CaKDP: Category-aware Knowledge Distillation and Pruning Framework for Lightweight 3D Object Detection

Haonan Zhang[1] Longjun Liu[1*] Yuqi Huang[1] Zhao Yang[1] Xinyu Lei[1] Bihan Wen[2]
[1]National Key Laboratory of Human-Machine Hybrid Augmented Intelligence,
National Engineering Research Center for Visual Information and Applications,
and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
[2]Nanyang Technological University

Abstract

*Knowledge distillation (KD) possesses immense potential to accelerate the deep neural networks (DNNs) for LiDAR-based 3D detection. However, in most of prevailing approaches, the suboptimal teacher models and insufficient student architecture investigations limit the performance gains. To address these issues, we propose a simple yet effective Category-aware Knowledge Distillation and Pruning (CaKDP) framework for compressing 3D detectors. Firstly, CaKDP transfers the knowledge of two-stage detector to one-stage student one, mitigating the impact of inadequate teacher models. To bridge the gap between the heterogeneous detectors, we investigate their differences, and then introduce the student-motivated category-aware KD to align the category prediction between distillation pairs. Secondly, we propose a category-aware pruning scheme to obtain the customizable architecture of compact student model. The method calculates the category prediction gap before and after removing each filter to evaluate the importance of filters, and retains the important filters. Finally, to further improve the student performance, a modified IOU-aware refinement module with negligible computations is leveraged to remove the redundant false positive predictions. Experiments demonstrate that CaKDP achieves the compact detector with high performance. For example, on WOD, CaKDP accelerates CenterPoint by **half** while boosting L2 mAPH by **1.61**%. The code is available at* https://github.com/zhnxjtu/CaKDP.

1. Introduction

LiDAR-based 3D object detection (LiDAR-3DOD) is one of the effective ways of scene understanding, and it is crucial for autonomous driving [11, 51, 53], VR [23], and robotics [34], et al. Recently, the release of several high-quality point cloud datasets [1, 12, 43] promote the application of DNNs on LiDAR-3DOD. However, the cumbersome parameters and computations hinder the practical deployment of these DNNs. Hence, effective model compression schemes should be investigated for detector acceleration.

Knowledge distillation (KD) [4, 16, 33, 52, 62] can be used for model compression. It applies a compact student model to capture the knowledge of large-scale teacher model, and then the student model is used for inferring like the teacher model. Besides, network pruning [14, 27, 44, 63] is also one of the useful compression techniques. It removes the redundant connections or structures to get the lightweight model. These compression methods are well-explored on 2D vision [9, 16, 31, 33], but often yield unsatisfactory gains on 3D detection, due to the distinct information recorded in different types of raw data [7, 60].

Recently, a few methods start to explore the KD [7, 50, 56, 60] and pruning techniques [20, 26] for compression on 3D detection task. However, several methods [7, 50, 60] face challenges that can lead to inadequate results: **(1) Leaking the application of potent teacher models**. To ensure the consistency of transferred information, most of previous works [7, 50, 60] conduct KD between the homogeneous distillation pairs. However, when achieving a one-stage student detector, employing another detector with the same architecture but wider width as the teacher hinders student's improvement, due to the capacity limitation of teacher. **(2) Leaking the exploration for lightweight student architectures**. Different from KD on 2D vision tasks [4, 9, 22, 47, 62], there are few off-the-shelf student networks in 3D detection. Previous methods [7, 50, 60] obtain the student network by reducing the width of each layer or backbone with the same ratio. However, the number of indispensable filters for each layer is different, thus these schemes neglect to search the optimal compact architecture, leading to inferior student. **(3) Leaking the removal of student's prediction errors**. After distillation, the student detectors still generate a large number of false positive (FP) predictions, reducing the detection precision.

[*] Corresponding author, email: liulongjun@xjtu.edu.cn

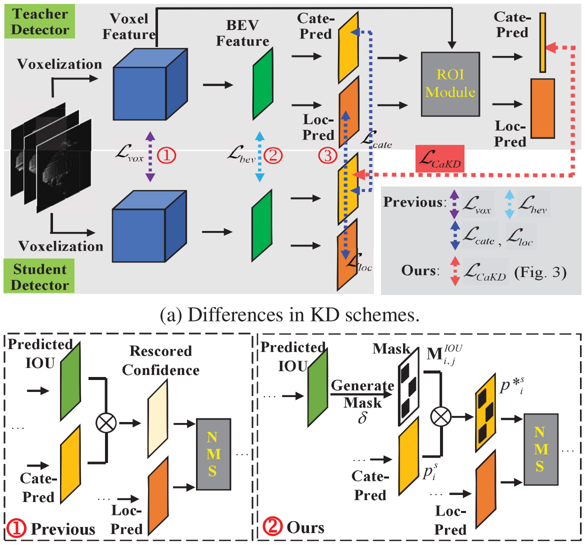
To address these issues, in this paper, we propose a simple yet effective Category-aware Knowledge Distillation and Pruning (CaKDP) framework. CaKDP mainly consists of the following three components:

Firstly, to prevent imprecise student detectors caused by distilling the knowledge of capacity-limited teacher detectors, our method conducts KD between heterogeneous detectors. By comparing the output of one- and two-stage detectors, a notable gap is observed in the category predictions (Cate-Preds). Hence, we propose student-motivated category-aware KD (SKD) to achieve precise one-stage student detectors by bridging the gap of Cate-Pred of distillation pairs. In each epoch, our SKD utilizes non-maximum suppression (NMS) to select Cate-Preds of representative samples (RSs) from the student detector, which serve as the student's knowledge. Besides, we employ the positions of these RSs as queries to retrieve corresponding features in the teacher detector, and then the corresponding second stage Cate-Preds of these selected features are utilized as teacher's knowledge (as shown in Fig. 3(b)).

Furthermore, to search for the optimal lightweight architecture of the student model and make student customizable, we introduce a category-aware pruning scheme. Inspired by [19, 28, 48, 64] and our observations (Section 3.2), the Cate-Pred plays an important role in 3D detection. Hence, we propose to measure the importance of each filter by the Cate-Pred gap before and after pruning, and selectively remove unimportant filters to prune the detector.

Finally, we remove FP samples in prediction results to enhance the detection precision. Since FP samples exhibit smaller intersection over union (IOU) with the ground truth bounding boxes, we leverage an IOU head to predict IOU values and subsequently eliminate predicted bounding boxes with smaller IOU scores. Our approach is different from previous methods [19, 48, 64], which utilize the IOU head for category correction (as shown in Fig. 1(b)).

Our contributions can be summarized as: **(1)** We propose category-aware KD to conduct distillation between heterogeneous detectors. Our method can achieve the one-stage detectors with higher performance. **(2)** We introduce category-aware architecture pruning scheme to the 3D detector compression. Our method simultaneously measures different types of filters to obtain the customizable student model. **(3)** We propose modified IOU-aware refinement module. It removes redundant FP samples to further ensure the precision of detectors. **(4)** Extensive experiments on KITTI [12] and Waymo Open Dataset (WOD) [43] illustrate that our CaKDP can achieve faster and more accurate detectors. For example, on KITTI, CaKDP reduces the parameters of SECOND by **3.1**× while achieving a **5.05**% improvement in moderate mAP@R40; Besides, on WOD-mini, it **halves** the computational overhead for CenterPoint while boosting L2 mAPH by **1.60**%.



