

**Abstract**

Recent advances in 3D object detection (3DOD) have obtained remarkably strong results for LiDAR-based models. In contrast, surround-view 3DOD models based on multiple camera images underperform due to the necessary view transformation of features from perspective view (PV) to a 3D world representation which is ambiguous due to missing depth information. This paper introduces X$^3$KD, a comprehensive knowledge distillation framework across different modalities, tasks, and stages for multi-camera 3DOD. Specifically, we propose cross-task distillation from an instance segmentation teacher (X-IS) in the PV feature extraction stage providing supervision without ambiguous error backpropagation through the view transformation. After the transformation, we apply cross-modal feature distillation (X-FD) and adversarial training (X-AT) to improve the 3D world representation of multi-camera features through the information contained in a LiDAR-based 3DOD teacher. Finally, we also employ this teacher for cross-modal output distillation (X-OD), providing dense supervision at the prediction stage. We perform extensive ablations of knowledge distillation at different stages of multi-camera 3DOD. Our final X$^3$KD model outperforms previous state-of-the-art approaches on the nuScenes and Waymo datasets and generalizes to RADAR-based 3DOD. Qualitative results video at https://youtu.be/1do9DPFmr38.

**1. Introduction**

3D object detection (3DOD) is an essential task in various real-world computer vision applications, especially autonomous driving. Current 3DOD approaches can be categorized by their utilized input modalities, e.g., camera images [28,40,46] or LiDAR point clouds [25,55,60], which dictates the necessary sensor suite during inference. Recently, there has been significant interest in surround-view

---

*These authors contributed equally to this work.
†Automated Driving, Qualcomm Technologies, Inc.
‡Qualcomm AI Research, an initiative of Qualcomm Technologies, Inc.
§Automated Driving, QT Technologies Ireland Limited
Table 1. Analysis of BEVDepth† (re-implementation of [28]): We compare the architectural improvement of a larger Lift-Splat-Shoot (LSS++) transform to using depth supervision (DS).

<table>
<thead>
<tr>
<th>Model</th>
<th>LSS++</th>
<th>DS</th>
<th>GFLOPS</th>
<th>mAP↑</th>
<th>NDS↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEVDepth†</td>
<td>✗</td>
<td>✗</td>
<td>298</td>
<td>32.4</td>
<td>44.9</td>
</tr>
<tr>
<td></td>
<td>✗</td>
<td>✓</td>
<td>298</td>
<td>33.1</td>
<td>44.9</td>
</tr>
<tr>
<td></td>
<td>✔</td>
<td>✓</td>
<td>316</td>
<td>34.9</td>
<td>47.0</td>
</tr>
<tr>
<td></td>
<td>✔</td>
<td>✔</td>
<td>316</td>
<td>35.9</td>
<td>47.2</td>
</tr>
<tr>
<td>X³KD (Ours)</td>
<td>✓</td>
<td>✓</td>
<td>316</td>
<td>39.0</td>
<td>50.5</td>
</tr>
</tbody>
</table>

During training but not during inference. The recently introduced BEVDepth [28] approach pioneers using accurate 3D information from LiDAR data at training time to improve multi-camera 3DOD, see Fig. 1 (top part). Specifically, it proposed an improved Lift-Splat-Shoot PV-to-BEV transform (LSS++) and depth supervision (DS) by projected LiDAR points, which we analyze in Table 1. We observe that the LSS++ architecture yields significant improvements, though depth supervision seems to have less effect. This motivates us to find additional types of supervision to transfer accurate 3D information from LiDAR point clouds to multi-camera 3DOD. To this end, we propose cross-modal knowledge distillation (KD) to not only use LiDAR data but a high-performing LiDAR-based 3DOD model, as in Fig. 1 (middle part). To provide an overview of the effectiveness of cross-modal KD at various multi-camera 3DOD network stages, we present three distillation techniques: feature distillation (X-FD) and adversarial training (X-AT) to improve the feature representation by the intermediate information contained in the LiDAR 3DOD model as well as output distillation (X-OD) to enhance output-stage supervision.

