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

Yadan Luo[†]   Zhuoxiao Chen[†]   Zhen Fang[‡]   Zheng Zhang[♯]   Zi Huang[†]   Mahsa Baktashmotlagh[†]

[†]The University of Queensland   [‡]University of Technology Sydney   [♯]Harbin Institute of Technology, Shenzhen

{y.luo, zhuoxiao.chen, helen.huang, m.baktashmotlagh}@uq.edu.au,
zhen.fang@uts.edu.au, darrenzz219@gmail.com

Abstract

Achieving a reliable LiDAR-based object detector in autonomous driving is paramount, but its success hinges on obtaining large amounts of precise 3D annotations. Active learning (AL) seeks to mitigate the annotation burden through algorithms that use fewer labels and can attain performance comparable to fully supervised learning. Although AL has shown promise, current approaches prioritize the selection of unlabeled point clouds with high uncertainty and/or diversity, leading to the selection of more instances for labeling and reduced computational efficiency. In this paper, we resort to a novel kernel coding rate maximization (KECOR) strategy which aims to identify the most informative point clouds to acquire labels through the lens of information theory. Greedy search is applied to seek desired point clouds that can maximize the minimal number of bits required to encode the latent features. To determine the uniqueness and informativeness of the selected samples from the model perspective, we construct a proxy network of the 3D detector head and compute the outer product of Jacobians from all proxy layers to form the empirical neural tangent kernel (NTK) matrix. To accommodate both one-stage (i.e., SECOND) and two-stage detectors (i.e., PV-RCNN), we further incorporate the classification entropy maximization and well trade-off between detection performance and the total number of bounding boxes selected for annotation. Extensive experiments conducted on two 3D benchmarks and a 2D detection dataset evidence the superiority and versatility of the proposed approach. Our results show that approximately 44% box-level annotation costs and 26% computational time are reduced compared to the state-of-the-art AL method, without compromising detection performance.

1. Introduction

Being a crucial component in the realm of scene understanding, LiDAR-based 3D object detection [34, 58, 59, 63] identifies and accurately localizes objects in a 3D scene with the oriented bounding boxes and semantic labels. This technology has facilitated a wide range of applications in environmental perceptions, including robotics, autonomous driving, and augmented reality. With the recent advancements in 3D detection models [14, 25, 53], highly accurate recognition of objects can be achieved through point cloud projection [64], point feature extraction [34, 57, 59, 66, 67] or voxelization [13, 58, 63]. However, achieving such performance often comes at the expense of requiring a large volume of labeled point cloud data, which can be costly and time-consuming.

To mitigate the labeling costs and optimize the value of annotations, active learning (AL) [37, 49] has emerged as a promising solution. Active learning involves iteratively selecting the most beneficial samples for label acquisition from a large pool of unlabeled data until the labeling budget is exhausted. This selection process is guided by the selection criteria based on *sample uncertainty* [29, 36, 48, 50] and/or *diversity* [16, 22, 55, 65]. Both measures are used to assess the ***informativeness*** of the unlabeled samples. Aleatoric uncertainty-driven approaches search for samples that the model is least confident of by using metrics like maximum entropy [62] or estimated model changes [44, 68]. On the other hand, epistemic uncertainty based methods attempt to find the most representative samples to avoid sample redundancy by using greedy coreset algorithms [55] or clustering based approaches [5].

While active learning has proven to be effective in reducing labeling costs for recognition tasks, its application in LiDAR-based object detection has been limited [18, 30, 54]. This is largely due to its high computational costs and involvement of both detection and regression tasks, which pose significant challenges to the design of the selection criteria. A very recent work CRB [41] manually designed three heuristics that allow the acquisition of labels by hierarchically filtering out concise, representative, and geometrically balanced unlabelled point clouds. While effective, it remains unclear how to characterize the sample informativeness for both classification and regression tasks with *one unified measurement*.

In this paper, we propose a novel AL strategy called ker-

nel coding rate maximization (KECOR) for efficient and effective active 3D detection. To endow the model with the ability to reason about the trade-off between information and performance autonomously, we resort to the coding rate theory and modify the formula from feature selection to sample selection, by replacing the covariance estimate with the empirical neural tangent kernel (NTK). The proposed KECOR strategy allows us to pick the most informative point clouds from the unlabeled pool such that their latent features require the maximal coding length for encoding. To characterize the non-linear relationships between the latent features and the corresponding box predictions spending the least computational costs, we train a proxy network of the 3D detector head with labeled samples and extract the outer product of Jacobians from all proxy layers to form the NTK matrix of all unlabeled samples. Empirical studies evidence that the NTK kernel not only captures non-linearity but takes the aleatoric and epistemic uncertainties into joint consideration, assisting detectors to recognize challenging objects that are of sparse structure. To accommodate both one-stage (*i.e.*, SECOND) and two-stage detectors (*i.e.*, PV-RCNN), we further incorporate the classification entropy maximization into the selection criteria. Our contributions are summarized as below:

1. We propose a novel information-theoretic based criterion KECOR for cost-effective 3D box annotations that allows for the greedy search of informative point clouds by maximizing the kernel coding rate.

2. Our framework is flexible to accommodate different choices of kernels and 3D detector architectures. Empirical NTK kernel used in KECOR demonstrates a strong capacity to unify both aleatoric and epistemic uncertainties from the model perspective, which helps detectors learn a variety of challenging objects.

3. Extensive experiments have been conducted on both 3D benchmarks (*i.e.*, KITTI and Waymo Open) and 2D object detection dataset (*i.e.*, PASCAL VOC07), verifying the effectiveness and versatility of the proposed approach. Experimental results show that the proposed approach achieves a 44.4% reduction of annotations and up to 26.4% less running time compared to the state-of-the-art active 3D detection methods.

2. Related Work

2.1. Active Learning (AL)

Active learning has been widely applied to image classification and regression tasks, where the samples that lead to unconfident predictions (*i.e., aleatoric uncertainty*) [7, 15, 19, 29, 32, 36, 48, 50, 51, 56, 61, 61, 68] or not resemble the training set (*i.e., epistemic uncertainty*) [2, 16, 22, 24, 44, 45, 55, 65] will be selected and acquired for annotations. Hybrid methods [5, 11, 26, 31, 33, 40, 43] unify both types of uncertainty to form an acquisition criterion. Examples

include BADGE [5] and BAIT [4], which select a batch of samples that probably induce large and diverse changes to the model based on the gradients and Fisher information.

