding confidence scores as the response guidance. The student model is trainable to generate BEV feature maps and 3D object detection results from monocular images, and uses the soft guidance in both feature level and response level from the teacher model for useful knowledge transfer.

Inference. In the inference stage, we use the student model alone to perform 3D object detection with monocular images as input only.

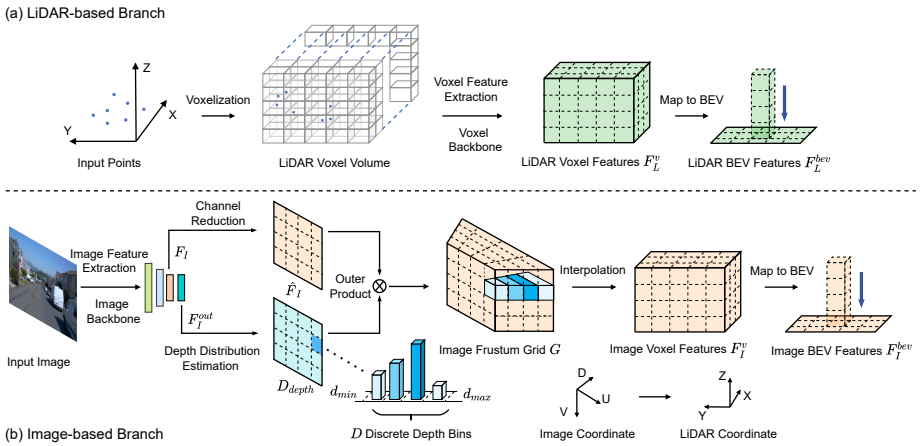

Fig. 4. BEV feature map generation. (a) The LiDAR-based branch. (b) The image-based branch.

3.2 BEV Feature Learning

LiDAR-based. We use SECOND [61], a simple LiDAR-based baseline as the teacher model to extract the BEV features from LiDAR points. The input points are subdivided into 3D voxels, which are fed to a voxel backbone to extract voxel features $F_L^v \in \mathbb{R}^{X \times Y \times Z \times C}$, where X, Y, Z are the width, length and height of the voxel feature volume, and C is the number of feature channels. Then, the voxel features F_L^v are collapsed to a LiDAR BEV feature map with features $F_L^{bev} \in \mathbb{R}^{X \times Y \times Z * C}$ by stacking the height dimension. When pre-training the teacher model, we use Intersection over Unions (IoUs) as the continuous quality labels with the Quality Focal Loss [33] instead of the original one-hot labels in the classification head. Thus, the predicted confidence scores are more IoU-aware to represent the 'quality' of the predictions.

Image-based. For the image-based model, we use the architecture in CaDDN [50] to obtain the BEV features from the monocular image $I \in \mathbb{R}^{W \times H \times 3}$. We use an image backbone, e.g., ResNet [19] to extract image features $\hat{F}_I \in \mathbb{R}^{W_I \times H_I \times C}$, and a depth distribution estimation network, e.g., DeepLabV3 [4] to predict the pixel-wise depth distribution $D_{depth} \in \mathbb{R}^{W_I \times H_I \times D}$. We use the image features together with the estimated depth distributions to construct a frustum grid G with features $F_G \in \mathbb{R}^{W_I \times H_I \times D \times C}$, where D is the number of discrete depth bins, and C is the number of the feature channels. Then, the frustum volume is converted to a cuboid volume in LiDAR coordinate via interpolation operation with known calibration parameters, and we obtain the image voxel features $F_I^v \in \mathbb{R}^{X_I \times Y_I \times Z_I \times C}$. The voxel features are collapsed to a BEV feature map with features $\tilde{F}_I^{bev} \in \mathbb{R}^{X_I \times Y_I \times Z_I * C}$, which then goes through a channel compression network to obtain the image BEV feature map with features $F_I^{bev} \in \mathbb{R}^{X_I \times Y_I \times C}$.

We visualize the BEV feature map generation process in Fig. 4. More details can be found in the *Supplementary Materials*.

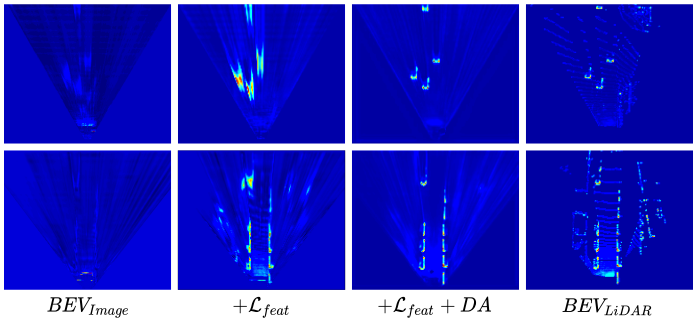

Fig. 5. Illustration of BEV feature maps: the initial BEV feature map from image (1^{st} column), with feature distillation loss $\mathcal{L}_{feat}$ (2^{nd} column), with $\mathcal{L}_{feat}$ and DA module (3^{rd} column), and the corresponding LiDAR BEV feature map (4^{th} column).