For optimal camera-based 3DOD, extracting useful PV features before the view transformation to BEV is equally essential. However, gradient-based optimization through an ambiguous view transformation can induce non-optimal supervision signals. Recent work proposes pre-training the PV feature extractor on instance segmentation to improve the extracted features [49]. Nevertheless, neural networks are subject to catastrophic forgetting [23] such that knowledge from pre-training will continuously degrade if not retained by supervision. Therefore, we propose cross-task instance segmentation distillation (X-IS) from a pre-trained instance segmentation teacher into a multi-camera 3DOD model, see Fig. 1 (bottom part). As shown in Table 1, our X³KD framework significantly improves upon BEVDepth without additional complexity during inference.

To summarize, our main contributions are as follows:
- We propose X³KD, a KD framework across modalities, tasks, and stages for multi-camera 3DOD.
- Specifically, we introduce cross-modal KD from a strong LiDAR-based 3DOD teacher to the multi-camera 3DOD student, which is applied at multiple network stages in bird’s eye view, i.e., feature-stage (X-FD and X-AT) and output-stage (X-OD).
- Further, we present cross-task instance segmentation distillation (X-IS) at the PV feature extraction stage.
- X³KD outperforms previous approaches for multi-camera 3DOD on the nuScenes and Waymo datasets.
- We transfer X³KD to RADAR-based 3DOD and train X³KD only through KD without using ground truth.
- Our extensive ablation studies on nuScenes and Waymo provide a comprehensive evaluation of KD at different network stages for multi-camera 3DOD.

2. Related Work

Multi-View Camera-Based 3D Object Detection: Current multi-view 3D object detectors can be divided into two main streams: First, DETR3D and succeeding works [30, 32, 33, 46, 59] project a sparse set of learnable 3D queries/priors onto 2D image features with subsequent sampling and an end-to-end 3D bounding box regression. Second, LSS and following works [2, 18, 28, 40] employ a view transformation consisting of a depth prediction, a point cloud reconstruction, and a voxel pooling to project points to BEV. 3D bounding boxes are predicted from these BEV features. While such works focus on improving the network architecture and view transformation, we focus on better model optimization. In this direction, M³BEV [49] proposed segmentation [3, 4] pre-training of the PV feature extraction. We propose cross-task instance segmentation distillation to retain this knowledge during 3DOD training.

Most current state-of-the-art works focus on incorporating temporal information either through different kinds of feature-level aggregation [17, 28, 30, 33] or by improving depth estimation by temporal stereo approaches [27, 47]. While the usual setting considers data from 2 time steps, recently proposed SOLOFusion [38] separately models long-range and short-range temporal dependencies in input data from 16 time steps. Our work focuses on a different direction, i.e., we try to optimally exploit the information contained in LiDAR point clouds. In this direction, BEVDepth [28] and succeeding works [27, 38] supervise the depth estimation with projected LiDAR points. We explore this path further by using cross-modal knowledge distillation (KD) from a LiDAR-based 3DOD teacher.

Multi-Modal 3D Object Detection: Recently, there has been a trend to fuse different sensor modalities, especially camera and LiDAR, with the idea of combining modality-specific useful information, hence improving the final 3DOD performance [1, 22, 29, 35, 50, 53]. Existing 3DOD methods mostly perform multi-modal fusion at one of the three stages: First, various approaches [44, 45, 50] propose to decorate/augment the raw LiDAR points with image features. Second, intermediate feature fusion of the modalities in a shared representation space, such as the BEV space, has been explored [8, 22, 29, 35, 53]. Third,
knowledge. The current superiority of LiDAR-based 3DOD over multi-camera 3DOD can be attributed to the ambiguous view transformation in multi-camera models, which may feature features at the wrong position in the final representation (e.g., a BEV grid). Meanwhile, LiDAR-based models operate on a 3D point cloud, which can easily be projected onto any view representation. Thereby, the extracted features preserve 3D information. Our cross-modal KD com-
We present $X^3$KD, a knowledge distillation (KD) framework for multi-camera 3DOD. We employ an inference setup (middle blue box) relying only on multi-camera image input (LiDAR point cloud in the output is just shown for visualization). During training, we apply KD across several network stages (red arrows originating from the blue box): In perspective-view (PV) feature extraction, we apply cross-task instance segmentation distillation (X-IS) from an instance segmentation teacher (yellow box). In the bird’s eye view (BEV), we apply cross-modal feature distillation (X-FD), adversarial training (X-AT), and output distillation (X-OD) from a LiDAR-based 3DOD teacher (green box). $X^3$KD significantly enhances the multi-camera 3DOD without inducing extra complexity during inference.