AL for Detection. Research on active learning for object detection [9, 23, 30, 52, 69, 71] has not bee as widespread as that for image classification, due in part to the challenges in quantifying aleatoric and epistemic uncertainties in bounding box regression. Kao *et al.* [30] proposed two metrics to quantitatively evaluate the localization uncertainty, where the samples containing inconsistent box predictions will be selected. Choi *et al.* [9] predicted the parameters of the Gaussian mixture model and computes epistemic uncertainty as the variance of Gaussian models. Agarwal *et al.* [1] proposed a contextual diversity measurement for selecting unlabeled images containing objects in diverse backgrounds. Park *et al.* [47] determined epistemic uncertainty by using evidential deep learning along with hierarchical uncertainty aggregation to effectively capture the context within an image. Wu *et al.* [62] introduced a hybrid approach, which utilizes an entropy-based non-maximum suppression to estimate uncertainty and a diverse prototype strategy to ensure diversity. Nevertheless, the application of active learning to 3D point cloud detection is still under-explored due to the high computational costs, which makes AL strategies such as adding additional detection heads [9] or augmentations [69] impractical. Previous solutions [6, 18, 20, 54] rely on generic metrics such as Shannon entropy [61], localization tightness [30] for measuring aleatoric uncertainty. Only a recent work CRB [41] exploited both uncertainties jointly by greedily searching point clouds that have concise labels, representative features and geometric balance. Different from CRB that hierarchically filters samples with three criteria, in this work, we derive an informatic-theoretic criterion, namely kernel coding rate, that enables informativeness measurement in a single step and saves computational costs by 26%.

2.2. Coding Rate

Entropy, rate distortion [8, 12] and coding rate [42] are commonly used measurements to quantify the uncertainty and compactness of a random variable $\mathbf{Z}$. They can be interpreted as the "goodness" of the latent representations in deep neural networks with respect to generalizability [39], transferability [27] and robustness. Entropy $\mathcal{H}(\mathbf{Z})$ calculates the expected value of the negative logarithm of the probability, while it is not well-defined for a continuous random variable with degenerate distributions [70]. To address this, rate distortion $\mathfrak{R}(\mathbf{Z}, \epsilon)$ is proposed in the context of lossy data compression, which quantifies the minimum average number of binary bits required to represent $\mathbf{Z}$. Given the calculation difficulty of distortion rate, coding rate $\mathfrak{R}(\mathbf{Z})$ emerges as a more feasible solution for quantifying random variables from a complex distribution (refer to Section 3.2). Unlike prior works mentioned above, our
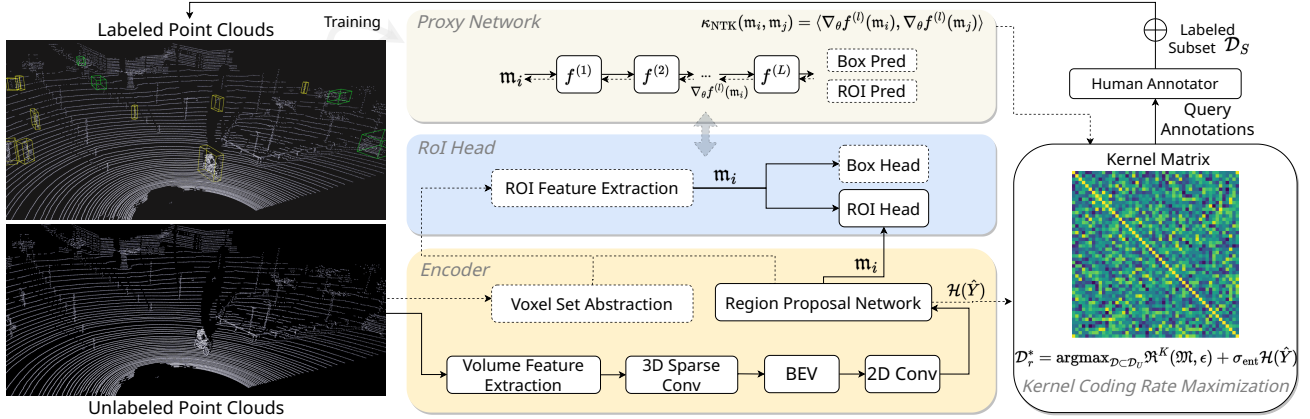
Figure 1: An illustration of the workflow of the proposed kernel coding rate maximization for active 3D detection. Dotted boxes indicate the unique components in two-stage 3D detectors (*e.g.*, PV-RCNN), while solid boxes indicate the shared components in both one-stage (*e.g.*, SECOND) and two-stage detectors.

work explores a new kernel coding rate for sample selection in active learning rather than feature selection.

2.3. Neural Tangent Kernel

Neural tangent kernel (NTK) [3, 28, 35, 46] is a kernel that reveals the connections between infinitely wide neural networks trained by gradient descent and kernel methods. NTK enables the study of neural networks using theoretical tools from the perspective of kernel methods. There have been several studies that have explored the properties of NTK: Jacot *et al.* [28] proposed the concept of NTK and showed that it could be used to explain the generalization of neural networks. Lee *et al.* [35] expanded on this work and demonstrated that the dynamics of training wide but finite-width NNs with gradient descent can be approximated by a linear model obtained from the first-order Taylor expansion of that network around its initialization. In this paper, rather than exploring the interpretability of infinite-width neural networks, we explore empirical (*i.e.*, finite-width) neural tangent kernels to improve linear kernels and non-linear RBF kernels. The NTK is used to characterize the sample similarity based on 3D detector head behaviors, which naturally takes aleatoric and epistemic uncertainties into consideration.

3. Preliminaries

In this section, we present the mathematical formulation of the problem of active learning for 3D object detection, along with the establishment of the necessary notations.