(a) Differences in KD schemes.

(b) Differences in IOU-aware refining.

Figure 1. Differences between our approach and previous methods. In (a), [60] conducts KD at ① and ②. [7, 50] conduct KD at ② and ③. Compared to these previous KD schemes, our method transfers the Cate-Preds of the second stage of teacher to one-stage student (detailed in Fig. 3), rather than the aligned knowledge at the first stage. In (b), [19, 48, 64] leverage scheme ①.

2. Related Work

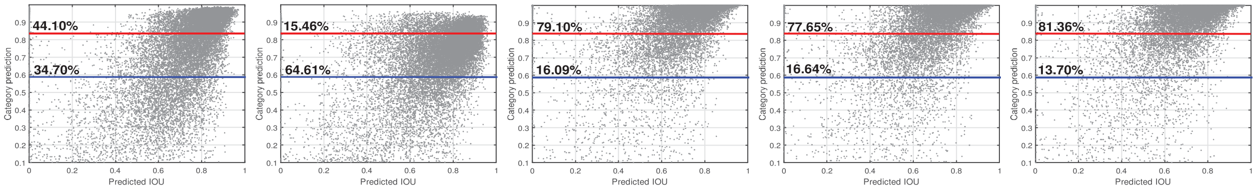
2.1. LiDAR-based 3D Object Detection

LiDAR-3DOD aims to detect the objects by analyzing point cloud data. Recently, plenty of DNNs are introduced for LiDAR-3DOD, which can be clustered into two types:

One-stage detectors for LiDAR-3DOD. One-stage 3D detectors are faster with less memory footprint. Some point-based one-stage detectors [35, 36] directly extract the features from raw points for detection. In contrast, numerous voxel-based methods [6, 49, 66] execute detection by analyzing the voxels generated by point clouds. VoxelNet [66] is the pioneer of this kind of work. To accelerate VoxelNet, abundant detectors, such as SECOND [49], PointPillar [25], VoxelNeXt [6] and CenterPoint [54] are further proposed.

Two-stage detectors for LiDAR-3DOD. To achieve more accurate detection, two-stage 3D detectors [10, 38–41] introduce a region of interest (ROI) module to capture the shallower features for prediction refinement. For example, Point-RCNN [38] introduces the SA module to achieve accurate results. PV-RCNN [39] refines the predictions by aggregating multi-scale features. Moreover, Voxel-RCNN [10] proposes voxel ROI pooling to correct the predictions.

In this paper, we mainly focus on achieving efficient and accurate one-stage voxel-based detectors, because the second stage of detectors brings more memory and computation usage, and point-based methods exhibit more irregular memory access patterns and are general less efficient [26].



(a) SECOND. (b) CenterPoint. (c) Voxel-RCNN. (d) PV-RCNN. (e) PartA2.

Figure 2. Visualization of predictions of detectors. x- and y-axis represent the predicted IOU and category prediction, respectively. SECOND [49] and CenterPoint [54] are one-stage detectors. Voxel-RCNN [10], PV-RCNN [39] and PartA2 [40] are two-stage detectors. The proportions of predictions above the red line and between the red and blue lines are marked on the red and blue lines, respectively.

2.2. Knowledge Distillation

Knowledge Distillation (KD) can be leveraged for model compression by transferring the knowledge from large-scale teacher model to the compact student one. Dozens of KD schemes achieve consistent effectiveness in 2D vision [2–4, 9, 16, 24, 33, 37, 52, 55, 62]. However, since the point clouds are sparse and irregular, directly conducting KD on 3D detection by image-based KD schemes usually leads to inadequate performance [60]. Recently, a few methods initially explore dedicated KD for detector acceleration on LiDAR-3DOD [8, 50, 60, 65]. For example, Yang et al. [50] propose pivotal position logit KD to compress 3D detectors.

While these approaches [7, 50, 60, 65] are applicable for 3D detector compression, most of them conduct KD on homogeneous distillation pairs, and neglect to delve into the gap between heterogeneous detectors (as shown in Fig. 1). Although a few methods [5, 32] in 2D vision distill between heterogeneous models, they introduce additional specialized modules in inference phase [32], or involve complex feature decomposition [5]. Additionally, in LiDAR-3DOD, previous KD methods [7, 50, 60] typically predefine the same retaining width of each layer to get student models in a coarse-grained fashion, which ignores the fine-grained exploration of architectures.

2.3. Architecture Pruning

Structured pruning [17, 27, 30, 31, 42, 61] is another effective technique for model compression, it removes the redundant structures to obtain the compact models, without generating the sparse and irregular pruned weights [13, 14]. Li et al. [27] utilize L1 norm to evaluate and prune the filters. Besides, a few methods [18, 31, 57, 59] leverage the intermediate layer feature and its transformation to measure the importance of filters for pruning.

The above methods have shown promising results in 2D classification, and a few recent researches [20, 26] have delved into pruning for 3D vision. Lee et al. [26] propose a parameter pruning scheme based on the spatial point distribution. This method leads to irregular weights, and can only compress 3D convolutions. Huang et al. [20] propose the channel pruning plug-in that only compress the point-based models. In this paper, we introduce an effective structured pruning scheme for 3D detector compression, which can prune both 2D and 3D backbones simultaneously.

3. Methodology

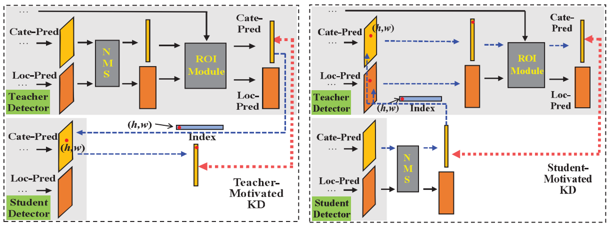
3.1. Preliminary

We first briefly introduce the notations in our method. Most of one-stage voxel-based detectors [49, 54] consist of L^1-layer 3D backbone, L^2-layer 2D backbone and L^3-layer detection head. $\mathcal{W} = \{\mathbf{W}_1, ..., \mathbf{W}_L\}$ is utilized to represent the convolutions (filters) in the model, where $\mathbf{W}_i = \{\mathbf{w}_{i,1}, ..., \mathbf{w}_{i,N_i}\}$ represents the i-th layer with N_i convolutions, $L = L^1 + L^2 + L^3$. Given a point cloud dataset $\{x_i | i = 1, ..., N^{PC}\}$ with N^{PC} frames. After voxelization, 3D backbone and 2D backbone are leveraged to extract features sequentially, and then the features are input to detection head to generate the category and location predictions. We leverage $p_i^s \in R^{N^s \times C}$ to represent the category prediction of the i-th frame, where N^s and C represent the number of anchors (for anchor-based methods) and categories, respectively. For two-stage detectors [39, 41], an additional ROI head is introduced to refine the predictions of the first stage. $p_i^t \in R^{N^t \times C}$ represents the category prediction of the second stage, where N^t represents the number of ROIs. $N^t \ll N^s$, as ROIs are partial high-quality predictions selected through NMS from all anchors.