LiDAR-based 3DOD Model Architecture: Our LiDAR-based 3DOD model is mainly based on Center-Point [55]. First, the point cloud $I \in \mathbb{R}^{P \times 5}$ is processed by the Sparse Encoder from SECOND [51], yielding 3D sparse features $f^3D \in \mathbb{R}^{H_{BEV} \times W_{BEV} \times D_3 \times C_3D}$ with volumetric extent $H_{BEV} \times W_{BEV} \times D_3$ and number of channels $C_{3D}$. Then, the features are projected onto the same BEV plane as in the camera-based 3DOD model, yielding BEV features $f_{BEV} \in \mathbb{R}^{H_{BEV} \times W_{BEV} \times C_{BEV}}$ with $C_{BEV} = D_3 \times C_{3D}$. These are further processed by an encoder-decoder network, yielding refined BEV features $f_{REF} \in \mathbb{R}^{H_{BEV} \times W_{BEV} \times C_{REF}}$. Finally, the features are passed through a prediction head, yielding probability score maps $\hat{b}^{cls} \in \mathbb{R}^{H_{BEV} \times W_{BEV} \times |S|}$ and regression maps $\hat{b}^{reg} \in \mathbb{R}^{H_{BEV} \times W_{BEV} \times 9}$ analogous to the outputs $b^{cls}$ and $b^{reg}$ of the multi-camera 3DOD model.

Output-stage Distillation (X-OD): Following many approaches in KD [6, 11, 15, 54, 57], we distill knowledge at the output stage by imposing losses between the teacher’s outputs $\hat{b}^{cls}$ and $\hat{b}^{reg}$ and the student’s outputs $\hat{b}^{cls}$ and $\hat{b}^{reg}$.

Specifically, we impose a Gaussian focal loss $L_{GFocal}$ [26] between $\hat{b}^{cls}$ and $\hat{b}^{cls}$ to put more weight on rare classes and compensate for the class imbalance. As this loss only considers pseudo labels as a positive sample if they are exactly 1, we select high-confidence teacher output probabilities $\hat{b}^{cls}$, i.e., probability values over a threshold $\alpha_{3D-box}$, and set them to 1. Further, the regression output of the student $\hat{b}^{reg}$ is supervised by the corresponding output $\hat{b}^{reg}$ of the teacher by imposing a Smooth L1 loss $L_{SmoothL1}$ [12]. Finally, we propose to weigh the regression loss by the teacher’s pixel-wise averaged output probabilities $\langle \hat{b}^{cls} \rangle = \frac{1}{|S|} \sum_{s \in S} \hat{b}^{cls} \in \mathbb{R}^{H_{BEV} \times W_{BEV}}$ to weigh regions which likely contain objects higher than the background. Overall, X-OD is defined as:

$$L_{X-OD}(\hat{b}, \hat{b}) = L_{GFocal}(\hat{b}^{cls}, \hat{b}^{cls}) + L_{SmoothL1}(\hat{b}^{reg}, \hat{b}^{reg})$$ (3)

Feature-stage Distillation (X-FD): Our X-FD compo-
to preserve the knowledge contained in the PV features con-
trophic forgetting such that this initial knowledge is not nec-
training. However, deep neural networks exhibit catas-
ting on the refined features \( \hat{\tilde{f}} \) from both modalities directly after the
both modalities in BEV space. Due to the structural dis-
ther propose X-AT to encourage a more global feature sim-
\[ L_{\text{X-FD}} = L(\hat{h}, \hat{\tilde{h}}) \]  