3.3 Domain Adaptation via Self-Calibration

The image BEV features F_I^{bev} are different from LiDAR BEV features F_L^{bev} in spatial-wise and channel-wise feature distribution due to the fact that they come from different input modalities with different backbones. We employ a domain adaptation (DA) module to align the feature distribution of F_I^{bev} to that of F_L^{bev} and enhance F_I^{bev} at the meantime. Specifically, we stack five Self-Calibrated Blocks [36] after F_I^{bev} to apply spatial-wise and channel-wise transformations:

$$\hat{F}_I^{bev} = DA(F_I^{bev}) \tag{1}$$

where $\hat{F}_I^{bev} \in \mathbb{R}^{X_I \times Y_I \times C}$ are the enhanced BEV features after the DA module. More details can be found in the *Supplementary Materials*.

3.4 Feature-based Knowledge Distillation

We use the BEV features F_L^{bev} from LiDAR points as the intermediate high-dimensional feature distillation guidance for $\hat{F}_I^{bev}$. We use the mean square error (MSE) to calculate the feature distillation loss:

$$\mathcal{L}_{feat} = MSE(\hat{F}_I^{bev}, F_L^{bev}) \tag{2}$$

Our monocular 3D detector benefits from the feature-based knowledge distillation due to the following aspects. Firstly, F_L^{bev} contains accurate 3D information directly extracted from LiDAR points, e.g., depth and geometry. And the feature representation of F_L^{bev} is well-trained for 3D object detection from point clouds which is more robust to diverse scenarios such as low-light condition and weather changing. We can distill such patterns from F_L^{bev} and transfer them to $\hat{F}_I^{bev}$. As shown in Fig. 5, after feature-based knowledge distillation with the proposed DA module, the object features are highlighted and the patterns of the image BEV features are close to the LiDAR BEV features, which are the

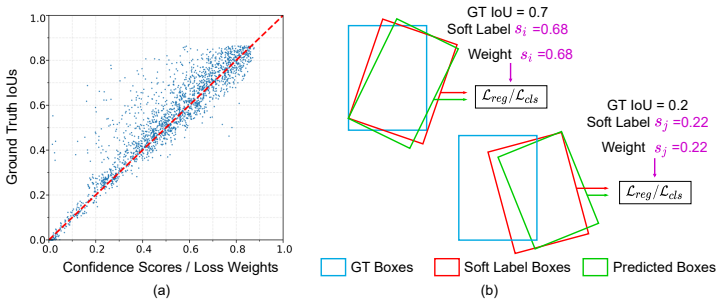

Fig. 6. (a) The IoU confidence scores of soft labels are trained to be positively correlated with the ground truth IoUs. (b) We use the IoU confidence score of the soft label box to indicate its 'quality' and weight the loss $\mathcal{L}_{reg}/\mathcal{L}_{cls}$ in response distillation.

key information to detect 3D objects. Besides, an intermediate feature guidance can ease the condition of over-fitting with high-dimensional information as the regularization term in the overall loss function [51, 17].

3.5 Response-based Knowledge Distillation

The predictions of the teacher model are in form of $(x, y, z, h, w, l, \theta, c, s)$, where (x, y, z) is the center of the 3D bounding box, (h, w, l) is the size of the 3D bounding box, θ is the rotation angle, c is the predicted category and s is the confidence score. And we use the predictions as the response guidance for the student model. Compared with the hard labels, the soft labels contain more information per training sample [20, 65]. Moreover, the teacher model can act as a sample filter for the training samples, e.g., samples which are very difficult to detect for the teacher model tend to be eliminated or assigned with low confidence scores, and the stable samples are assigned with high confidence scores.

Quality-aware Distillation. The loss for response-based distillation includes the regression loss $\mathcal{L}_{reg}$ for 3D bounding boxes and the classification loss $\mathcal{L}_{cls}$ for object classes following the teacher model [61]:

$$\mathcal{L}_{res} = \mathcal{L}_{reg} + \mathcal{L}_{cls} \tag{3}$$

For the i-th anchor, we use the Smooth L1 loss as the regression loss which is penalized by the IoU confidence score of the soft label:

$$\mathcal{L}_{reg} = Smooth\,L1(a_i^{soft}, a_i^{pred}) \times s_i \tag{4}$$

where a_i^{soft} and a_i^{pred} are the bounding box parameters of the soft label and the prediction, and s_i is the IoU confidence score of the soft label box predicted by the teacher model to indicate its 'quality'. Similarly, we use the Quality Focal Loss (QFL) [33] that is penalized by s_i for classification:

$$\mathcal{L}_{cls} = QFL(C_i^{soft}, C_i^{pred}) \times s_i \tag{5}$$

where C_i^{soft} and C_i^{pred} are the classification parameters of the soft label and the prediction. As shown in Fig. 6, the IoU confidence scores of the soft labels are trained to be positively correlated with their ground truth IoUs, which serve to weight the loss produced by each prediction of the student model. Thus, our quality-aware distillation can provide more meaningful and flexible guidance.