3.2. Category-aware Knowledge Distillation

We first introduce category-aware KD in our framework to improve the performance of compact one-stage detectors.

The gap between heterogeneous detectors. To build a bridge for KD, we begin by investigating the gaps between heterogeneous detectors. In [19], the classification and regression heads are added to a bare one-stage network in turn to illustrate the second stage can significantly affect the category prediction (Cate-Pred). Inspired by this, we further visualize the prediction results of heterogeneous detectors. As the location prediction (Loc-Pred) involves multiple dimensions, we represent it by IOU between the prediction result and the nearest ground truth bounding box (Bbox), which is referred as predicted IOU. Fig. 2 illustrates the remarkable gaps in Cate-Preds between heterogeneous detectors. Above the red line, the two-stage detector generates more predictions, while the one-stage detector shows dense predictions between the red and blue lines compared to the



(a) Teacher-motivated KD. (b) Student-motivated KD.

Figure 3. TKD vs. SKD. The backbone is omitted for simplicity. Red dashed line represents the KD loss, blue dashed line represents the pipeline to select the knowledge for distillation.

two-stage detector. This suggests that the two-stage detector achieves higher category confidence. Moreover, the Loc-Preds of heterogeneous detectors are similar.

Category-aware KD (CaKD). According to Fig. 2, to make one-stage detector get accurate predictions, we distill the Cate-Preds to narrow the gap between heterogeneous detectors. However, the Cate-Preds from one-stage detectors indicate the confidence of dense anchors, while those from two-stage detectors pertain to sparse sampled ROIs. Consequently, the crucial challenge lies in *how to align the Cate-Preds between heterogeneous detectors*.

Teacher-motivated KD. To tackle this problem, we can use the positions of ROIs in the two-stage teacher detector as queries to select the Cate-Preds of anchors at corresponding positions in the one-stage detector. Subsequently, the Cate-Preds form heterogeneous detectors are aligned for KD. We refer to this scheme as teacher-motivated KD (TKD), as shown in Fig. 3(a). However, this scheme is suboptimal, because the ROIs in the teacher network remain unchanged across different epochs, and thus the fixed-position Cate-Preds in the student detector are utilized to capture the teacher's knowledge in the training phase. Besides, the knowledge derived from teachers' ROI predictions may not fully meet the requirements of students.

Student-motivated KD. To flexibly capture knowledge from the teacher model, as shown in Fig. 3(b), we propose an adaptive student-motivated KD (SKD). In SKD, we first use NMS to select the Cate-Preds of representative samples (named to distinguish from ROIs, and abbreviated as RSs) from the student network. After that, we utilize the positions of these RSs to find the anchors (from the first stage of teacher) as teacher's ROIs, and the Cate-Preds of these ROIs from the second stage of teacher are used to supervise the selected student predictions. We formulate SKD as:

$$\mathcal{L}_{CaKD} = \sum_{i=1}^{N^{PC}} \frac{1}{N_i^{RS}} \sum_{j=1}^{N_i^{RS}} \left(\hat{p}_{i,j}^{s} - p_{i,j}^{t|s} \right)^2 \tag{1}$$

where N_i^{RS} represents the number of selected RSs in the i-th frame. $\hat{p}_{i,j}^{s}$ represents the Cate-Pred of the j-th selected RS from one-stage student detector, and $p_{i,j}^{t|s}$ represents the corresponding Cate-Pred from two-stage teacher detector.

Compared with other KD methods for LiDAR-3DOD [7, 50, 60], CaKD fully investigates the second stage knowledge for KD, without requiring feature consistency and avoiding the complexities of feature transfer. Moreover, CaKD elegantly tackles the foreground-background imbalance problem [9, 47, 50] in KD for object detection tasks, because it focuses on transferring knowledge related to the object areas through RS selection.

Furthermore, we attempt to elucidate the effectiveness of Eq. (1) from an alternative perspective. Eq. (1) can be viewed as a more potent localization quality estimation (LQE) [29, 58]. Unlike previous methods that use IOU or centerness information to rescore Cate-Preds [19, 21, 46, 48, 64], the LQE information in Eq. (1) is derived from the more robust two-stage teacher networks. Moreover, the evaluation information in Eq. (1) is also Cate-Pred (from teachers), which does not have semantic gaps.

3.3. Category-aware Pruning

A well-designed architecture is also crucial for achieving the high-performance student detectors, hence we introduce category-aware pruning method in our framework for student architecture exploration.

Previous KD methods for 2D detection use mature student networks (e.g., ResNet-18 [15]). However, there are few predefined students in LiDAR-3DOD. A few KD schemes [7, 50, 60] get the student network by reducing the width of each layer with the same ratio, which can be expressed as:

$$\begin{aligned} &\min_{\lambda_{i^k,j^k}} \sum_{k=1}^{3} \sum_{i^k=1}^{L^k} \sum_{j^k=1}^{N_{i^k}} \lambda_{i^k,j^k} \mathcal{M}(\mathbf{w}_{i^k,j^k}) \\ &s.t. \quad \frac{\sum_{j^k=1}^{N_{i^k}} \lambda_{i^k,j^k}}{N_{i^k}} = \eta^k, i^k = 1, ..., L^k, k = 1, 2, 3. \end{aligned} \tag{2}$$

where λ is an indicator. If the filter needs to be retained, then $\lambda = 0$, otherwise $\lambda = 1$. $\mathcal{M}$ represents the pruning metric for the filter, the larger it is, the more important the filter is, and the filter should be retained. L^k represents the number of layers of different modules. N_{i^k} represents the number of filers. η^k represents the pruning ratio of different modules. Eq. (2) exposes three issues of previous schemes [7, 50, 60]: **(1)** Each module is narrowed independently. However, all modules should be evaluated and pruned simultaneously, as they collaboratively generate predictions. **(2)** The same pruning ratio is used for each layer. However, the optimal architectures need to be explored, due to different number of critical filters are contained in each layer. **(3)** These methods reduce the width using empirical knowledge (i.e., $\mathcal{M}$ is a constant) and train from scratch to create a compact detector, leading the compact detector to overlook the crucial parameters of original pretrained detector.

To address these problems, we propose category-aware pruning (CaPr). We first reformulate Eq. (2) to:

$$\begin{aligned} &\min_{\lambda_{i,j}} \quad \sum_{i=1}^{L}\sum_{j=1}^{N_i} \lambda_{i,j}\mathcal{M}(\mathbf{w}_{i,j}) \\ &s.t. \quad \left(\sum_{i=1}^{L}\sum_{j=1}^{N_i}\lambda_{i,j}\right)\Big/\sum_{i=1}^{L} N_i = \eta \end{aligned} \tag{3}$$

In Eq. (3), filters from all modules, including 3D and 2D convolutions, are measured and pruned concurrently. This implies that the number of filters in each layer is determined by filters' importance ranking across all convolutions. Subsequently, we introduce a metric capable of evaluating both 2D and 3D convolutions simultaneously.