Feature-stage Adversarial Training (X-AT): We further propose X-AT to encourage a more global feature sim-
both modalities directly after the BEV projection (Fig. 3), we apply the adversarial training on the refined features \( \hat{f} \) and \( \tilde{f} \). We pass these cross-modal features through a gradient reversal layer and a
branch. As output, we obtain \( N^{\text{IS}} \) bounding boxes \( \hat{y} = \{(\hat{y}^{\text{bbox}}_n, \hat{y}^{\text{cls}}_n, \hat{y}^{\text{core}}_n), n \in \{1, \ldots, N^{\text{IS}}\}\} \) with four parame-
ters for bounding box center and spatial extent \( \hat{y}^{\text{bbox}}_n \in \mathbb{R}^4 \), a classification result \( \hat{y}^{\text{cls}}_n \in \mathcal{S}^{\text{IS}} \) from the set of IS classes \( \mathcal{S}^{\text{IS}} \), and an objectness score \( \hat{y}^{\text{core}}_n \in \mathbb{I} \) with \( I = [0, 1] \). Addition-
ally, we obtain corresponding object masks \( \hat{m} = \{\hat{m}_n, n \in \{1, \ldots, N^{\text{IS}}\}\} \) with single masks \( \hat{m}_n \in \{0, 1\}^{H_{\text{mask}} \times W_{\text{mask}}} \) and spatial resolution \( H_{\text{mask}} \times W_{\text{mask}} \). We select all samples with a score \( \hat{y}^{\text{score}}_n > \theta_{2D-bbox} \) as pseudo labels.

X-IS Loss Computation: The teacher-generated pseudo labels are used to supervise an additional PV instance seg-
head, cf. Fig. 2 (bottom right), which uses the same
same and ROI head architectures as the teacher. The
head outputs region proposals \( \hat{a} = (\hat{a}^{\text{cls}}, \hat{a}^{\text{reg}}) \) with foreground/background scores \( \hat{a}^{\text{cls}} \in \mathbb{I}^{H_{\text{PV}} \times W_{\text{PV}} \times 2K} \) and regression parameters \( \hat{a}^{\text{reg}} \in \mathbb{R}^{H_{\text{PV}} \times W_{\text{PV}} \times 4K} \) relative to each of the \( K \) anchors. Our RPN loss \( L_{\text{RPN}} \) is then com-
comprised of an assignment strategy between pseudo GT and
head outputs as detailed in [42] and subsequent application of BCE and L1 differences for optimizing \( \hat{a}^{\text{cls}} \) and \( \hat{a}^{\text{reg}} \), respectively. The \( \mathcal{X}^{\text{RPN}} \) region proposals with the
high differences are subsequently passed through the
ROI head, which outputs refined bounding boxes \( \hat{y} = \{(\hat{y}^{\text{bbox}}_n, \hat{y}^{\text{cls}}_n), n \in \{1, \ldots, \mathcal{X}^{\text{RPN}}\}\} \) class probabilities \( \hat{y}^{\text{cls}}_n \in \mathbb{I}^{\mathcal{S}^{\text{IS}}} \), four bounding box regression parameters \( \hat{y}^{\text{bbox}}_n \in \mathbb{R}^4 \) as well as class-specific mask probabilities \( \hat{m}_n = \{\hat{m}_n, n \in \{1, \ldots, \mathcal{X}^{\text{RPN}}\}\} \) with single masks \( \hat{m}_n \in \{0, 1\}^{H_{\text{mask}} \times W_{\text{mask}}} \). Our bounding box loss \( L_{\text{bbox}} \) is comprised of an assignment strategy between ground truth
and prediction \( \hat{y} \) and subsequent application of L1 difference between \( \hat{y}^{\text{bbox}}_n \) and \( \hat{y}^{\text{bbox}}_n \) as well as cross-entropy (CE) difference between \( \hat{y}^{\text{cls}}_n \) and one-hot encoded \( \hat{y}^{\text{cls}}_n \). For comput-
ing the mask loss \( L_{\text{mask}} \), we apply a binary cross entropy (BCE) difference between ground truth \( \hat{m} \) and prediction \( \hat{m} \), selecting only the output corresponding to the
ground truth mask’s class. More details can be found in [14]. Over-
all, our X-IS loss \( L_{\text{X-IS}} \) can be written as:

\[ L_{\text{X-IS}} = L_{\text{RPN}}(\hat{a}, \hat{y}) + L_{\text{bbox}}(\hat{y}, \hat{y}) + L_{\text{mask}}(\hat{m}, \hat{m}) \] 