3.6 Loss Function

Teacher Model. We train the teacher model with the regression loss $\mathcal{L}_{reg}$ and the classification loss $\mathcal{L}_{cls}$ inherited from SECOND [61] except for replacing the Focal Loss [34] with the Quality Focal Loss [33]:

$$\mathcal{L}_{teacher} = \mathcal{L}_{reg} + \mathcal{L}_{cls} \tag{6}$$

Backbone Pre-training. As with other methods discussed in this paper, we use the depth pre-trained backbone to make the network depth-aware, also, we initialize the backbone with the weights pre-trained on COCO [35] before pre-training. We inherit the depth loss from CaDDN [50] for backbone pre-training:

$$\mathcal{L}_{pre} = \mathcal{L}_{depth} \tag{7}$$

Student Model. The loss function for the student model is defined as the combination of the feature-based and the response-based distillation loss:

$$\mathcal{L}_{student} = \mathcal{L}_{feat} + \mathcal{L}_{res} \tag{8}$$

3.7 Extension: Distilling Unlabeled Data

After the teacher model is pre-trained with the labeled samples, every loss term in the overall loss function for the student model $\mathcal{L}_{student}$ in Eq. (8) does not use any information from manual hard labels. Thus, we can easily and naturally extend CMKD as a semi-supervised training framework with large-scale unlabeled data that is easy to collect for autonomous driving cars. With the teacher model extracting beneficial information and transferring it to the student model as the soft guidance, we can use the partial labeled samples and train the model with the whole unlabeled set. This extended ability of CMKD to handle unlabeled data significantly reduces the annotation cost and brings performance improvements, which generalizes the application of CMKD in real-world scenarios.

Note that, the utilization of unlabeled data is not new for monocular 3D detection task, especially for Pseudo-LiDAR methods. Our contribution is to improve the utilization of unlabeled data with our cross-modality knowledge distillation network. The main difference is that other methods use unlabeled data only for the depth pre-training, a sub-task, but we further use it for knowledge distillation with all components of the network jointly optimized.

Table 1. Results for Car on KITTI *test* set. The best results are in **bold** and the second best results are underlined. We present the results for two experimental setups, CMKD and CMKD*. CMKD is trained with the official training set KITTI *trainval* ($\sim 7.5k$) and CMKD* is trained with the unlabeled KITTI Raw ($\sim 42k$).

Methods	Reference	$3D\ AP$				$BEV\ AP$			
		Easy	Moderate	Hard	Average	Easy	Moderate	Hard	Average
M3D-PRN [1]	ICCV 2019	14.76	9.71	7.42	10.63	21.02	13.67	10.23	14.97
AM3D [41]	ICCV 2019	16.50	10.74	9.52	12.25	25.03	17.32	14.91	19.08
PatchNet [40]	ECCV 2020	15.68	11.12	10.17	12.32	22.97	16.86	14.97	18.27
DA-3d [62]	ECCV2020	16.80	11.50	8.90	12.40	-	-	-	-
D4LCN [10]	CVPR 2020	16.65	11.72	9.51	12.63	22.51	16.02	12.55	17.03
Monodle [42]	CVPR 2021	17.23	12.26	10.29	13.26	24.79	18.89	16.00	19.89
MonoRUn [3]	CVPR 2021	19.65	12.30	10.58	14.18	27.94	17.34	15.24	20.17
MonoRCNN [54]	ICCV 2021	18.36	12.65	10.03	13.68	25.48	18.11	14.10	19.23
PCT [57]	NIPS 2021	21.00	13.37	11.31	15.23	29.65	19.03	15.92	21.53
DFR-Net [67]	ICCV 2021	19.40	13.63	10.35	14.46	28.17	19.17	14.84	20.73
CaDDN [50]	CVPR 2021	19.17	13.41	11.46	14.68	27.94	18.91	17.19	21.35
GUPNet [38]	ICCV 2021	22.26	15.02	13.12	16.80	30.29	21.19	18.20	23.23
DD3D [44]	ICCV 2021	23.22	16.34	14.20	17.92	30.98	22.56	20.03	24.52
CMKD	-	**25.09**	**16.99**	**15.30**	**19.13**	**33.69**	**23.10**	**20.67**	**25.82**
Improvement	-	**+1.87**	**+0.65**	**+1.10**	**+1.21**	**+2.71**	**+0.54**	**+0.64**	**+1.30**
CMKD*	-	**28.55**	**18.69**	**16.77**	**21.34**	**38.98**	**25.82**	**22.80**	**29.20**
Improvement	-	**+5.33**	**+2.35**	**+2.57**	**+3.42**	**+8.00**	**+3.26**	**+2.77**	**+4.68**