3.1. Problem Formulation

3D Object Detection. The typical approach for detecting objects in an orderless point cloud $\mathcal{P}_i$ involves training a 3D object detector to identify and locate the objects of interest, consisting of a set of 3D bounding boxes and their labels $\mathfrak{B}_i = \{b_k, y_k\}_{k \in [N_i]}$, with N_i indicating the number of bounding boxes in the i-th point cloud. Each

point in $\mathcal{P}_i = \{(x, y, z, r)\}$ is represented by xyz spatial coordinates and additional features such as reflectance r. The box annotations $b_k \in \mathbb{R}^7$ include the relative center xyz spatial coordinates to the object ground planes, the box size, the heading angle, and the box label $y_k \in \mathbb{R}^C$, where C indicates the number of classes. As illustrated in Figure 1, modern 3D detectors extract latent features $\mathfrak{m}_i = g(\mathcal{P}_i; \theta_g) \in \mathbb{R}^d$ through projection [64], PointNet encoding [34,57,59,66,67] or voxelization [13,58,63], where dimension $d = W \times H \times F$ is the product of width W, length H, and channels F of the feature map. The detection head $h(\cdot; \theta_h)$ uses $\mathfrak{m}_i$ as inputs and generates detection outcomes $\hat{\mathfrak{B}}_i = \{\hat{b}_k, \hat{y}_k\}$:

$$\mathcal{P}_i \xmapsto{g(\cdot; \theta_g)} \mathfrak{m}_i \xmapsto{h(\cdot; \theta_h)} \hat{\mathfrak{B}}_i. \tag{1}$$

Active Learning Setup. In an active learning setup, a small set of labeled point clouds $\mathcal{D}_L = \{\mathcal{P}_i, \mathfrak{B}_i\}_{i \in L}$ and a large pool of raw point clouds $\mathcal{D}_U = \{\mathcal{P}_j\}_{j \in U}$ are given at training time, where L and U are the index sets corresponding to $\mathcal{D}_L$ and $\mathcal{D}_U$, respectively, and the cardinality of each set satisfy that $|L| \ll |U|$. During each active learning round $r \in \{1, \ldots, R\}$, a subset of point clouds $\mathcal{D}_r^*$ is selected from $\mathcal{D}_U$ based on a defined active learning policy. The labels of 3D bounding boxes for the chosen point clouds are queried from an oracle $\Omega : \mathcal{P} \mapsto \mathfrak{B}$ to create a labeled set $\mathcal{D}_S = \{\mathcal{P}_j, \mathfrak{B}_j\}_{\mathcal{P}_j \in \mathcal{D}_r^*}$. The 3D detection model is pretrained with $\mathcal{D}_L$ for active selection and then retrained with $\mathcal{D}_S \cup \mathcal{D}_L$. The process is repeated until the selected samples reach the final budget B, *i.e.*, $\sum_{r=1}^{R} |\mathcal{D}_r^*| = B$.

3.2. Coding Rate

As explained in Section 2.2, information theory [12] defines the coding rate $\mathfrak{R}(\cdot, \epsilon)$ [42] as a measure of lossy data compression, quantifying the achievability of maximum compression while adhering to a desired error upper bound. It is commonly used as an empirical estimation of rate distortion [8, 12] indicating the minimal number of bi-

nary bits required to represent random variable $\mathbf{Z}$ with the expected decoding error below ϵ. Given a finite set of n samples $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n] \in \mathbb{R}^{d \times n}$, the coding rate [42] with respect to $\mathbf{Z}$ and a distortion ϵ is given by:

$$\Re(\mathbf{Z}, \epsilon) = \frac{1}{2} \log \det(\mathbf{I} + \frac{d}{\epsilon^2 n} \hat{\Sigma}), \qquad (2)$$

where $\mathbf{I}$ is the d-dimensional identify matrix and $\hat{\Sigma} = \mathbf{Z}\mathbf{Z}^\mathsf{T} \in \mathbb{R}^{d \times d}$ is an estimate of covariance. Theoretical justifications have been provided in [42] that the coding vectors in $\mathbf{Z}$ can be explained by packing ϵ-balls into the space spanned by $\mathbf{Z}$ (*sphere packing* [12]) or by computing the number of bits needed to quantize the SVD of $\mathbf{Z}$ subject to the precision. As coding rate produces a good estimate of the compactness of latent features, a few attempts [10, 39] have been made in the areas of multi-view learning and contrastive learning, which select informative features from d dimensions by maximizing the coding rate.

4. Proposed Approach

4.1. Kernel Coding Rate Maximization

The core task in pool-based active learning is to select the most informative samples from the unlabeled pool $\mathcal{D}_U$, which motivates us to replace the **covariance estimate** of features with the **kernel matrix** of samples in the coding rate formula (see Equation (2)). To each point cloud subset $\mathcal{D} = \{\mathcal{P}_i\}_{i=1}^n \subset \mathcal{D}_U$ of size n, we refer to this new coding length $\Re^K(\mathfrak{M}, \epsilon)$ as the **kernel coding rate**, which represents the minimal number of bits to encode features $\mathfrak{M}$:

$$\mathfrak{M} = \boldsymbol{g}(\mathcal{D}, \boldsymbol{\theta}_g) = [\mathfrak{m}_1, \mathfrak{m}_2, ..., \mathfrak{m}_n] \in \mathbb{R}^{d \times n}.$$