Measuring features for filter evaluation and pruning is an effective paradigm, as features encapsulate crucial information from both the filter and input data [20, 31, 57, 59]. The Cate-Pred is the vital feature of detectors (as stated in Section 3.2), making it an ideal information for filter evaluation. Considering that, regardless of the convolution type, a filter plays an important role in the network as long as it helps ensure accurate Cate-Preds. Consequently, we formulate the Cate-Pred gap before and after the removal of each filter to demonstrate the importance of the corresponding filter:

$$\mathcal{M}(\mathbf{w}_{i,j}) = \mathcal{M}({}_{/\mathbf{w}_{i,j}}p_n^s) = \sum_{n=1}^{N^{VAL}} \frac{1}{N^A}\left|p_n^s - {}_{/\mathbf{w}_{i,j}}p_n^s\right| \tag{4}$$

where N^A and N^{VAL} represent the number of anchors and the number of frames in validation set. ${}_{/\mathbf{w}_{i,j}}p_n^s$ represents the Cate-Pred of model after pruning the filter $\mathbf{w}_{i,j}$. If $\mathcal{M}(\mathbf{w}_{i,j})$ is larger, then $\mathbf{w}_{i,j}$ plays a crucial role for the Cate-Pred, making it pivotal in the model, and thus it should be retained. We substitute Eq. (4) into Eq. (3):

$$\begin{aligned} &\min_{\lambda_{i,j}} \quad \sum_{i=1}^{L}\sum_{j=1}^{N_i} \lambda_{i,j} \sum_{n=1}^{N^{VAL}} \frac{1}{N^A}\left|p_n^s - {}_{/\mathbf{w}_{i,j}}p_n^s\right| \\ &s.t. \quad \left(\sum_{i=1}^{L}\sum_{j=1}^{N_i}\lambda_{i,j}\right)\Big/\sum_{i=1}^{L} N_i = \eta \end{aligned} \tag{5}$$

Obviously, Eq. (5) can be minimized by pruning $\sum_{i=1}^{L}\sum_{j=1}^{N_i}\lambda_{i,j}$ filters with the lowest rankings in terms of Cate-Pred gaps. We summarize the pruning pipeline as:

Pruning pipeline. As shown in Fig. 4: (1) Start with the original pretrained detector containing all its filters $\mathcal{W}$, and its Cate-pred is p_n^s (Step '①'); (2) Then, individually remove each filter $\mathbf{w}_{i,j}$, and perform inference on the verification set to get the corresponding Cate-Pred ${}_{/\mathbf{w}_{i,j}}p_n^s$ (Step '②'); (3) Calculate the metric in Eq. (4) for each filter (Step '③'); (4) Sort the metric values in ascending order, and remove filters corresponding to the top η metric values (Step '④'); (5) Conduct Fine-tuning on the pruned detector to restore the performance and get the customizable compact student model (Step '⑤').

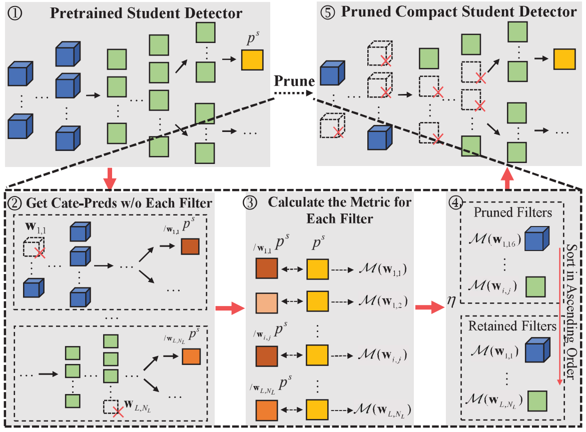
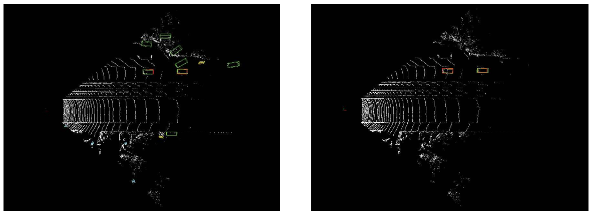

① Pretrained Student Detector
p^s
Prune
⑤ Pruned Compact Student Detector
② Get Cate-Preds w/o Each Filter
$\mathbf{w}_{1,1}$
${}_{/\mathbf{w}_{1,1}}p^s$
${}_{/\mathbf{w}_{L,N_L}}p^s$
$\mathbf{w}_{L,N_L}$
③ Calculate the Metric for Each Filter
${}_{/\mathbf{w}_{1,1}}p^s$
p^s
$\mathcal{M}(\mathbf{w}_{1,1})$
$\mathcal{M}(\mathbf{w}_{1,2})$
${}_{/\mathbf{w}_{i,j}}p^s$
$\mathcal{M}(\mathbf{w}_{i,j})$
${}_{/\mathbf{w}_{L,N_L}}p^s$
$\mathcal{M}(\mathbf{w}_{L,N_L})$
η
④ Pruned Filters
$\mathcal{M}(\mathbf{w}_{1,16})$
$\mathcal{M}(\mathbf{w}_{i,j})$
Retained Filters
$\mathcal{M}(\mathbf{w}_{1,1})$
$\mathcal{M}(\mathbf{w}_{L,N_L})$
Sort in Ascending Order

Figure 4. Pipeline of category-aware pruning.



(a) Predict w/o IOU.

(b) Predict with IOU.

Figure 5. Visualization of predictions before and after IOU-aware refinement. Red boxes represent the ground truth Bboxes, while the remaining boxes represent the predicted Bboxes. The predicted IOUs for objects in (b) are 0.75 (left) and 0.73 (right), respectively.

3.4. Modified IOU-aware Refinement

Even after distillation and careful investigation of the architecture, the student network still produces plenty of false positive (FP) samples, which impacts the detector accuracy (as shown in Fig. 5(a)). To address this issue, in our framework, we leverage an additional IOU head to help remove redundant FP samples (as depicted in Fig. 1(b)).

In training phase, the loss function for training IOU head can be expressed as:

$$\mathcal{L}_{IOU} = \sum_{i=1}^{N^{PC}} \frac{1}{N_i^{IOU+}} \sum_{j=1}^{N_i^s} \left(p_{i,j}^{IOU} - p_{i,j}^{IOU\_label}\right)^2 \tag{6}$$

where $p_{i,j}^{IOU}$ and $p_{i,j}^{IOU\_label}$ represent predicted IOU and IOU label, respectively. N_i^{IOU+} represents the number of samples with $p_{i,j}^{IOU\_label} > 0$. N_i^s represents the number of elements in IOU head.

In inference phase, we apply predicted IOU to generate mask $\mathbf{M}_{i,j}^{IOU}$ to remove FP samples (as shown in Fig. 1(b)). If $p_{i,j}^{IOU} > \delta$, then $\mathbf{M}_{i,j}^{IOU} = 1$, and the corresponding Cate-Pred participates in NMS; If $p_{i,j}^{IOU} \leq \delta$, then $\mathbf{M}_{i,j}^{IOU} = 0$, and the Cate-Pred does not participate in NMS, where δ is a small threshold. The final Cate-Pred used for NMS can be expressed as: $p*_i^s = \mathbf{M}_i^{IOU} \cdot p_i^s$.