4 Experiments

4.1 Datasets

KITTI 3D. KITTI 3D [16] is the most widely used benchmark for 3D object detection consisting of 7481 training images and 7518 testing images as well as the corresponding point clouds, which are denoted as KITTI *trainval* and KITTI *test* respectively. The training set is commonly divided into training split with 3712 samples and validation split with 3769 samples following [5], which are denoted as KITTI *train* and KITTI *val* respectively. The official evaluation metrics are 3D IoU and BEV IoU with the average precision metric, which we denote as $3D\ AP$ and $BEV\ AP$ respectively.

KITTI Raw. KITTI Raw [15] is a raw dataset with $\sim 42k$ unlabeled samples in sequence form. And KITTI 3D is a subset of KITTI Raw chosen with high-quality samples for 3D object detection. Moreover, KITTI Raw is the official depth prediction training set where the training samples are commonly divided into *Eigen* splits [11]. However, there is an overlap [55, 58] between *Eigen train* and KITTI *val*. To avoid this, we use the *Eigen clean* split from DD3D [44] that filters out KITTI *val* from *Eigen train* for the validation experiments.

Waymo Open Dataset. The Waymo Open Dataset [12] is a more recently released dataset with 798 training sequences and 202 validation sequences which consist of about $200k$ samples in total, and we denote them as Waymo *train* and Waymo *val* respectively. CaDDN [50] is the first monocular detector reporting

Table 2. Results for Cyclist and Pedestrian on KITTI *test* set. The best results are in **bold** and the second best results are underlined. We present two setup results, CMKD and CMKD*. CMKD is trained with the official training set KITTI *trainval* ($\sim 7.5k$) and CMKD* is trained with the unlabeled KITTI Raw ($\sim 42k$).

Methods	Cyclist $3D\,AP$ / $BEV\,AP$			Pedestrian $3D\,AP$ / $BEV\,AP$		
	Easy	Moderate	Hard	Easy	Moderate	Hard
DFR-Net [67]	5.69 / 5.99	3.58 / 4.00	3.10 / 3.95	6.09 / 6.66	3.62 / 4.52	3.39 / 3.71
MonoFlex [64]	4.17 / 4.41	2.35 / 2.67	2.04 / 2.50	9.43 / 10.36	6.31 / 7.36	5.26 / 6.29
CaDDN [50]	7.00 / 9.67	3.41 / 5.38	3.30 / 4.75	12.87 / 14.72	8.14 / 9.41	6.76 / 8.17
MonoPSR [25]	8.37 / 9.87	4.74 / 5.78	3.68 / 4.57	8.37 / 9.87	4.74 / 5.78	3.68 / 4.57
GUPNet [38]	5.58 / 6.94	3.21 / 3.85	2.66 / 3.64	14.95 / 15.62	9.76 / 10.37	8.41 / 8.79
DD3D [44]	2.39 / 3.20	1.52 / 1.99	1.31 / 1.79	13.91 / 15.90	9.30 / 10.85	8.05 / 9.41
CMKD	**9.60 / 12.53**	**5.24 / 7.24**	**4.50 / 6.21**	**17.79 / 20.42**	**11.69 / 13.47**	**10.09 / 11.64**
Improvement	**+1.23/+2.66**	**+0.50/+1.46**	**+0.72/+1.46**	**+2.84/+4.52**	**+1.93/+2.62**	**+1.68/+2.23**
CMKD*	**12.52 / 14.66**	**6.67 / 8.15**	**6.34 / 7.23**	13.94 / **16.03**	8.79 / 10.28	7.42 / 8.85
Improvement	**+4.15/+4.79**	**+1.93/+2.37**	**+2.66/+2.48**	-1.01/**+0.13**	-0.97/-0.57	-0.99/-0.56

the performance on Waymo *val* set using samples from the front-camera only, and we follow the same settings for a fair comparison. The official evaluation metrics are 3D IoU with mean average precision and mean average precision weighted by heading, which are denoted as $3D\,mAP$ and $3D\,mAPH$ respectively.