The latent features extracted from $\boldsymbol{g}(\cdot; \boldsymbol{\theta}_g)$ can help find the most informative samples irrespective of the downstream tasks of classification and/or regression. We mathematically define the kernel coding rate $\Re^K(\mathfrak{M}, \epsilon)$ as:

$$\Re^K(\mathfrak{M}, \epsilon) := \frac{1}{2} \log \det(\mathbf{I} + \frac{n}{\epsilon^2 d} \mathbf{K}_{\mathfrak{M}, \mathfrak{M}}), \qquad (3)$$

with the kernel matrix $\mathbf{K}_{\mathfrak{M}, \mathfrak{M}} = [K(\mathfrak{m}_i, \mathfrak{m}_j)] \in \mathbb{R}^{n \times n}$. In each round $r \in \{1, \ldots, R\}$, we use *greedy search* to find an optimal subset $\mathcal{D}_r^*$ with size n from the unlabeled pool $\mathcal{D}_U$ by maximizing the kernel coding rate:

$$\mathcal{D}_r^* = \underset{\mathcal{D} \subset \mathcal{D}_U \text{with} |\mathcal{D}| = n}{\arg \max} \Re^K(\mathfrak{M}, \epsilon), \qquad (4)$$

where $\mathfrak{M} = \boldsymbol{g}(\mathcal{D}; \boldsymbol{\theta}_g)$. Notably, in the above equation, we consider positive semi-definite (PSD) kernel $K : \mathfrak{m} \times \mathfrak{m} \to \mathbb{R}$, which characterizes the similarity between each pair of embeddings of point clouds, and hence, helps with avoiding redundancy. The most basic type of PSD kernel to consider is linear kernel, which is defined by the dot product between two features:

$$K_{\text{Linear}}(\mathfrak{m}_i, \mathfrak{m}_j) = \langle \mathfrak{m}_i, \mathfrak{m}_j \rangle = \mathfrak{m}_i^\mathsf{T} \mathfrak{m}_j. \qquad (5)$$

This kernel can be computed very quickly yet it has limitations when dealing with high-dimensional input variables,

such as in our case where $d = W \times L \times F$. The linear kernel may capture the noise and fluctuations in the data instead of the underlying pattern, making it less generalizable to the unseen data. Therefore, while the linear kernel can be a useful starting point, it may be necessary to consider other PSD kernels that are better suited to the specific characteristics of the point cloud data at hand. More discussion on non-linear kernels (*e.g.*, Laplace RBF kernel) is provided in the supplementary material. In the following subsection, we explain a more appropriate PSD kernel K to be used in KECOR, where we can jointly consider *aleatoric* and *epistemic* uncertainties from the model perspective.

4.1.1 Empirical Neural Tangent Kernel K_{NTK}

Compared with linear kernel, empirical neural tangent kernel (NTK) [28, 46] defined as the outer product of the neural network Jacobians, has been shown to lead to improved generalization performance in deep learning models. The yielded NTK matrix quantifies *how changes in the inputs affect the outputs* and captures the relationships between the inputs and outputs in a compact and interpretable way.

To efficiently compute the NTK kernel matrix, we first consider a $(L + 1)$-layer fully connected neural network $\boldsymbol{f}(\cdot; \boldsymbol{\theta}) : \mathfrak{m} \mapsto \hat{\mathfrak{B}}$ as a proxy network for the detection head $\boldsymbol{h}(\cdot; \boldsymbol{\theta}_h)$, as shown in Figure 1. The l-th layer in the proxy network $\boldsymbol{f}$ has d_l neurons, where l ranges from 0 to L. In the forward pass computation, the output from the l-th layer is defined as,

$$\boldsymbol{f}^{(l)}(\mathfrak{m}_i; \boldsymbol{\theta}^{(l)}) = \sigma(\frac{1}{\sqrt{d_l}} \boldsymbol{W}^{(l)} \boldsymbol{f}^{(l-1)}(\mathfrak{m}_i) + \beta \boldsymbol{b}^{(l)}), \quad (6)$$

where $\beta \geq 0$ is a constant controlling the effect of bias and $\boldsymbol{f}^0(\mathfrak{m}_i) = \mathfrak{m}_i$. $\sigma(\cdot)$ stands for a pointwise nonlinear function. Note that the weight matrix $\boldsymbol{W}^{(l)} \in \mathbb{R}^{d_l \times d_{l-1}}$ is rescaled by $1/\sqrt{d_l}$ to avoid divergence, which refers to *NTK parameterization* [28]. For notation simplicity, we denote $\boldsymbol{f}^{(l)}(\mathfrak{m}_i; \boldsymbol{\theta}^{(l)})$ as $\boldsymbol{f}_i^{(l)}$. We omit the bias term and rewrite Equation (6) as

$$\boldsymbol{f}_i^{(l)} = \tilde{\boldsymbol{W}}^{(l)} \tilde{\mathfrak{m}}_i^{(l-1)}, \qquad (7)$$

where $\tilde{\boldsymbol{W}}^{(l)} = [\boldsymbol{W}^{(l)}, \boldsymbol{b}^{(l)}] \in \mathbb{R}^{d_l \times (d_{l-1})+1}$, $\tilde{\mathfrak{m}}_i^{(l-1)} = [\frac{\sigma}{\sqrt{d_l}} \boldsymbol{f}_i^{(l-1)}; \sigma\beta] \in \mathbb{R}^{d_{l-1}+1}$. We denote all parameters in the proxy network as $\boldsymbol{\theta} = [\tilde{\boldsymbol{W}}^{(1)}, \ldots, \tilde{\boldsymbol{W}}^{(L)}]$. To endow the proxy network $\boldsymbol{f}$ with the capability to mimic the behavior of the detector head, we train the proxy $\boldsymbol{f}$ with the labeled data $\mathcal{D}_L$ by using an empirical regression loss function $\mathcal{L} : \mathbb{R}^{d_L} \to \mathbb{R}_+$ *e.g.*, mean squared error (MSE) to supervise the 3D box and ROI predictions. It is found that training neural networks using the MSE loss involves solving a linear regression problem with the kernel trick [28], where the kernel K_{NTK} is defined as the derivative of the output of a neural network with respect to its inputs at the l-th layer, evaluated at the initial conditions:

$$K_{\mathrm{NTK}}(\mathfrak{m}_i, \mathfrak{m}_j) = \langle \nabla_{\boldsymbol{\theta}} \boldsymbol{f}^{(l)}(\mathfrak{m}_i), \nabla_{\boldsymbol{\theta}} \boldsymbol{f}^{(l)}(\mathfrak{m}_j) \rangle. \quad (8)$$