4. Experiments

We first provide our experimental setup (Sec. 4.1) and a state-of-the-art comparison (Sec. 4.2). Next, we verify and analyze our method’s components in Secs. 4.3 and 4.4. Last, we evaluate RADAR-based models (Sec. 4.5).
Table 2. Performance comparison on the nuScenes dataset: We ensure comparability regarding backbone and image resolution. Baseline results are cited except for BEVDepth† which we reproduced in our framework; † indicates recent ArXiv works; best numbers in boldface.

<table>
<thead>
<tr>
<th>Model</th>
<th>Backbone</th>
<th>Resolution</th>
<th>mATE↓</th>
<th>mASE↓</th>
<th>mAOE↓</th>
<th>mAVE↓</th>
<th>mAAE↓</th>
<th>mAP↑</th>
<th>NDS↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEVDet [18]</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.725</td>
<td>0.279</td>
<td>0.589</td>
<td>0.860</td>
<td>0.245</td>
<td>29.8</td>
<td>37.9</td>
</tr>
<tr>
<td>BEVDet4D [49]</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.703</td>
<td>0.278</td>
<td>0.495</td>
<td>0.354</td>
<td>0.206</td>
<td>32.2</td>
<td>45.7</td>
</tr>
<tr>
<td>BEVDepth [28]</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.601</td>
<td>0.275</td>
<td>0.450</td>
<td>0.446</td>
<td>0.212</td>
<td>37.7</td>
<td>48.9</td>
</tr>
<tr>
<td>ResNet-101</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.710</td>
<td>0.270</td>
<td>0.438</td>
<td>0.367</td>
<td>0.190</td>
<td>37.2</td>
<td>50.0</td>
</tr>
<tr>
<td>ResNet-101</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.710</td>
<td>0.270</td>
<td>0.438</td>
<td>0.367</td>
<td>0.190</td>
<td>37.2</td>
<td>50.0</td>
</tr>
<tr>
<td>ResNet-50</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.716</td>
<td>0.274</td>
<td>0.372</td>
<td>0.394</td>
<td>0.198</td>
<td>41.6</td>
<td>51.7</td>
</tr>
<tr>
<td>ResNet-50</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.631</td>
<td>0.257</td>
<td>0.405</td>
<td>0.435</td>
<td>0.143</td>
<td>44.5</td>
<td>53.5</td>
</tr>
<tr>
<td>ResNet-50</td>
<td>ResNet-50</td>
<td>256 × 704</td>
<td>0.589</td>
<td>0.270</td>
<td>0.438</td>
<td>0.367</td>
<td>0.190</td>
<td>37.2</td>
<td>50.0</td>
</tr>
</tbody>
</table>

Table 3. Performance comparison on the Waymo dataset. We compare X^3KD to our re-implemented baseline BEVDepth† [28].

4.1. Experimental Setup

X^3KD is implemented using mmdetection3d [10] and PyTorch [39] libraries and trained on 4 NVIDIA A100 GPUs. Here, we describe our main setup on nuScenes while more details are provided in the supplementary.

Datasets: Similar to most recent works [1, 27, 28, 30, 55], we evaluate on the nuScenes and Waymo benchmark datasets. The nuScenes dataset [5] contains 28K, 6K, and 6K samples for training, validation, and test, respectively. We use data from a LiDAR sensor and 6 cameras with bounding box annotations for 10 classes. For the Waymo dataset [43], we use the data from a LiDAR sensor and 5 cameras with annotations for cars, pedestrians, and cyclists. It provides 230K annotated frames from 798, 202, and 150 sequences for training, validation, and test, respectively.