4.2 Experiment Settings

KITTI. We pre-train the teacher model SECOND [61] on KITTI *trainval* for 80 epochs. For ablation studies, we train CMKD on KITTI *train* for 80 epochs or KITTI *train* and *Eigen clean* for 30 epochs according to different experiment settings, and report the performance for Car on KITTI *val*. The image backbone uses depth pre-training on KITTI *train* for 40 epochs. For comparisons on KITTI *test*, we present two experiment setups, CMKD and CMKD*. CMKD is trained with the official training set KITTI *trainval* ($\sim 7.5k$) for 80 epochs, and CMKD* is trained with the unlabeled KITTI Raw ($\sim 42k$) for 30 epochs. Following DD3D [44], the image backbone uses depth pre-training on *eigen clean* split for 10 epochs. We report the performance for all classes on KITTI *test*.
Waymo. We pre-train SECOND [61] on Waymo *train* for 10 epochs with a sampling interval 10. We train CMKD on Waymo *train* for 10 epochs with a sampling interval 5 and report the performance for Vehicle on Waymo *val*. The input image is resized to [960×640]. We do not use depth pre-training on Waymo.

4.3 Results on KITTI test set

We show the results on KITTI *test* in Tab. 1 and Tab. 2. Until submission, for all the three classes, either CMKD or CMKD* achieves new state-of-the-art results with significant improvements on KITTI *test*. With the official KITTI *trainval*, CMKD significantly surpasses the top ranking methods. With additional unlabeled data from KITTI Raw and our semi-supervised training framework,

Table 3. Results for Vehicle on Waymo *val* set. The Best results are in **bold**.

Difficulty	Method	$3D\,mAP$				$3D\,mAPH$			
		Overall	0-30m	30-50m	50m-∞	Overall	0-30m	30-50m	50m-∞
LEVEL 1	M3D-RPN [1]	0.35	1.12	0.18	0.02	0.34	1.10	0.18	0.02
	CaDNN [50]	5.03	14.54	1.47	0.10	4.99	14.43	1.45	0.10
	CMKD	**12.95**	**33.45**	**6.84**	**0.74**	**12.82**	**33.21**	**6.79**	**0.73**
	Improvement	**+7.92**	**+18.91**	**+5.37**	**+0.64**	**+7.83**	**+18.78**	**+5.34**	**+0.63**
LEVEL 2	M3D-RPN [1]	0.33	1.12	0.18	0.02	0.33	1.10	0.17	0.02
	CaDNN [50]	4.49	14.50	1.42	0.09	4.45	14.38	1.41	0.09
	CMKD	**11.44**	**33.04**	**6.22**	**0.58**	**11.33**	**32.80**	**6.17**	**0.57**
	Improvement	**+7.45**	**+18.54**	**+4.80**	**+0.49**	**+6.88**	**+18.42**	**+4.76**	**+0.48**

Table 4. Effectiveness of both distillation and the extension to handle unlabeled data. *Pre.* denotes using depth pre-trained backbone. *Feat.* denotes feature distillation. *Res.* denotes response distillation. *Un.* denotes distilling additional unlabeled data.

Pre.	*Feat.*	*Res.*	*Un.*	$3D\,AP$		
				Easy	Moderate	Hard
×	×	×	×	11.88	8.52	7.40
✓	×	×	×	17.60	13.48	11.81
✓	×	✓	×	18.81	14.49	12.16
✓	✓	×	×	22.20	15.46	13.47
✓	✓	✓	×	23.53	16.33	14.44
✓	✓	✓	✓	**30.17**	**21.54**	**19.44**

CMKD* achieves further boosted performance with significant improvements for Car and Cyclist. This implies that the extension to a semi-supervised framework is efficient in distilling beneficial information from massive unlabeled data and improves the performance. However, the performance for Pedestrian becomes worse with additional unlabeled data, and we conduct extra experiments to explore the reasons for this observation. This lies in the fact that the soft labels provided by the teacher model for Pedestrian are of insufficient quality, which can not provide good guidance for the student model. Detailed experiments and discussions can be found in the *Supplementary Materials*.

Note that DD3D [44], the top method before ours, uses large-scale extra dataset DDAD15M with $\sim 15M$ samples for depth training besides KITTI, while CMKD/CMKD* uses only KITTI and surpasses DD3D by a large margin. Also, other top methods like DD3D [44] or GUPNet [38], works well for Car and Pedestrian but poor for Cyclist, while CMKD works well for all three object classes, which demonstrates its good generalization performance across different object classes. We visualize some prediction results in Fig. 7.

4.4 Results on Waymo Open Dataset

We show the results for Vehicle on Waymo *val* in Tab. 3. With fewer training samples and lower image resolution than that in M3D-RPN [1] and CaDDN [50], CMKD achieves significant improvements on the two difficulty levels considering different distance ranges. We visualize some prediction results in Fig. 7.

Table 5. Effectiveness of components in feature distillation. $\mathcal{L}_{feat}$ denotes the feature distillation loss. DA denotes the domain adaptation module.