Our method is orthogonal to previous IOU-aware re-

	Model	RR	KD	Car	Ped.	Cyc.	Para.	FLOPs	mAP
Stu	SECOND [49]	1.00	✘	81.33	52.95	65.89	5.3	80.7	66.72
	CenterPoint [54]	1.00	✘	78.50	51.85	67.78	5.8	96.5	66.04
Tea	Voxel-RCNN [10]	1.00	✘	84.97	57.74	73.73	11.0	81.6	72.15
Stu	SECOND [49]	1.00	✔	83.27	60.88	75.13	5.3	81.0	73.09
		0.75	✔	83.09	62.91	73.48	3.3	54.2	73.16
		0.50	✔	82.72	60.44	72.15	1.5	30.2	71.77
		0.30	✔	79.18	53.78	66.61	0.6	17.7	66.52
	CenterPoint [54]	1.00	✔	82.85	60.84	73.73	5.8	97.9	72.48
		0.75	✔	82.72	59.59	73.51	3.5	67.3	71.94
		0.50	✔	80.40	60.65	72.18	1.8	39.5	71.07
		0.35	✔	73.98	58.65	66.73	1.1	31.3	66.45
Tea	PV-RCNN [39]	1.00	✘	84.25	57.67	72.33	13.1	93.1	71.42
Stu	SECOND [49]	1.00	✔	83.01	58.61	73.82	5.3	81.0	71.81
		0.75	✔	82.69	59.92	71.99	3.3	54.2	71.53
		0.50	✔	82.40	60.12	71.17	1.5	30.2	71.23
		0.30	✔	79.46	52.96	65.86	0.6	17.7	66.10
	CenterPoint [54]	1.00	✔	83.40	57.56	70.89	5.8	97.9	70.62
		0.75	✔	83.32	58.10	72.82	3.5	67.3	71.41
		0.50	✔	80.53	59.11	70.25	1.8	39.5	69.96
		0.35	✔	74.06	58.12	67.08	1.1	31.3	66.42
Tea	PartA2 [40]	1.00	✘	82.22	60.42	72.65	63.6	93.3	71.77
Stu	SECOND [49]	1.00	✔	82.66	60.08	73.51	5.3	81.0	72.08
		0.75	✔	82.85	60.51	73.86	3.3	54.2	72.41
		0.50	✔	82.71	57.62	72.10	1.5	30.2	70.81
		0.30	✔	77.18	53.38	66.51	0.6	17.7	65.69
	CenterPoint [54]	1.00	✔	82.99	58.12	72.07	5.8	97.9	71.06
		0.75	✔	83.05	58.12	73.29	3.5	67.3	71.49
		0.50	✔	80.14	59.28	72.24	1.8	39.5	70.55
		0.35	✔	74.85	57.73	65.15	1.1	31.3	65.91

Table 1. Results of CaKDP on KITTI dataset. 'RR' represents retaining ratio. The moderate AP@R40 and moderate mAP@R40 are reported. The best result is marked in blue.

finement methods [19, 48, 64]: (1) Our method utilizes predicted IOU to remove redundant FP samples ('②' in Fig. 1(b)), rather than rescoring the entire Cate-Preds ('①' in Fig. 1(b)). (2) Our method leverages all samples for training, whereas previous methods [19, 48, 64] only train with samples having IOU greater than 0.

3.5. Final Loss of CaKDP Framework

The compression pipeline of CaKDP framework is that we first prune the student detector by CaPr (Section 3.3), and then restore the performance of pruned student detector by training with task loss, CaKD loss (Eq. (1)) and IOU-aware loss (Eq. (6)). The final loss can be formulated as:

$$\mathcal{L}_{CaKDP} = \mathcal{L}_{Task} + \alpha \cdot \mathcal{L}_{CaKD} + \beta \cdot \mathcal{L}_{IOU} \tag{7}$$

where $\mathcal{L}_{Task}$ represents the vanilla loss, involving classification loss and regression loss, for training detectors [6, 10, 54]. α and β represent the factors to modulate the influence of KD loss and IOU-aware loss.

4. Experiment

4.1. Experiment Setting

Datasets and Networks. We leverage various distillation pairs on different datasets to verify our proposed method. For KITTI dataset [12], several two-stage detectors, including Voxel-RCNN [10], PV-RCNN [39] and PartA2 with the sparse convolution based UNet [40], are used as teachers to assist the training of one-stage detectors (SECOND [49] and CenterPoint [54] with different detection head). Additionally, for large-scale Waymo Open Dataset (WOD) [43], CenterPoint with residual-based 3D backbone is used as student to capture the knowledge of Voxel-RCNN and PV-RCNN++ [41], respectively. Moreover, similar to [50], we also extract 20% of the training data and all validation set from WOD to generate WOD-mini for fast verification.

	Scheme	Config.	Car	Ped.	Cyc.	Para.	FLOPs	mAP
Stu	SECOND [49]	1.00	81.33	52.95	65.89	5.3	80.7	66.72
Tea	Voxel-RCNN [10]	1.00	84.97	57.75	73.73	11.0	81.6	72.15
Stu	+ Vanilla KD [16]	0.75	81.04	53.22	63.62	3.0	45.7	65.96
	+ GID [9]	0.75	81.61	53.04	67.62	3.0	45.7	67.43
	+ PD [60]	0.75	81.34	50.64	66.42	3.0	45.7	66.13
	+ SparseKD [50]	0.75	81.18	51.51	67.66	3.0	45.7	66.78
	+ CaKDP	0.50	**82.72**	**60.44**	**72.15**	**1.5**	**30.2**	**71.77**
Tea	PV-RCNN [39]	1.00	84.25	57.67	72.33	13.1	93.1	71.42
Stu	+ Vanilla KD [16]	0.75	81.60	50.19	64.30	3.0	45.7	65.36
	+ GID [9]	0.75	81.71	51.78	64.51	3.0	45.7	66.00
	+ PD [60]	0.75	81.64	47.81	67.19	3.0	45.7	65.55
	+ SparseKD [50]	0.75	81.64	50.76	66.76	3.0	45.7	66.39
	+ CaKDP	0.50	**82.40**	**60.12**	**71.17**	**1.5**	**30.2**	**71.23**
Tea	PartA2 [40]	1.00	82.22	60.42	72.65	63.6	93.3	71.77
Stu	+ Vanilla KD [16]	0.75	81.41	49.30	65.34	3.0	45.7	65.35
	+ GID [9]	0.75	81.84	53.55	67.16	3.0	45.7	67.52
	+ PD [60]	0.75	80.90	51.22	64.37	3.0	45.7	65.50
	+ SparseKD [50]	0.75	81.97	51.68	63.63	3.0	45.7	65.76
	+ CaKDP	0.50	**82.71**	**57.62**	**72.10**	**1.5**	**30.2**	**70.81**

Table 2. Comparison between CaKDP and previous KD methods on SECOND (trained on KITTI). 'Config.' is the width retaining ratio of each layer for other methods, while it represents the retaining ratio for CaKDP. The result of CaKDP is marked in bold.