By incorporating Equation (7) and the chain rule, we obtain the factorization of derivates as the ultimate form of empirical NTK kernel:

$$K_{\mathrm{NTK}}(\mathfrak{m}_i, \mathfrak{m}_j) = \sum_{l=1}^{L} \langle \frac{\mathrm{d}\boldsymbol{f}_i^{(L)}}{\mathrm{d}\boldsymbol{f}_i^{(l)}} \left(\tilde{\mathfrak{m}}_i^{(l-1)} \right)^{\mathsf{T}}, \frac{\mathrm{d}\boldsymbol{f}_j^{(L)}}{\mathrm{d}\boldsymbol{f}_j^{(l)}} \left(\tilde{\mathfrak{m}}_j^{(l-1)} \right)^{\mathsf{T}} \rangle_F$$
$$= \sum_{l=1}^{L} \left\langle \tilde{\mathfrak{m}}_i^{(l-1)}, \tilde{\mathfrak{m}}_j^{(l-1)} \right\rangle \cdot \left\langle \frac{\mathrm{d}\boldsymbol{f}_i^{(L)}}{\mathrm{d}\boldsymbol{f}_i^{(l)}}, \frac{\mathrm{d}\boldsymbol{f}_j^{(L)}}{\mathrm{d}\boldsymbol{f}_j^{(l)}} \right\rangle,$$

where $\langle \cdot, \cdot \rangle_F$ indicates the Frobenius inner product. The above equation demonstrates that the NTK kernel is constructed by taking into account the gradient contributions from multiple layers, which naturally captures the *epistemic uncertainty* in the detector's behavior.

4.1.2 Last-layer Gradient Kernel K_{Last}

To verify the validity of aggregating gradients from multiple layers, we derive a simplified variant of the NTK kernel K_{NTK}, which only considers the gradients *w.r.t* the parameters from the last layer of the proxy network:

$$K_{\mathrm{Last}}(\mathfrak{m}_i, \mathfrak{m}_j) = \langle \nabla_{\tilde{\boldsymbol{W}}^{(L)}} \boldsymbol{f}^{(l)}(\mathfrak{m}_i), \nabla_{\tilde{\boldsymbol{W}}^{(L)}} \boldsymbol{f}^{(l)}(\mathfrak{m}_j) \rangle. \quad (9)$$

We have conducted extensive experiments to compare the impact of different kernels selected in the kernel coding rate maximization criteria as shown in Section 5.4.1. Empirical results suggest that the one-stage detectors generally favor K_{Last} while two-stage detectors tend to perform better with K_{NTK} on 3D detection recognition tasks.

4.2. Acquisition Function

As described in Equation (4), our approach selects the most informative point clouds based on the extracted features $\mathfrak{m}$ and gradient maps and thereby facilitate downstream predictions in the detector head. However, for two-stage detectors like PV-RCNN, the classification prediction is made in the region proposal network (refer to dotted boxes and lines in Figure 1) before feeding features into the detector head. Therefore, the features $\mathfrak{m}$ alone cannot determine the informativeness for the box classification task. To make the proposed KECOR strategy applicable to both one-stage and two-stage detectors, we introduce the modified acquisition function by including an entropy regularization term as below:

$$\mathcal{D}_r^* = \operatorname*{arg\,max}_{\mathcal{D} \subset \mathcal{D}_U \mathrm{\,with\,} |\mathcal{D}| = n} \mathfrak{R}^K(\mathfrak{M}, \epsilon) + \sigma_{\mathrm{ent}} \mathcal{H}(\hat{Y}), \quad (10)$$

where $\mathcal{H}(\cdot)$ represents the mean entropy of all classification logits generated from the classifier. The effect of the hyperparameter σ_{ent} is studied in Section 5.4.2. The overall algorithm is summarized in the supplementary material.

5. Experiments

5.1. Experimental Setup

3D Point Cloud Detection Datasets. KITTI [21] is one of the most representative datasets for point cloud based object detection. The dataset consists of 3,712 training samples (*i.e.,* point clouds) and 3,769 *val* samples. The dataset includes a total of 80,256 labeled objects with three commonly used classes for autonomous driving: cars, pedestrians, and cyclists. The Waymo Open dataset [60] is a challenging testbed for autonomous driving, containing 158,361 training samples and 40,077 testing samples. The sampling intervals for KITTI and Waymo are set to 1 and 10, respectively. To fairly evaluate baselines and the proposed method on KITTI dataset [21], we follow the work of [58]: we utilize Average Precision (AP) for 3D and Bird Eye View (BEV) detection, and the task difficulty is categorized to EASY, MODERATE, and HARD, with a rotated IoU threshold of 0.7 for cars and 0.5 for pedestrian and cyclists. The results evaluated on the validation split are calculated with 40 recall positions. To evaluate on Waymo dataset [60], we adopt the officially published evaluation tool for performance comparisons, which utilizes AP and the Average Precision Weighted by Heading (APH). The respective IoU thresholds for vehicles, pedestrians, and cyclists are set to 0.7, 0.5, and 0.5. Regarding detection difficulty, the Waymo test set is further divided into two levels. LEVEL 1 (and LEVEL 2) indicates there are more than five inside points (at least one point) in the ground-truth objects.

2D Image Detection Dataset. On the PASCAL VOC 2007 dataset [17], we use 4,000 images in the *trainval* set for training and 1,000 images in the *test* set for testing. For active selection, we set 500 labeled images as random initialization. Then $n = 500$ images are labeled at each cycle until reaching 2,000. The trained SSD [38] detectors are evaluated with mean Average Precision (mAP) at IoU = 0.5 on VOC07. Unspecified training details are the same as in [9].

Implementation Details. To ensure the reproducibility of the baselines and the proposed approach, we implement KECOR based on the public ACTIVE-3D-DET [41] toolbox that can accommodate most of the public LiDAR detection benchmark datasets. The hyperparameter σ_{ent} is fixed to 0.1 and 0.5 on the KITTI and Waymo Open datasets, respectively. The hyperparameter β is set to 0.1, which is consistent with [42]. For the proxy network, we build two-layer fully connected networks, with the latent dimensions d_1 and d_2 fixed to 256. The source code and other implementation details of active learning protocols can be found in the supplementary material for reference.