KITTI *train*					KITTI *train* + *Eigen clean*				
$\mathcal{L}_{feat}$	DA	$3D\ AP$ Easy	Moderate	Hard	$\mathcal{L}_{feat}$	DA	$3D\ AP$ Easy	Moderate	Hard
×	×	18.81	14.49	12.16	×	×	26.07	19.17	17.45
✓	×	21.72	15.24	12.93	✓	×	28.52	20.74	18.73
✓	✓	**23.53**	**16.33**	**14.44**	✓	✓	**30.17**	**21.54**	**19.44**

Table 6. Effectiveness of components in response distillation. $\mathcal{L}_{res}$ denotes the response distillation loss. $Conf.$ denotes the IoU-aware confidence scores of soft labels used to perform weighted supervision.

KITTI *train*					KITTI *train* + *Eigen clean*				
$\mathcal{L}_{res}$	$Conf.$	$3D\ AP$ Easy	Moderate	Hard	$\mathcal{L}_{res}$	$Conf.$	$3D\ AP$ Easy	Moderate	Hard
×	×	20.20	13.46	11.47	×	×	27.24	19.56	17.67
✓	×	22.78	15.69	13.97	✓	×	28.16	20.67	18.97
✓	✓	**23.53**	**16.33**	**14.44**	✓	✓	**30.17**	**21.54**	**19.44**

4.5 Ablation Studies

Effectiveness of both distillation. As discussed earlier in this paper, existing Pseudo-LiDAR methods [58, 63, 41, 59] leverage the LiDAR data via depth pre-training, while we further exploit the LiDAR data via knowledge distillation. As can be seen in Tab. 4, when using the depth pre-trained image backbone, the performance significantly improves against the baseline, indicating that the accurate depth information provided by LiDAR points is helpful for the task. And when each of our distillation module is applied, the performance is further significantly improved, indicating that our novel utilization of the LiDAR data via distillation can more fully exploit the potential of the LiDAR data and further improve the performance of the monocular 3D detector.

Effectiveness of distilling unlabeled data. In Sec. 3.7, we introduced the improved utilization of unlabeled data in a semi-supervised manner. As shown in Tab. 4, the performance of CMKD is further improved when unlabeled data is added to distillation pipeline, indicating that our method is efficient in extracting beneficial information from massive unlabeled data and improves the performance. Specifically, we use $\sim 18k$ samples for training with $\sim 3.7k$ labeled and we reduce about 80% annotation cost. Also, we conducted experiments on the impact of different amounts of unlabeled data on the performance. Detailed experiments and discussions can be found in the *Supplementary Materials*.

Apart from jointly applying both distillation, we present novel designs in each distillation module, e.g., the DA module and the quality-aware supervision. We conduct experiments to show that the novel components are helpful for the task.

Effectiveness of components in feature distillation. Here, the baseline is the full version of CMKD without the feature distillation loss $\mathcal{L}_{feat}$ and the DA

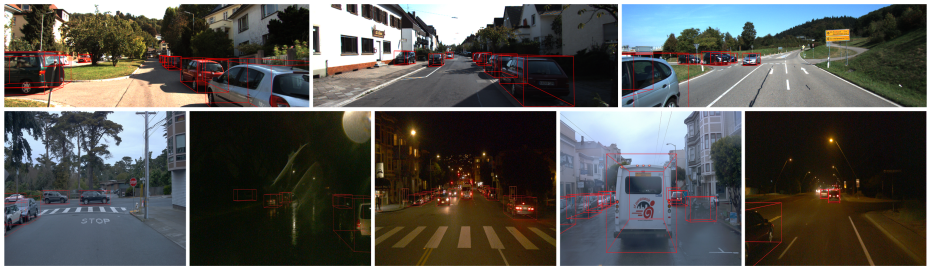

Fig. 7. Qualitative results on KITTI *test* (top line) and Waymo *val* (bottom line). None of the samples were seen during training.

module. As shown in Tab. 5, the performance improves significantly with the two components in the feature distillation. As can be seen from Fig. 5, the BEV feature map shows more clear patterns with highlighted object features when $\mathcal{L}_{feat}$ is added, and avoids smearing effects with aligned BEV features when DA is added. This shows that the components are effective in transferring the knowledge between the two modalities in the feature space.

Effectiveness of components in response distillation. Here, the baseline is the full version of CMKD without the response distillation loss $\mathcal{L}_{res}$ and the quality-aware penalization weights. As shown in Tab. 6, the performance improves with the response distillation loss, and achieves further improvements with the awareness of soft label quality, i.e., with the adaptive supervision. This shows that the components are effective in transferring the knowledge between the two modalities in the response space.

5 Conclusion

In this work, we propose the cross-modality knowledge distillation (CMKD) network to directly and efficiently transfer the knowledge from LiDAR modality to image modality on both features and responses, and significantly improve monocular 3D detection accuracy. Moreover, we extend CMKD as a semi-supervised training framework to distill useful knowledge from large-scale unlabeled data, further boosting the performance while reducing the annotation cost. CMKD achieves new state-of-the-art performance on both KITTI and Waymo benchmarks for monocular 3D object detection with significant performance gains compared to other methods, which shows its great effectiveness.