Configurations. In all experiments, β is set to 1.0. When distilling knowledge to SECOND, the threshold of NMS is set to 0.7 to select RSs for KD, and the minimal Cate-Pred of selected RSs is set to 0.25. When training the lightweight CenterPoint on WOD, the threshold and minimal Cate-Pred are set to 0.7 and 0.1 in NMS to select RSs for KD, respectively. For each student detector, we set different η to compress the model, while the retaining ratio (1-η) is recorded in the subsequent tables. We keep other training and evaluation configurations in OpenPCDet [45] as default. All the experiments are deployed on 8 GeForce RTX 3090 or XP GPUs. More detailed configurations are demonstrated in the supplementary material.

4.2. Results on KITTI Dataset

We first demonstrate the effectiveness of our method on KITTI dataset. As shown in Table 1, CaKDP can achieve the accurate and compact student detectors. For example, when Voxel-RCNN is used to supervise SECOND, CaKDP reduces the storage requirement of detector by 8.8× without affecting its detection precision (SECOND-×0.3).

To further illustrate the effectiveness of CaKDP, we take SECOND as student to compare the results of CaKDP with those of other KD methods. As depicted in Table 2, CaKDP

	Model	RR	KD	Para.	FLOPs	L2 mAP	L2 mAPH
Stu	CenterPoint [54]	1.00	✗	7.8	114.8	66.32	63.88
Tea	PV-RCNN++ [41]	1.00	✗	16.1	123.5	70.07	67.67
Stu	CenterPoint [54]	1.00	✔	7.8	116.2	68.77	66.51
		0.70	✔	4.7	79.1	68.58	66.27
		0.50	✔	2.8	55.6	67.85	65.48
		0.35	✔	1.8	39.0	65.44	62.97
Tea	Voxel-RCNN [10]	1.00	✔	18.7	117.6	69.70	67.46
Stu	CenterPoint [54]	1.00	✔	7.8	116.2	68.78	66.51
		0.70	✔	4.7	79.1	68.67	66.36
		0.50	✔	2.8	55.6	67.67	65.32
		0.35	✔	1.8	39.0	65.33	62.88

Table 3. Results on WOD-mini. 'L2' represents 'LEVEL 2'.

	Scheme	Config.	Para.	FLOPs	L2 mAP	L2 mAPH
Stu	CenterPoint [54]	1.00	7.8	114.8	66.32	63.88
Tea	PV-RCNN++ [41]	1.00	16.1	123.5	70.07	67.67
Stu	+ Vanilla KD [16]	0.65	3.4	54.5	66.04	62.07
	+ GID [9]	0.65	3.4	54.5	66.04	62.57
	+ PD [60]	0.65	3.4	54.5	64.20	61.69
	+ SparseKD [50]	0.65	3.4	54.5	66.04	63.46
	+ CaKDP	0.50	**2.8**	**55.6**	**67.85**	**65.48**
Tea	Voxel-RCNN [41]	1.00	18.7	117.6	69.70	67.46
Stu	+ Vanilla KD [16]	0.65	3.4	54.5	64.91	62.38
	+ GID [9]	0.65	3.4	54.5	65.12	62.59
	+ PD [60]	0.65	3.4	54.5	64.10	61.56
	+ SparseKD [50]	0.65	3.4	54.5	66.29	63.71
	+ CaKDP	0.50	**2.8**	**55.6**	**67.67**	**65.32**

Table 4. Comparison between CaKDP and previous KD methods on CenterPoint (trained on WOD-mini).

surpasses its competitors by a large margin. Regarding the combination of "PV-RCNN & SECOND", our method yields an impressive mAP improvement exceeding 5% with less memory footprint. In summary, in contrast to other approaches, our method exhibits a more prominent enhancement in the performance of the student models.

4.3. Results on Waymo Open Dataset

Results and Comparisons on WOD-mini. We further deploy experiments on large-scale WOD. Firstly, WOD-mini is applied for fast verification, the results in Table 3 demonstrate that CaKDP can gain the high performance student detectors. Moreover, we compare CaKDP with other KD methods. As shown in Table 4, our proposed method still outperforms its opponents. For example, when PV-RCNN++ is used as teacher, CenterPoint obtained by CaKDP demonstrates higher mAP and mAPH with similar computational consumption and fewer parameters.

Results and Comparisons on WOD. Additionally, similar to [50], we conduct compression on full WOD, and compare the obtained compact model with other detectors. The results are shown in Table 5. For example, our CenterPoint‡† outperforms original CenterPoint by 1.61% with around 2.8× fewer parameters, 2.1× fewer FLOPs. Hence, our method can obtain efficient and accurate detectors on complicated dataset.

Model	Para.	FLOPs	L2 mAP	L2 mAPH
SECOND [49]	5.3	84.5	62.29	58.74
CenterPoint [54]	7.8	114.8	68.07	65.66
PV-RCNN [39]	13.1	118.5	67.06	63.74
PV-RCNN++ [41]	16.1	123.5	71.47	69.27
Voxel-RCNN [10]	16.8	103.3	70.15	67.90
CenterPoint + SparseKD [50]	4.0	47.8	-	65.75
CenterPoint + SparseKD [50]	2.8	36.9	-	64.83
CenterPoint† + CaKDP	**7.8**	**116.2**	**69.74**	**67.59**
CenterPoint‡ + CaKDP	**4.7**	**79.1**	**69.56**	**67.36**
CenterPoint‡† + CaKDP	**2.8**	**55.6**	**69.54**	**67.27**
CenterPoint‡‡ + CaKDP	**1.8**	**39.0**	**68.01**	**65.63**

Table 5. Results and comparison on full WOD. CenterPoint†, CenterPoint‡, CenterPoint‡† and CenterPoint‡‡ represent the pruned CenterPoint with retaining ratio equal to 1.00, 0.70, 0.50 and 0.35, respectively. PV-RCNN++ is used as teacher.

5. Ablation Study

5.1. Comparison of Different Modes of CaKD

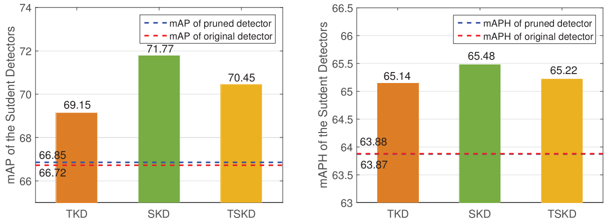
This subsection conducts various ablations to demonstrate the influence of different modes in CaKD, including **SKD**, **TKD**, and **TKD-SKD joint mode (TSKD)**. When training with TKD and TSKD, we only modify the distillation module while keeping other configurations unchanged. As shown in Fig. 6, compared with vanilla training strategy (red and blue lines), all three modes can significantly improve the mAP of student detectors. Besides, among the three schemes, SKD consistently provides the best compression results. Hence, CaKD proves effective in transferring knowledge between heterogeneous distillation pairs, and SKD can transfer the student-customized knowledge in each iteration to get the higher performance detectors.

5.2. Effectiveness of CaPr

To demonstrate the effectiveness of category-aware pruning (CaPr), we compare it with its various variants, including, **(1) Proportional pruning (ProPr)**: This approach removes filters in each layer of the network with the same proportion; **(2) Reverse CaPr (RV-CaPr)**: This mode uses Eq. (4) to evaluate the importance of each filter. Subsequently, it removes filters corresponding to the larger metric values; **(3) Random CaPr (RD-CaPr)**: Similar to CaPr, this scheme also removes the filters corresponding to smaller metric values, but it requires random initialization of the parameter in the compact model after pruning. It should be mentioned that all the pruning schemes are followed by a retraining step to restore the accuracy. As shown in Table 6, RV-CaPr yields the worst and CaPr the best detectors in different distillation pairs. Hence, the presented metric is useful to illustrate the importance of network architectures, and CaPr enables flexible generation of lightweight student detectors with appropriate architectures and initialization parameters.