5.2. Baselines

For fair comparisons, eleven active learning baselines are included in our experiments: **RAND** is a random sampling method selecting n samples. **ENTROPY** [61] is an

Table 1: Performance comparisons on the 3D AP (%) scores with generic AL and applied AL for detection on KITTI *val* set with 1% queried bounding boxes. PV-RCNN is used as the backbone architecture for all approaches.

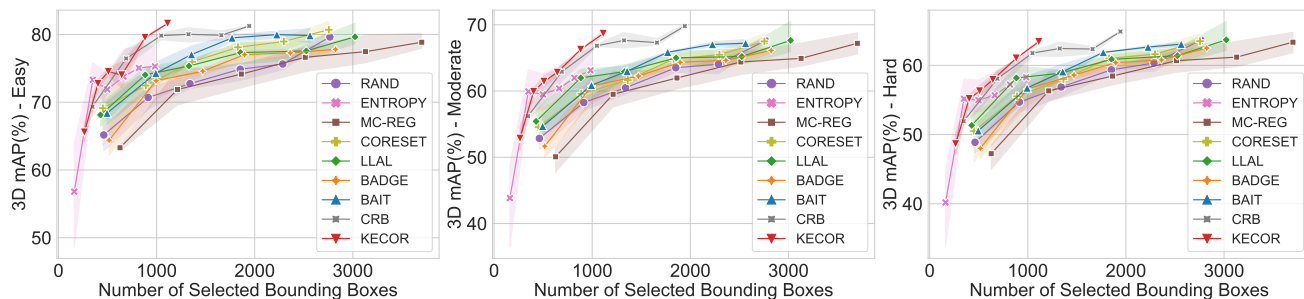|	Method	CAR			PEDESTRIAN			CYCLIST			AVERAGE		
		EASY	MOD.	HARD	EASY	MOD.	HARD	EASY	MOD.	HARD	EASY	MOD.	HARD
Generic	CORESET [55]	87.77	77.73	72.95	47.27	41.97	38.19	81.73	59.72	55.64	72.26	59.81	55.59
	BADGE [5]	89.96	75.78	70.54	51.94	46.24	40.98	84.11	62.29	58.12	75.34	61.44	56.55
	LLAL [68]	89.95	78.65	**75.32**	56.34	49.87	45.97	75.55	60.35	55.36	73.94	62.95	58.88
AL Detection	MC-REG [41]	88.85	76.21	73.47	35.82	31.81	29.79	73.98	55.23	51.85	66.21	54.41	51.70
	MC-MI [18]	86.28	75.58	71.56	41.05	37.50	33.83	86.26	60.22	56.04	71.19	57.77	53.81
	CONSENSUS [54]	90.14	78.01	74.28	56.43	49.50	44.80	78.46	55.77	53.73	75.01	61.09	57.60
	LT/C [30]	88.73	78.12	73.87	55.17	48.37	43.63	83.72	63.21	59.16	75.88	63.23	58.89
	CRB [41]	90.98	79.02	74.04	64.17	54.80	50.82	86.96	67.45	63.56	80.70	67.81	62.81
	KECOR	**91.71**	**79.56**	74.05	**65.37**	**57.33**	**51.56**	**87.80**	**69.13**	**64.65**	**81.63**	**68.67**	**63.42**

Figure 2: 3D mAP (%) of KECOR and AL baselines on the KITTI *val* split with PV-RCNN.

uncertainty-based approach that selects n samples with the highest entropy of predicted labels. **LLAL** [68] is an uncertainty-based method using an auxiliary network to predict indicative loss and select samples that are likely to be mispredicted. **CORESET** [55] is a diversity-based method that performs core-set selection using a greedy furthest-first search on both labeled and unlabeled embeddings. **BADGE** [5] is a hybrid approach that selects instances that are both diverse and of high magnitude in a hallucinated gradient space. The comparison involved four variants of deep active learning (**MC-MI** [18], **MC-REG** [41], **CRB** [41]), and two adapted from 2D detection, **LT/C** [30] and **CONSENSUS** [54]) for 3D detection. MC-MI used Monte Carlo dropout and mutual information to determine the uncertainty of point clouds, while MC-REG used M-round MC-DROPOUT to determine regression uncertainty and select top-n samples with the greatest variance for label acquisition. LT/C measures class-specific localization tightness, while CONSENSUS calculates the variation ratio of minimum IoU value for each RoI-match of 3D boxes. To testify the active learning performance on the 2D detection task, we compare KECOR with **AL-MDN** [9] approach, which predicts the parameter of Gaussian mixture model and computes epistemic uncertainty as the variance of Gaussian modes.

5.3. Results on KITTI and Waymo Open Datasets

To validate the effectiveness of the proposed KECOR, active learning approaches were evaluated under various settings on the KITTI and Waymo Open datasets.

Results of PV-RCNN on KITTI. Figure 6 depicts the 3D mAP scores of PV-RCNN trained by different AL approaches with an increasing number of selected 3D bounding boxes. Specifically, ENTROPY selects point clouds with the least number of bounding boxes, as higher classification entropy indicates less chance of containing objects in point clouds. To elaborate further, the number of bounding boxes selected by MC-REG is generally high and of a large variance, as more instances contained in point clouds will trigger higher aleatoric uncertainty in the box regression. It is observed that AL methods KECOR, CRB and BAIT which jointly consider aleatoric and epistemic uncertainties, effectively balance between annotation costs and 3D detector performance across all detection difficulty levels. Among these three methods, the proposed KECOR outperforms CRB and BAIT, reducing the number of required annotations by 36.8% and 64.0%, respectively, without compromising detection performance. A detailed AP score for each class is reported in Table 1 when the box-level annotation budget is set to 800 (*i.e.*, 1% queried bounding boxes). It is worth noting that the AP scores yield by KECOR are observed to be higher than all other AL baselines. The BEV scores and the detailed analysis are provided in the supple-
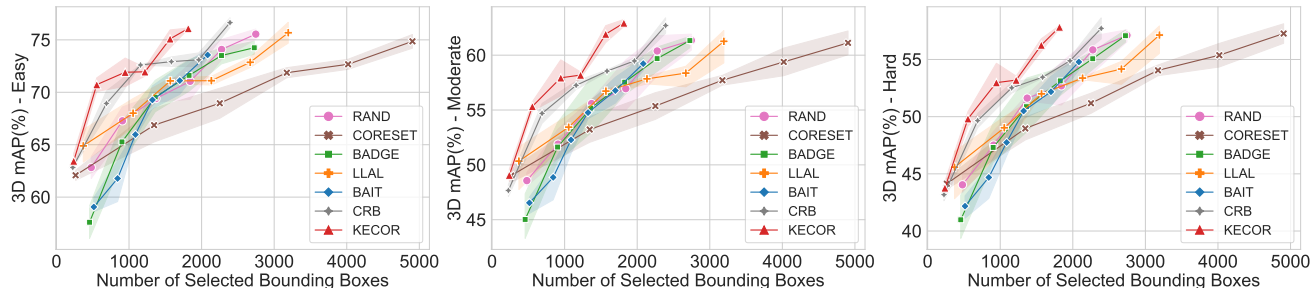
Figure 3: 3D mAP (%) of KECOR and AL baselines on the KITTI *val* split with SECOND.