Broader Impact. Our CMKD framework opens up a new perspective in monocular 3D detection. We believe the effective distillation of unlabeled data demonstrates the potential of CMKD to generalize its application in real-world scenarios, where the unlabeled data is easy to collect for autonomous driving cars.

Acknowledgement. This work was supported by the National Key Research and Development Program of China (Grant No. 2018YFE0183900) and the YUNJI Technology Co. Ltd.

References

1. Brazil, G., Liu, X.: M3d-rpn: Monocular 3d region proposal network for object detection. In: ICCV (2019)
2. Caesar, H., Bankiti, V., Lang, A.H., et al.: nuscenes: A multimodal dataset for autonomous driving. In: CVPR (2020)
3. Chen, H., Huang, Y., Tian, W., et al.: Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In: CVPR (2021)
4. Chen, L., Papandreou, G., Schroff, F., et al.: Rethinking atrous convolution for semantic image segmentation. CoRR **abs/1706.05587** (2017)
5. Chen, X., Kundu, K., Zhu, Y., et al.: 3d object proposals for accurate object class detection. In: NIPS (2015)
6. Chen, Y.N., Dai, H., Ding, Y.: Pseudo-stereo for monocular 3d object detection in autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 887–897 (2022)
7. Chong, Z., Ma, X., Zhang, H., Yue, Y., Li, H., Wang, Z., Ouyang, W.: Monodistill: Learning spatial features for monocular 3d object detection (2022)
8. Dai, X., Jiang, Z., Wu, Z., et al.: General instance distillation for object detection. In: CVPR (2021)
9. Deng, J., Shi, S., Li, P., et al.: Voxel r-cnn: Towards high performance voxel-based 3d object detection. In: AAAI (2021)
10. Ding, M., Huo, Y., Yi, H., et al.: Learning depth-guided convolutions for monocular 3d object detection. CVPR (2020)
11. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NIPS (2014)
12. Ettinger, S., Cheng, S., Caine, B., et al.: Large scale interactive motion forecasting for autonomous driving : The waymo open motion dataset. CoRR **abs/2104.10133** (2021)
13. Fu, H., Gong, M., Wang, C., others.: Deep Ordinal Regression Network for Monocular Depth Estimation. In: CVPR (2018)
14. Furlanello, T., Lipton, Z.C., Tschannen, M., et al.: Born-again neural networks. In: Proceedings of International Conference on Machine Learning (ICML) (2018)
15. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. International Journal of Robotics Research (IJRR) (2013)
16. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: CVPR (2012)
17. Gülçehre, Ç., Bengio, Y.: Knowledge matters: Importance of prior information for optimization. In: ICLR (2013)
18. Guo, X., Shi, S., et al.: Liga:learning lidar geometry aware representations for stereo-based 3d detector. In: ICCV (2021)
19. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. In: CVPR (2016)
20. Hinton, G.E., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. CoRR **abs/1503.02531** (2015)
21. Huang, Z., Wang, N.: Like what you like: Knowledge distill via neuron selectivity transfer. CoRR **abs/1707.01219** (2017)
22. Jörgensen, E., Zach, C., Kahl, F.: Monocular 3d object detection and box fitting trained end-to-end using intersection-over-union loss. CoRR **abs/1906.08070**
23. Ku, J., Mozifian, M., Lee, J., Harakeh, A., Waslander, S.L.: Joint 3d proposal generation and object detection from view aggregation. In: IROS (2018)