Evaluation Metrics: For nuScenes, we employ the officially defined mAP and NDS metrics. The NDS metric considers mAP as well as true positive (TP) metrics \( TP = \{ mATE, mASE, mAOE, mAVE, mAAE \} \) for translation, scale, orientation, velocity, and attribute, respectively, i.e., \( NDS = \frac{1}{5} (5 \cdot mAP) + \sum_{TP} (1 - min(1, TP)) \). For Waymo, we employ the official metrics of the camera-only 3D object detection track [19]: The \( LET-3D-AP \) calculates average precision after longitudinal error correction, while \( LET-3D-APL \) also penalizes the longitudinal error.

Network Architecture and Training: For a fair comparison, our network architecture follows previous works [17, 21, 27, 28, 30, 47]. We consider the ResNet-50-based setting with a resolution of 256 × 704 and the ResNet-101-based setting with resolutions of 512 × 1408 or 640 × 1600. Further network design choices are adopted from [17]. We train all models for 24 epochs using the CBGS training strategy [62], a batch size of 16 and AdamW [36] with an initial learning rate of \( 2 \cdot 10^{-4} \). The loss weights are set to \( \lambda_{GT} = 1, \lambda_{X-FD} = 10, \lambda_{X-AF} = 10, \lambda_{X-OD} = 1 \) and \( \lambda_{X-IS} = 1 \) while the thresholds are set to \( \alpha_{3D-box} = 0.6 \) and \( \alpha_{2D-box} = 0.2 \). Our LiDAR teacher is based on the CenterPoint architecture [55] and the TransFusion training schedule [1]. The supplementary contains further explanations, hyperparameter studies, and configurations for the Waymo dataset.

4.2. State-of-the-art Comparisons

We perform a comparison of X^3KD with all contributions, i.e., X^3KD-all, to other SOTA methods in Table 2. In the ResNet-50-based setting, our model achieves the best results with scores of 39.0 and 50.5 in mAP and NDS, respectively. In the high-resolution ResNet-101-based setting, our model achieves SOTA scores of 46.1 and 56.7. At this resolution, we outperform all previous SOTA methods in all considered metrics and outperform the second best result by 2.9 points in mAP and 2.5 points in NDS. To explicitly show that our method improves on top of current SOTA baselines, we retrain our strongest baseline among
published works, i.e., BEVDeph [28], in our code framework, dubbed BEVDeph†. At all resolutions, we are able to closely reproduce the reported results and improve by about 3 points in both mAP and NDS upon them. On the test set, we outperform the second best approach PolarFormer [21] by 1.8 points in terms of the main NDS metric and achieve best results in 5 out of 7 metrics. We also show results for BEVDeph† and X3KD variants on the Waymo dataset in Table 3. As on nuScenes, our X3KD_all model clearly outperforms the baseline in all metrics.

### 4.3. Method Ablation Studies

#### Effectiveness of the Proposed Components

We incrementally add our contributions in Table 4 and evaluate them in terms of NDS and mAP. First, we individually add X-OD, X-FD, and X-AT. For all three components, there is an improvement in the NDS metric, while the mAP metric remains similar or slightly worse. Adding all three components (X3KD_modal) gives a clear improvement over the baseline as well as applying each component individually. Particularly, we observe that the additional cross-modal supervision improves bounding box velocity estimation from multi-camera input as can be seen by the apparent improvement in the mAVE metric. Using X-IS, surprisingly gives an even more substantial improvement. This might indicate that supervision in BEV cannot completely compensate for the lack of rich features in PV. Finally, adding all components together to our proposed X3KD_all model clearly outperforms all other variants in terms of the main NDS and mAP metrics and is best in 4 out of 7 metrics in Table 4.

