Supplementary Material of KECOR: Kernel Coding Rate Maximization for Active 3D Object Detection

1. Implementation Details

Following the same setting in [2], the batch sizes for training and evaluation are fixed to 6 and 16 on both KITTI and Waymo Open datasets. The Adam optimizer is adopted with a learning rate initiated as 0.01, and scheduled by one cycle scheduler. The number of MC-DROPOUT stochastic passes is set to 5.

Active Learning Protocols. For all experiments, we first randomly select m fully labeled point clouds from the training set as the initial $\mathcal{D}_L$. With the annotated data, the 3D detector is trained with E epochs, which is then freezed to select n candidates from $\mathcal{D}_U$ for label acquisition. We set the m and n to $2.5 \sim 3\%$ point clouds (*i.e.*, $n = m = 100$ for KITTI, $n = m = 400$ for Waymo Open) to trade-off between reliable model training and high computational costs. The aforementioned training and selection steps will alternate for R rounds. Empirically, we set $E = 30$, $R = 6$ for KITTI, and fix $E = 40$, $R = 5$ for Waymo Open. All 3D detection experiments are conducted on a GPU cluster with three V100 GPUs and the runs on the VOC07 dataset are conducted on a server with two NVIDIA GeForce RTX 2080 Ti. The runtime for an active learning experiment on KITTI and Waymo is around 11 hours and 65 hours, respectively. Note that, training PV-RCNN on the full set typically requires 40 GPU hours for KITTI and 800 GPU hours for Waymo.

2. Additional Results on the KITTI Dataset

In this section, we provide an additional study on the BEV mAP scores on the KITTI dataset across different difficulty levels. The detector backbone is set to PV-RCNN for all AL approaches. The results of the compared AL baselines and the proposed KECOR are plotted in Figure 1. A similar trend is observed to the one shown in Figure 2 in the main body. The proposed KECOR demonstrates a higher performance boost over the state-of-the-art CRB and BAIT at the moderate and hard levels.

3. Performance of K_{RBF} on the KITTI Dataset

To study the performance of the non-linear K_{RBF}, we conducted a series of experiments on the KITTI dataset, with both one-stage and two-stage detectors. The experimental results are shown in Figure 2, where the top row is with SECOND and the bottom row is with PV-RCNN, respectively. It can be observed that the Laplace RBF kernel performs better than the linear kernel with SECOND, yet very similar results with PV-RCNN. It implies that the one-stage detectors may have a simpler architecture, thus needing the non-linear kernel to help capture the non-linear relationship among the features. However, the performance of KECOR equipped with RBF kernel is still inferior to K_{Last} and K_{NTK}, which evidence that the empirical NTK kernel can capture not only the non-linear relationship between the inputs and outputs, but also measure the aleatoric uncertainty, thus helping detectors to identify more challenging objects.

Table 1: Comparisons with the weakly- and self-supervised 3D object detection approaches on KITTI.

	ANNO.	EASY	MOD.	HARD
SSL [1]	1% bbox	66.47	51.42	44.63
SSL [1]	20% bbox	88.65	79.52	74.87
Weakly [3]	3,712 weak+534 bbox	87.57	77.62	**76.94**
KECOR	1% (800) bbox	**91.71**	**79.56**	74.05

4. Comparison with Weakly- and Self- supervised Approaches

Weakly- [3] and self-supervised [1] approaches **complement** our pooled-based AL method by providing additional supervision for samples from $\mathcal{D}_U$. However, this inevitably increases computational costs for the next training round. As shown in Table 1, we summarized the performance comparison of self- and weakly-supervised methods *w.r.t* AP_{Car} on the KITTI val set in the right table. "weak" refers to fully annotated BEV maps. Note that Kecor achieves **high precision** with limited labeled data.

5. Impact of Kernels on Waymo Open

In addition to the ablation study on KITTI, we also run experiments on the Waymo Open dataset to examine the impact of kernels. The plots are illustrated in Figure 3. Similar

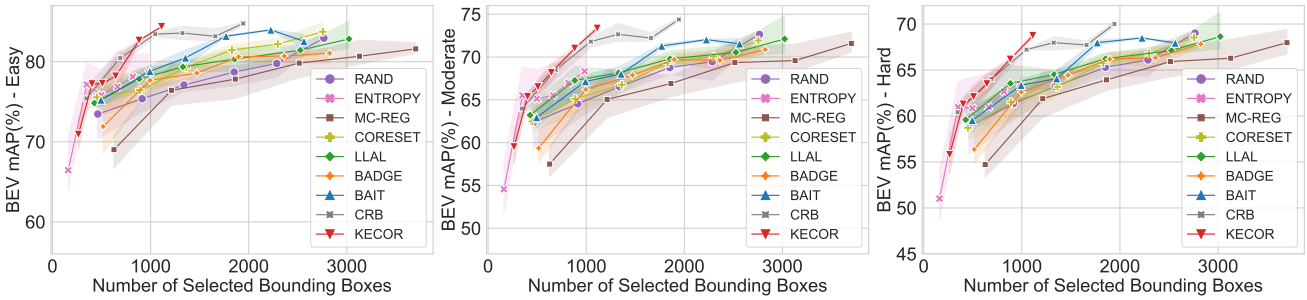

Figure 1: 3D mAP (%) of KECOR and AL baselines on the KITTI *val* split with PV-RCNN.