Table 2: 3D mAP and BEV scores (%) on the KITTI *val* split with one-stage 3D detector SECOND.

	VENUE	3D Detection mAP			BEV Detection mAP		
		EASY	MODERATE	HARD	EASY	MODERATE	HARD
RAND		$69.33_{\pm0.62}$	$55.48_{\pm0.43}$	$51.53_{\pm0.33}$	$75.66_{\pm1.10}$	$63.77_{\pm0.86}$	$59.71_{\pm0.95}$
CORESET [55]	ICLR'18	$66.86_{\pm2.27}$	$53.22_{\pm1.65}$	$48.97_{\pm1.42}$	$73.08_{\pm1.80}$	$61.03_{\pm1.98}$	$56.95_{\pm1.53}$
LLAL [68]	CVPR'19	$69.19_{\pm3.43}$	$55.38_{\pm3.63}$	$50.85_{\pm3.24}$	$76.52_{\pm2.24}$	$63.25_{\pm3.11}$	$59.07_{\pm2.80}$
BADGE [5]	ICLR'20	$69.92_{\pm2.90}$	$55.60_{\pm2.72}$	$51.23_{\pm2.58}$	$76.07_{\pm2.70}$	$63.39_{\pm2.52}$	$59.47_{\pm2.49}$
BAIT [4]	NeurIPS'21	$69.45_{\pm3.53}$	$55.61_{\pm2.94}$	$51.25_{\pm2.42}$	$76.04_{\pm1.75}$	$63.49_{\pm2.14}$	$53.40_{\pm2.00}$
CRB [41]	ICLR'23	$72.33_{\pm0.35}$	$58.06_{\pm0.30}$	$53.09_{\pm0.31}$	$78.84_{\pm0.27}$	$65.82_{\pm0.07}$	$61.25_{\pm0.22}$
KECOR-LINEAR		$70.55_{\pm1.17}$	$55.54_{\pm1.05}$	$50.91_{\pm0.84}$	$77.50_{\pm0.44}$	$63.97_{\pm0.61}$	$59.55_{\pm0.25}$
KECOR-RBF		$73.03_{\pm0.49}$	$58.54_{\pm0.94}$	$53.70_{\pm0.81}$	$79.00_{\pm0.67}$	$66.55_{\pm0.43}$	$61.92_{\pm0.52}$
KECOR-LAST		$\mathbf{74.30}_{\pm0.42}$	$\mathbf{60.68}_{\pm0.13}$	$\mathbf{55.26}_{\pm0.05}$	$\mathbf{80.50}_{\pm0.39}$	$\mathbf{68.31}_{\pm0.03}$	$\mathbf{63.26}_{\pm0.01}$
KECOR		$\mathbf{74.05}_{\pm0.16}$	$\mathbf{60.38}_{\pm0.06}$	$\mathbf{55.34}_{\pm0.23}$	$\mathbf{80.00}_{\pm0.12}$	$\mathbf{68.20}_{\pm0.35}$	$\mathbf{63.20}_{\pm0.25}$

mentary material.

Results of SECOND on KITTI. We further test the active learning performance of one-stage detector SECOND on the KITTI dataset. Table 2 reports the 3D mAP and BEV mAP scores across different difficulty levels with around 1,400 bounding boxes. A performance gain of KECOR over the state-of-the-art approach CRB is about 3.5% and 2.8% on average with respect to 3D and BEV mAP scores. Figure 3 shows a more intuitive trend that KECOR achieves a higher boost on the recognition mAP at the MODERATE and HARD levels. This implies that the incorporated NTK kernel helps capture the objects that are of sparse point clouds and generally hard to learn, which enhances the detector's capacity on identifying challenging objects.

Results of PV-RCNN on Waymo Open. To study the scalability and effectiveness of KECOR, we conduct experiments on the large-scale Waymo Open dataset, the results of which are illustrated in Figure 4a and Figure 4b for different difficulty levels. The proposed approach surpasses all existing AL approaches by a large margin, which verifies the validity of the proposed kernel coding rate maximization strategy. Notably, KECOR saves around 44.4% 3D annotations than CRB when reaching the same detection performance.

5.4. Ablation Study

We conducted a series of experiments to understand the impact of kernels and the coefficient σ_{ent} on the performance of our approach on the KITTI dataset. The central tendency of the performance (*e.g.*, mean mAP) and variations (*e.g.*, error bars) are reported based on outcomes from

the two trials for each variant.

5.4.1 Impact of Kernels

We conducted experiments on KITTI to evaluate the effect of kernels on the proposed method, and the active learning results yielded with the PV-RCNN and SECOND backbones are reported in Figure 5a and Table 2, respectively. We refer the variants of KECOR with the linear kernel, Laplace RBF kernel, last-layer gradient kernel, and NTK kernel as to KECOR-LINEAR, KECOR-RBF, KECOR-LAST and KECOR, respectively. Figure 5a shows that KECOR achieves the highest mAP scores (68.67%) among KECOR-LINEAR (66.82%) and KECOR-LAST (68.31%) at the moderate difficulty level. Regarding the box-level annotation costs, KECOR acquires a comparable amount as KECOR-LAST, while KECOR-LINEAR requires 1.91 times more bounding boxes. Table 2 shows that KECOR-LAST and KECOR gain a relative 7.6% and 5.5% improvement, respectively, over KECOR-LINEAR on the 3D mAP and BEV mAP scores with the SECOND detector. In particular, KECOR surpasses the variant KECOR-RBF by 2.5% and 1.9% on 3D mAP and BEV mAP, respectively. The performance gains evidence that the applied NTK kernel not only captures the non-linear relationship between the inputs and outputs, but also the aleatoric uncertainty for both tasks.