<table>
<thead>
<tr>
<th>Model</th>
<th>X-OD</th>
<th>X-FD</th>
<th>X-AT</th>
<th>X-IS</th>
<th>mATE↓</th>
<th>mASE↓</th>
<th>mAOE↓</th>
<th>mAVE↓</th>
<th>mAAE↓</th>
<th>mAP↑</th>
<th>NDS↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEVDeph†</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>0.636</td>
<td>0.272</td>
<td>0.493</td>
<td>0.499</td>
<td>0.198</td>
<td>35.9</td>
<td>47.2</td>
</tr>
<tr>
<td>X-OD</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>0.642</td>
<td>0.278</td>
<td>0.456</td>
<td>0.338</td>
<td>0.188</td>
<td>35.7</td>
<td>48.7</td>
</tr>
<tr>
<td>X-FD</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>0.644</td>
<td>0.276</td>
<td>0.479</td>
<td>0.361</td>
<td>0.200</td>
<td>36.1</td>
<td>48.5</td>
</tr>
<tr>
<td>X-AT</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>0.648</td>
<td>0.277</td>
<td>0.492</td>
<td>0.354</td>
<td>0.192</td>
<td>35.5</td>
<td>48.1</td>
</tr>
<tr>
<td>X3KD_modal</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>0.632</td>
<td>0.271</td>
<td>0.456</td>
<td>0.342</td>
<td>0.203</td>
<td>36.8</td>
<td>49.4</td>
</tr>
<tr>
<td>X-IS</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
<td>0.635</td>
<td>0.273</td>
<td>0.462</td>
<td>0.350</td>
<td>0.204</td>
<td>38.7</td>
<td>50.1</td>
</tr>
<tr>
<td>X3KD_all</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.615</td>
<td>0.269</td>
<td>0.471</td>
<td>0.345</td>
<td>0.203</td>
<td>39.0</td>
<td>50.5</td>
</tr>
</tbody>
</table>

#### Cross-Modal Output Distillation (X-OD)

We provide insights into our X-OD design in Table 5. In the top part, we observe that models trained with output distillation improve over the baseline in terms of NDS and that the confidence-based weighting is particularly effective for orientation (mAOE) and velocity (mAVE) prediction. Further, we train the multi-camera 3DOD without using annotations (Table 5, bottom part) solely from KD. In this setting, the weighting yields even more significant improvements in particular in terms of the NDS metric. Also, the X-OD_w/o GT model surprisingly outperforms the model variants trained with annotations in terms of the mAP metric. This promising result indicates that future work might be able to use large-scale pre-training with KD on unlabelled data for further performance improvements.

<table>
<thead>
<tr>
<th>Model</th>
<th>Dist.</th>
<th>Weight</th>
<th>w/o GT</th>
<th>mAOE↓</th>
<th>mAVE↓</th>
<th>mAP↑</th>
<th>NDS↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEVDeph†</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
<td>0.493</td>
<td>0.499</td>
<td>35.9</td>
<td>47.2</td>
</tr>
<tr>
<td>X-OD</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
<td>0.477</td>
<td>0.342</td>
<td>35.6</td>
<td>48.5</td>
</tr>
<tr>
<td>X-OD_w/o GT</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.456</td>
<td>0.338</td>
<td>35.7</td>
<td>48.7</td>
</tr>
<tr>
<td>X3KD_modal</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>1.090</td>
<td>0.972</td>
<td>36.1</td>
<td>35.3</td>
</tr>
<tr>
<td>X3KD_all</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.724</td>
<td>0.570</td>
<td>36.5</td>
<td>43.7</td>
</tr>
</tbody>
</table>

#### Cross-task Instance Segmentation Distillation (X-IS)

Ablations on our X-IS design are shown in Table 6. We observe that initialization of the backbone with weights from a pre-trained instance segmentation as well as cross-task distillation, improves the baseline’s result. Combining both aspects to X-IS yields the best result in both mAP and NDS. Using a different teacher model based on ConvNeXt-T yields similarly good results and shows that the feature extraction architectures of the instance segmentation teacher and the multi-camera 3DOD student do not need to match. Also, knowledge can be distilled from a simple ResNet-50-based model into a more sophisticated architecture such as ConvNeXt-T (bottom part of Table 6). Overall, cross-task distillation in terms of the main NDS and mAP metrics and is best in 4 out of 7 metrics.
Figure 4. **Qualitative results on nuScenes**: We show the multi-camera input (top) and bounding box visualizations (bottom). We compare ResNet-101-based X³KD_all to BEVDepth† and the ground truth (GT) for a resolution of 640 × 1600. Best viewed on screen and in color.