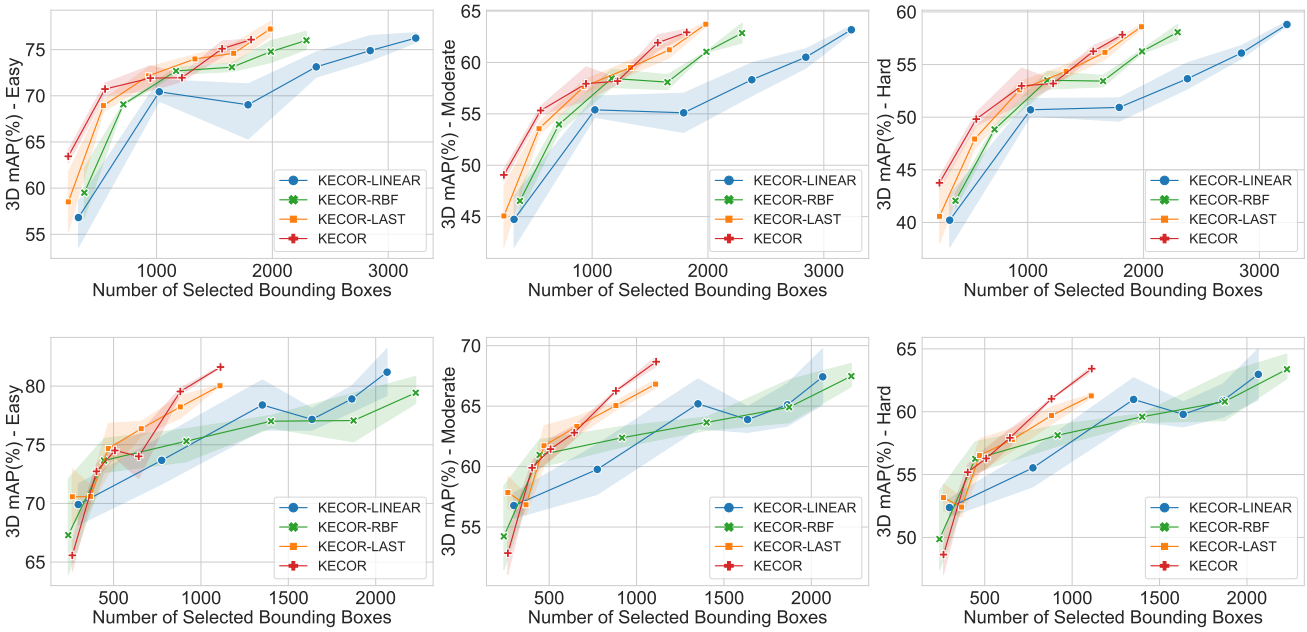

Figure 2: Ablation study on the different choices of kernels on the KITTI *val* split with SECOND (Top row) and PV-RCNN (Bottom row) across a variety of difficulty levels.

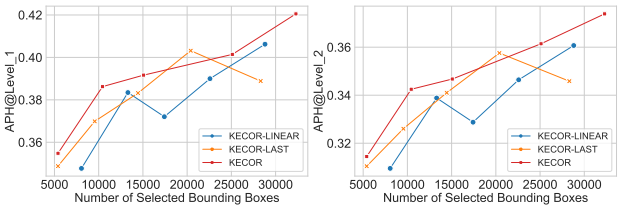

Figure 3: Ablation study on the different choices of kernels on the Waymo Open dataset with PV-RCNN.

to what we observed in KITTI, the KECOR and KECOR-LAST achieve better performance on both APH at different difficulty levels. However, we also notice that the KECOR-LINEAR does not select too many bounding boxes while it selects 2 times more bounding boxes on the KITTI dataset when reaching the same performance. We reason it is because, in Waymo datasets, most frames of point clouds are densely labeled and there are other irrelevant objects (*e.g.*, signs) that may trigger high entropy scores. Hence, to trade-off between the information and annotation costs, KECOR tends to prefer the point clouds having more information, yielding a slightly higher number of bounding boxes to annotate. How to lower the annotation costs on Waymo will leave an open question in future work.

6. Impact of σ_{ent} on Waymo Open

To study the impact of coefficient σ_{ent} on the Waymo Open dataset, we depict the results in the last round with

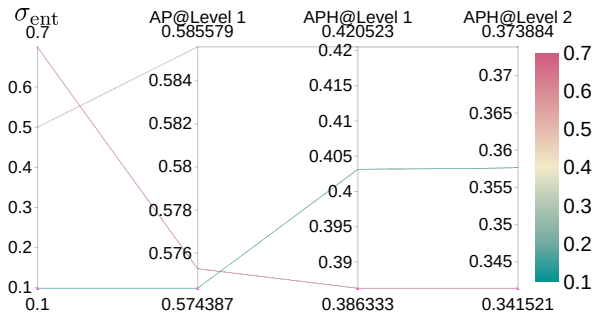

Figure 4: The parallel plot of the impact of σ_{ent} on Waymo.

regard to different evaluation metrics in Figure 4. We run three trials with the values of σ_{ent} varying in $\{0.1, 0.5, 0.7\}$ considering the high computational costs. The variant of KECOR with the $\sigma_{\text{ent}} = 0.7$ achieves the lowest performance. We infer this performance drop is caused by the dominance of the classification entropy regularization term. To trade-off between the high volume of information by kernel coding rate maximization and the lower costs of box annotation by classification entropy regularization, we select 0.5 as the value of σ_{ent} for the rest of the experiments on the Waymo Open dataset.

References

[1] Emeç Erçelik et al. 3d object detection with a self-supervised lidar scene flow backbone. In *ECCV*, 2022. 1

[2] Yadan Luo, Zhuoxiao Chen, Zijian Wang, Xin Yu, Zi Huang, and Mahsa Baktashmotlagh. Exploring active 3d object detection from a generalization perspective. In *Proc. International Conference on Learning Representations (ICLR)*, 2023. 1

[3] Qinghao Meng et al. Weakly supervised 3d object detection from lidar point cloud. In *ECCV*, 2020. 1