24. Ku, J., Mozifian, M., Lee, J., et al.: Joint 3d proposal generation and object detection from view aggregation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2018)
25. Ku*, J., Pon*, A.D., Waslander, S.L.: Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In: CVPR (2019)
26. Königshof, H., Salscheider, N.O., Stiller, C.: Realtime 3D Object Detection for Automated Driving Using Stereo Vision and Semantic Information. In: Proc. IEEE Intl. Conf. Intelligent Transportation Systems (2019)
27. Lee, J.H., Han, M.K., Ko, D.W., et al.: From big to small: Multi-scale local planar guidance for monocular depth estimation (2019)
28. Li, J., Dai, H., Shao, L., Ding, Y.: Anchor-free 3d single stage detector with mask-guided attention for point cloud. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 553–562 (2021)
29. Li, J., Dai, H., Shao, L., Ding, Y.: From voxel to point: Iou-guided 3d object detection for point cloud with voxel-to-point decoder. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 4622–4631 (2021)
30. Li, J., Luo, S., Zhu, Z., Dai, H., Krylov, A.S., Ding, Y., Shao, L.: 3d iou-net: Iou guided 3d object detector for point clouds. arXiv preprint arXiv:2004.04962 (2020)
31. Li, J., Sun, Y., Luo, S., Zhu, Z., Dai, H., Krylov, A.S., Ding, Y., Shao, L.: P2v-rcnn: point to voxel feature learning for 3d object detection from point clouds. IEEE Access **9**, 98249–98260 (2021)
32. Li, P., Chen, X., Shen, S.: Stereo r-cnn based 3d object detection for autonomous driving. In: CVPR (2019)
33. Li, X., Wang, W., Wu, L., et al.: Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In: NIPS (2020)
34. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
35. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
36. Liu, J., Hou, Q., Cheng, M., et al.: Improving convolutional networks with self-calibrated convolutions. In: CVPR (2020)
37. Lu, X., Li, Q., et al.: Mimicdet: Bridging the gap between one-stage and two-stage object detection. In: ECCV (2020)
38. Lu, Y., Ma, X., Y ang, L., et al.: Geometry uncertainty projection network for monocular 3d object detection. arXiv preprint arXiv:2107.13774 (2021)
39. Luo, S., Dai, H., Shao, L., Ding, Y.: M3dssd: Monocular 3d single stage object detector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6145–6154 (2021)
40. Ma, X., Liu, S., Xia, Z., et al.: Rethinking pseudo-lidar representation. In: ECCV (2020)
41. Ma, X., Wang, Z., Li, H., et al.: Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. In: ICCV (2019)
42. Ma, X., Zhang, Y., Xu, D., et al.: Delving into localization errors for monocular 3d object detection. In: CVPR (2021)
43. Pang, S., Morris, D.D., Radha, H.: Clocs: Camera-lidar object candidates fusion for 3d object detection. In: IROS (2020)
44. Park, D., Ambrus, R., Guizilini, V.o.: Is pseudo-lidar needed for monocular 3d object detection? In: ICCV (2021)

45. Peng, L., Liu, F., Yu, Z., et al.: Lidar point cloud guided monocular 3d object detection. CoRR (2021)
46. Qi, C.R., Wei, L., Wu, C., et al.: Frustum pointnets for 3d object detection from rgb-d data. In: CVPR (2018)
47. Qi, C.R., Su, H., Mo, K., et al.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: CVPR (2017)
48. Qi, C.R., Yi, L., Su, H., et al.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: NIPS (2017)
49. Qin, Z., Wang, J., Lu, Y.: Monogrnet: A geometric reasoning network for 3d object localization. AAAI (2019)
50. Reading, C., Harakeh, A., Chae, J., Waslander, S.L.: Categorical depth distribution network for monocular 3d object detection. CVPR (2021)
51. Romero, A., Ballas, N., Kahou, S.E., et al.: Fitnets: Hints for thin deep nets. In: ICLR (2015)
52. Shi, S., Guo, C., Jiang, L., et al.: Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In: CVPR (2020)
53. Shi, S., Wang, X., Li, H.: Pointrcnn: 3d object proposal generation and detection from point cloud. In: CVPR (2019)
54. Shi, X., Ye, Q., Chen, X., et al.: Geometry-based distance decomposition for monocular 3d object detection. In: ICCV (2021)
55. Simonelli, A., Bulò, S.R., Porzi, L., et al.: Demystifying pseudo-lidar for monocular 3d object detection. CoRR **abs/2012.05796** (2020)
56. Sun, J., Chen, L., Xie, Y., et al.: Disp r-cnn: Stereo 3d object detection via shape prior guided instance disparity estimation. In: CVPR (2020)
57. Wang, L., Zhang, L., Zhu, Y., et al.: Progressive coordinate transforms for monocular 3d object detection. In: NIPS (2021)
58. Wang, Y., Chao, W.L., Garg, D., et al.: Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In: CVPR (2019)
59. Weng, X., Kitani, K.: Monocular 3D Object Detection with Pseudo-LiDAR Point Cloud. arXiv:1903.09847 (2019)
60. Xu, Z., Hsu, Y., et al.: Training shallow and thin networks for acceleration via knowledge distillation with conditional adversarial networks. In: ICLR (2018)
61. Yan, Y., Mao, Y., Li, B.: SECOND: sparsely embedded convolutional detection. Sensors (2018)
62. Ye, X., Du, L., Shi, Y., et al.: Monocular 3d object detection via feature domain adaptation. In: ECCV (2020)
63. You, Y., Wang, Y., Chao, W.L., et al.: Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving. In: ICLR (2020)
64. Zhang, Y., Lu, J., Zhou, J.: Objects are different: Flexible monocular 3d object detection. In: CVPR (2021)
65. Zheng, W., Tang, W., Jiang, L., et al.: Se-ssd: Self-ensembling single-stage object detector from point cloud. In: CVPR (2021)
66. Zhou, Y., Tuzel, O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. CoRR **abs/1711.06396** (2017)
67. Zou, Z., Ye, X., Du, L., et al.: The devil is in the task: Exploiting reciprocal appearance-localization features for monocular 3d object detection. In: ICCV (2021)