<table>
<thead>
<tr>
<th>Model</th>
<th>RADAR Input</th>
<th>Cam. Input</th>
<th>Validation mAP</th>
<th>Validation NDS</th>
<th>Test mAP</th>
<th>Test NDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RADAR only</td>
<td>✓</td>
<td>✗</td>
<td>12.9</td>
<td>13.0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>X³KD modal</td>
<td>✓</td>
<td>✗</td>
<td>17.7</td>
<td>23.5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Fusion only</td>
<td>✓</td>
<td>✓</td>
<td>38.9</td>
<td>51.0</td>
<td>40.2</td>
<td>52.3</td>
</tr>
<tr>
<td>X³KD modal</td>
<td>✓</td>
<td>✓</td>
<td>42.3</td>
<td>53.8</td>
<td>44.1</td>
<td>55.3</td>
</tr>
</tbody>
</table>

Table 7. **Generalization of our method to RADAR**: We distill knowledge from a LiDAR-based 3DOD into a RADAR-based and a RADAR-camera fusion-based 3DOD model. For RADAR-based models, we report the mAP just for the car class as these models underperform on other classes due to the point cloud sparsity.

4.4. Method Analysis

**Performance-Complexity Trade-off**: We analyze our method’s efficiency compared to state-of-the-art methods [17, 28, 30] in Fig. 5. We compare to reimplementations of BEVDepth [28] and BEVDet4D [17] as well as reported results of BEVFormer [30]. All reported models are ResNet-50-based or ResNet-101-based to ensure that a better trade-off cannot be attributed to a more efficient backbone. We observe that X³KD (red curve) outperforms BEVDepth (blue curve) at equal complexity due to the improved supervision from KD. Also, compared to BEVDet4D and BEVFormer a better trade-off can be observed, likely because of the absence of LiDAR supervision in BEVDet4D and the complex Transformer model in BEVFormer. Accordingly, our results show that X³KD achieves a better complexity-performance trade-off than current state-of-the-art methods.

**Qualitative Results**: We further show qualitative results of X³KD and BEVDepth in Fig. 4. As highlighted by the white boxes, X³KD detects and places objects more accurately in the scene. In particular, the recognition of objects and the prediction of their orientation shows improved characteristics in the X³KD output, which is coherent with a better quantitative performance of X³KD in Table 4. Further qualitative results are given in the supplementary.

**4.5. Generalization to RADAR**

We also generalize X³KD to RADAR-based and camera-RADAR fusion-based models. For RADAR-based models, we cannot apply cross-task KD from the instance segmentation teacher. Hence, we only use the cross-modal KD contributions, *i.e.*, X³KD modal. Our results on the nuScenes validation set show that X³KD modal significantly enhances the performance in both settings. Notably, the transfer from camera to RADAR was straightforward as we achieved the reported improvements without requiring tuning of hyperparameters. Further, we evaluate our fusion-based X³KD modal model on the nuScenes test set, where we outperform all other Camera-RADAR, fusion-based models, hence setting the state-of-the-art result.

5. Conclusions

We proposed X³KD, a KD framework for multi-camera 3DOD. By distilling across tasks from an instance segmentation teacher and across modalities from a LiDAR-based 3DOD teacher into a RADAR-based and a RADAR-camera fusion-based 3DOD student, we show that the model performance can be enhanced without inducing additional complexity during inference. We evaluated X³KD on the nuScenes and Waymo datasets, outperforming previous approaches by 2.9% mAP and 2.5% NDS. The transferability to other sensors, such as RADAR, and the possibility to train 3DOD models without annotations further demonstrate X³KD’s effectiveness. Combining these two findings could be used in future applications to train 3DOD models for arbitrary sensors, requiring only a LiDAR-based 3DOD model.
References


[28] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. BEVDepth:
Acquisition of Reliable Depth for Multi-view 3D Object Detection. In *Proc. of AAAI*, pages 1–9, 2023. 1, 2, 3, 6, 7, 8