5.3. Comparison of IOU-aware Refinement Modes

In this subsection, to demonstrate the effectiveness of our modified IOU-aware refinement approach, we compare it with previous schemes [19, 48, 64], in which the predicted



(a) mAP of SECOND-×0.5. (b) mAPH of CenterPoint-×0.5.

Figure 6. Comparison of different modes in CaKD. The red and blue lines represent the metric of original models (without pruning and KD loss) and pruned models (without KD loss), respectively.

Model	Scheme	Config.	Para.	FLOPs	Eva.
SECOND	-	1.00	5.3	80.7	66.72
on KITTI	ProPr	0.62	2.0	31.1	65.38
	RV-CaPr	0.50	1.5	30.1	56.62
	RD-CaPr	0.50	1.5	30.1	64.97
	CaPr (ours)	0.50	**1.5**	**30.1**	**66.24**
CenterPoint	-	1.00	7.8	114.8	63.88
on WOD-mini	ProPr	0.65	3.4	54.5	61.77
	RV-CaPr	0.50	2.8	54.2	57.09
	RD-CaPr	0.50	2.8	54.2	62.14
	CaPr (ours)	0.50	**2.8**	**54.2**	**63.61**

Table 6. Comparison of different pruning schemes. 'Eva.' indicates mAP@R40 for KITTI and L2 mAPH for WOD-mini.

KD Pair	w/o IOU head	Previous	Ours
Voxel-RCNN & SECOND on KITTI	71.06	67.31	**71.77**
PV-RCNN++ & CenterPoint on WOD-mini	64.78	62.43	**65.48**

Table 7. Comparison of different IOU-aware refinement modes. The retaining ratios are set to 0.5 for experiments on both KITTI and WOD-mini, with moderate mAP@R40 reported for KITTI and L2 mAPH reported for WOD-mini.

IOU is used to rescore the category predictions (confidence scores). We only modify the training strategy for IOU head and the IOU-aware refinement approach, while keeping other configurations unchanged. As demonstrated in Table 7, when compared with non-IOU head detector, the previous scheme significantly degrades network performance. This occurs because the KD module has introduced knowledge from the teacher to achieve more accurate Cate-Preds, and further refinement by relatively inaccurate predicted IOU can disrupt the revised Cate-Preds, resulting in poorer detection capabilities. Our approach improves the network performance, since it only filters out the redundant samples with infinitesimal IOU values while preserving the accurate Cate-Preds provided by KD. To sum up, we emphasize that our approach is orthogonal to the previous scheme [19, 48, 64], and it can further enhance the ability of compact detectors in our framework.

5.4. Influence of Each Component in CaKDP

CaKDP framework consists of three modules aimed at enhancing the accuracy of student detectors: CaKD, IOU-aware refinement, and CaPr. In this subsection, we provide examples to assess the influence of different modules. The results of individual module and their combinations are presented in Table 8. Compared with the raw training strategy (the 1-st row), the inclusion of any module can effectively improve the detector performance (the 2-nd to 4-th rows). Moreover, combining different modules in pairs (the 5-th to 7-th rows) results in more accurate compact detectors. Furthermore, when we aggregate all three modules together (the 8-th row), the best results are obtained.

KD Pair	CaKD	IOU	CaPr	Para.	FLOPs	Eva.
"Voxel-RCNN	✘	✘	✘	2.0	31.1	65.38
& SECOND"	✔	✘	✘	2.0	31.1	69.18
on KITTI	✘	✔	✘	2.0	31.2	66.02
	✘	✘	✔	1.5	30.1	66.24
	✔	✔	✘	2.0	31.2	69.82
	✔	✘	✔	1.5	30.1	71.06
	✘	✔	✔	1.5	30.2	66.85
	✔	✔	✔	1.5	30.2	**71.77**
"PV-RCNN++	✘	✘	✘	3.4	54.5	61.77
& CenterPoint"	✔	✘	✘	3.4	54.5	63.06
on WOD-mini	✘	✔	✘	3.5	55.9	62.01
	✘	✘	✔	2.8	54.2	63.61
	✔	✔	✘	3.5	55.9	64.03
	✔	✘	✔	2.8	54.2	64.78
	✘	✔	✔	2.8	55.6	63.87
	✔	✔	✔	2.8	55.6	**65.48**

Table 8. Influence of Each Component. When distilling without CaPr, ProPr (stated in Section 5.2) is conducted. The retaining ratios are same as those in Table 6.

6. Conclusion

We propose category-aware knowledge distillation and pruning (CaKDP) framework for compressing point cloud-based 3D detectors. In our framework, we first present category-aware knowledge distillation (CaKD), which enhances the compact detector performance by narrowing the category predictions (Cate-Preds) between heterogeneous distillation pairs. Additionally, to flexibly select optimal architectures and parameters for the compact student detector in KD, we introduce category-aware pruning (CaPr) to evaluate filters' importance by calculating Cate-Pred gaps and remove unimportant filters. Furthermore, a modified IOU-aware refinement module is employed to eliminate redundant predicted FP samples. Extensive experiments on various datasets demonstrate the effectiveness of CaKDP.

In our framework, CaKD is limited to conduct KD between detectors belonging to the same model family (e.g., CNN-based models). It does not generalize well to distill knowledge between transformer-based and CNN-based architectures. We plan to explore this issue in the future.

Acknowledgments

This work is supported by the Natural Science Foundation of China under Grant NSFC62088102, National Basic Strengthen Research Program of ReRAM under Grant 2022-00-03, Natural Science Foundation of Shaanxi Province, China under Grant 2022JM-366 and by the Open-End Fund of Beijing Institute of Control Engineering under Grant OBCandETL-2022-03.

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 multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1

[2] Defang Chen, Jian-Ping Mei, Yuan Zhang, Can Wang, Zhe Wang, Yan Feng, and Chun Chen. Cross-layer distillation with semantic calibration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7028–7036, 2021. 3

[3] Defang Chen, Jian-Ping Mei, Hailin Zhang, Can Wang, Yan Feng, and Chun Chen. Knowledge distillation with the reused teacher classifier. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11933–11942, 2022.

[4] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5008–5017, 2021. 1, 3

[5] Yixin Chen, Zhuotao Tian, Pengguang Chen, Shu Liu, and Jiaya Jia. Sea: Bridging the gap between one-and two-stage detector distillation via semantic-aware alignment. *arXiv preprint arXiv:2203.00862*, 2022. 3

[6] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, and Jiaya Jia. Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21674–21683, 2023. 2, 6

[7] Hyeon Cho, Junyong Choi, Geonwoo Baek, and Wonjun Hwang. itkd: Interchange transfer-based knowledge distillation for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13540–13549, 2023. 1, 2, 3, 4

[8] 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. 3

[9] Xing Dai, Zeren Jiang, Zhao Wu, Yiping Bao, Zhicheng Wang, Si Liu, and Erjin Zhou. General instance distillation for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7842–7851, 2021. 1, 3, 4, 6, 7

[10] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1201–1209, 2021. 2, 3, 6, 7

[11] Lue Fan, Xuan Xiong, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Rangedet: In defense of range view for lidar-based 3d object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2918–2927, 2021. 1

[12] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 1, 2, 6

[13] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015. 3

[14] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015. 1, 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1, 3, 6, 7

[17] Zejiang Hou, Minghai Qin, Fei Sun, Xiaolong Ma, Kun Yuan, Yi Xu, Yen-Kuang Chen, Rong Jin, Yuan Xie, and Sun-Yuan Kung. Chex: Channel exploration for cnn model compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12287–12298, 2022. 3

[18] Hengyuan Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. Network trimming: A data-driven neuron pruning approach towards efficient deep architectures. *arXiv preprint arXiv:1607.03250*, 2016. 3

[19] Yihan Hu, Zhuangzhuang Ding, Runzhou Ge, Wenxin Shao, Li Huang, Kun Li, and Qiang Liu. Afdetv2: Rethinking the necessity of the second stage for object detection from point clouds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 969–979, 2022. 2, 3, 4, 6, 7, 8

[20] Yaomin Huang, Ning Liu, Zhengping Che, Zhiyuan Xu, Chaomin Shen, Yaxin Peng, Guixu Zhang, Xinmei Liu, Feifei Feng, and Jian Tang. Cp3: Channel pruning plug-in for point-based networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5302–5312, 2023. 1, 3, 5

[21] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 784–799, 2018. 4

[22] 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. 1

[23] Linh Kästner, Vlad Catalin Frasineanu, and Jens Lambrecht. A 3d-deep-learning-based augmented reality calibration method for robotic environments using depth sensor data. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1135–1141. IEEE, 2020. 1

[24] Taehyeon Kim, Jaehoon Oh, NakYil Kim, Sangwook Cho, and Se-Young Yun. Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation. *arXiv preprint arXiv:2105.08919*, 2021. 3

[25] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12697–12705, 2019. 2

[26] Yejin Lee, Donghyun Lee, JungUk Hong, Jae W Lee, and Hongil Yoon. Not all neighbors matter: Point distribution-aware pruning for 3d point cloud. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1240–1249, 2023. 1, 2, 3

[27] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016. 1, 3

[28] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in Neural Information Processing Systems*, 33:21002–21012, 2020. 2

[29] Xiang Li, Wenhai Wang, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11632–11641, 2021. 4

[30] Yawei Li, Kamil Adamczewski, Wen Li, Shuhang Gu, Radu Timofte, and Luc Van Gool. Revisiting random channel pruning for neural network compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 191–201, 2022. 3

[31] Mingbao Lin, Rongrong Ji, Yan Wang, Yichen Zhang, Baochang Zhang, Yonghong Tian, and Ling Shao. Hrank: Filter pruning using high-rank feature map. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1529–1538, 2020. 1, 3, 5

[32] Xin Lu, Quanquan Li, Buyu Li, and Junjie Yan. Mimicdet: Bridging the gap between one-stage and two-stage object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 541–557. Springer, 2020. 3

[33] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3967–3976, 2019. 1, 3

[34] Matthew Pitropov, Chengjie Huang, Vahdat Abdelzad, Krzysztof Czarnecki, and Steven Waslander. Lidar-mimo: Efficient uncertainty estimation for lidar-based 3d object detection. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pages 813–820. IEEE, 2022. 1

[35] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 2

[36] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 2

[37] 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

[38] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 770–779, 2019. 2

[39] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10529–10538, 2020. 2, 3, 6, 7

[40] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *IEEE transactions on pattern analysis and machine intelligence*, 43(8):2647–2664, 2020. 3, 6

[41] Shaoshuai Shi, Li Jiang, Jiajun Deng, Zhe Wang, Chaoxu Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. *International Journal of Computer Vision*, 131(2):531–551, 2023. 2, 3, 6, 7

[42] Yang Sui, Miao Yin, Yi Xie, Huy Phan, Saman Aliari Zonouz, and Bo Yuan. Chip: Channel independence-based pruning for compact neural networks. *Advances in Neural Information Processing Systems*, 34:24604–24616, 2021. 3

[43] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 1, 2, 6

[44] Yehui Tang, Yunhe Wang, Yixing Xu, Yiping Deng, Chao Xu, Dacheng Tao, and Chang Xu. Manifold regularized dynamic network pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5018–5028, 2021. 1

[45] OpenPCDet Development Team. Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet, 2020. 6

[46] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 4

[47] 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. 1, 4

[48] Shengkai Wu, Xiaoping Li, and Xinggang Wang. Iou-aware single-stage object detector for accurate localization. *Image and Vision Computing*, 97:103911, 2020. 2, 4, 6, 7, 8

[49] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 2, 3, 6, 7

[50] 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. 1, 2, 3, 4, 6, 7

[51] Maosheng Ye, Shuangjie Xu, and Tongyi Cao. Hvnet: Hybrid voxel network for lidar based 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1631–1640, 2020. 1

[52] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4133–4141, 2017. 1, 3

[53] Junbo Yin, Dingfu Zhou, Liangjun Zhang, Jin Fang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. Proposalcontrast: Unsupervised pre-training for lidar-based 3d object detection. In *European Conference on Computer Vision*, pages 17–33. Springer, 2022. 1

[54] 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. 2, 3, 6, 7

[55] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3903–3911, 2020. 3

[56] Jia Zeng, Li Chen, Hanming Deng, Lewei Lu, Junchi Yan, Yu Qiao, and Hongyang Li. Distilling focal knowledge from imperfect expert for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 992–1001, 2023. 1

[57] Haonan Zhang, Longjun Liu, Hengyi Zhou, Wenxuan Hou, Hongbin Sun, and Nanning Zheng. Akecp: Adaptive knowledge extraction from feature maps for fast and efficient channel pruning. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 648–657, 2021. 3, 5

[58] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sunderhauf. Varifocalnet: An iou-aware dense object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8514–8523, 2021. 4

[59] Haonan Zhang, Longjun Liu, Bingyao Kang, and Nanning Zheng. Hierarchical model compression via shape-edge representation of feature maps—an enlightenment from the primate visual system. *IEEE Transactions on Multimedia*, 2022. 3, 5

[60] 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. 1, 2, 3, 4, 6, 7

[61] Yuyao Zhang and Nikolaos M. Freris. Adaptive filter pruning via sensitivity feedback. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2023. 3

[62] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 11953–11962, 2022. 1, 3

[63] Chenglong Zhao, Bingbing Ni, Jian Zhang, Qiwei Zhao, Wenjun Zhang, and Qi Tian. Variational convolutional neural network pruning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2780–2789, 2019. 1

[64] Wu Zheng, Weiliang Tang, Sijin Chen, Li Jiang, and Chi-Wing Fu. Cia-ssd: Confident iou-aware single-stage object detector from point cloud. In *Proceedings of the AAAI conference on artificial intelligence*, pages 3555–3562, 2021. 2, 4, 6, 7, 8

[65] 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. 3

[66] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018. 2