5.4.2 Impact of Coefficient σ_{ent}

We delve into the susceptibility of our method to various values of the coefficient σ_{ent}, which varies in $\{0, 0.01, 0.05, 0.1, 0.3\}$. The performances of different variants are measured using the mean average precision
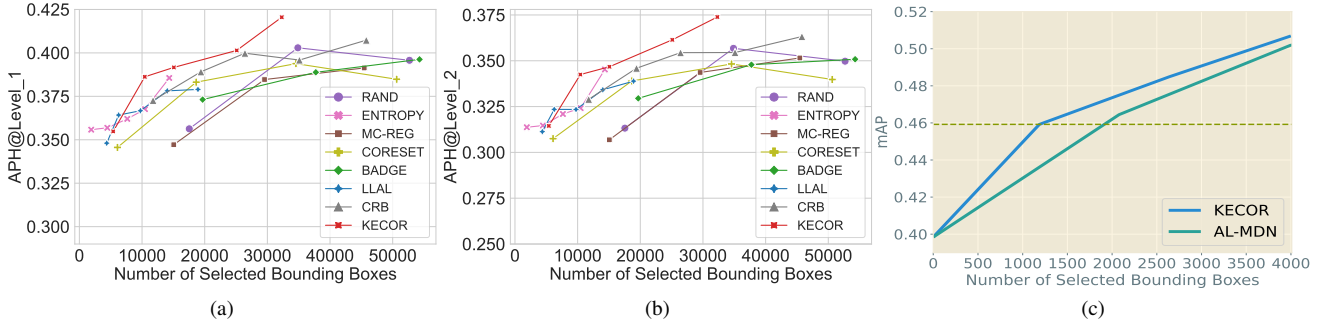
Figure 4: (a-b) Mean APH scores of KECOR and AL baselines on the Waymo Open *val* split with PV-RCNN at the difficulty Level 1 and Level 2, respectively. (c) Performance comparison of 2D object detection on PASCAL VOC07 dataset.
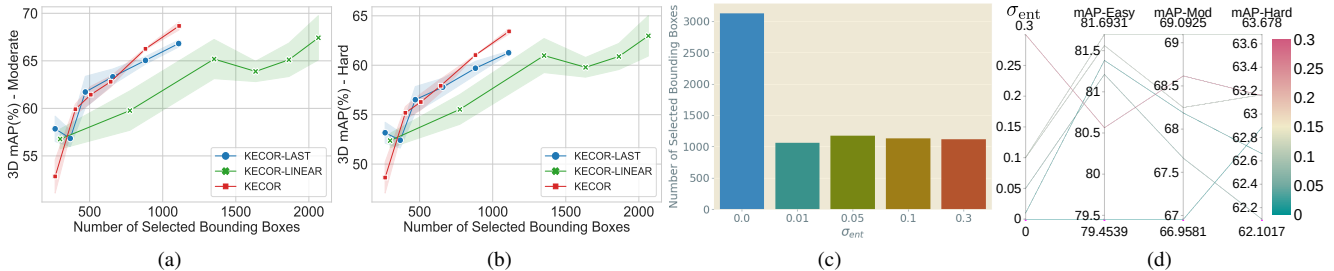

Figure 5: Ablation studies on the (a-b) impact of kernels in KECOR and (c-d) impact of coefficient σ_{ent} on the KITTI dataset.

(mAP) results on the KITTI dataset. The detection performance for the last round and the total amount of queried 3D bounding boxes for each variant are summarized in the barplot (Figure 5c) and the parallel plot (Figure 5d). The results show that different values of σ_{ent} only had a limited impact on the 3D mAP scores, with variations up to 2.8%, 3.0% and 2.5% across different difficulty levels, which affirms the resilience of the proposed method to the selection of σ_{ent}. Notably, the variant of KECOR without the classification entropy term ($\sigma_{\text{ent}} = 0$) produces approximately 3 times more bounding boxes to annotate than other variants as shown in Figure 5c. We infer this was attributed to the higher entropy of point clouds containing fewer repeated objects, which regularizes the acquisition criteria and ensures a minimal annotation cost. We provide an additional study on the impact of σ_{ent} on the Waymo Open dataset in the supplementary material.

5.5. Analysis on Running Time

To ensure the proposed approach is efficient and reproducible, we have conducted an analysis of the average runtime yielded by the proposed KECOR and the state-of-the-art active 3D detection method CRB on two benchmark datasets, *i.e.*, KITTI and Waymo Open. The training hours of each approach are reported in Table 3. With different choices on base kernels, our finding indicates that KECOR outperforms CRB in terms of running efficiency, achieving a relative improvement of $5.2\% \sim 6.4\%$ on the KITTI dataset and of $24.0\% \sim 26.4\%$ on the large-scale Waymo Open dataset. These results suggest that KECOR is a highly effec-

tive and efficient approach for active 3D object detection, especially for large datasets, and has the potential to benefit real-world applications.

Table 3: Running time (hours) comparisons with PV-RCNN.

AL Strategy	KITTI	Waymo Open	Improvement
CRB	11.935	86.595	
KECOR-LINEAR	11.170	63.701	+6.4% / +26.4%
KECOR-LAST	11.313	64.741	+5.2% / +25.2%
KECOR	11.313	65.782	+5.2% / +24.0%

5.6. Results on 2D Object Detection

To examine the versatility of the proposed KECOR strategy, we conducted additional experiments on the task of 2D object detection. To ensure a fair comparison with AL-MDN [9], we adopt the SSD [38] architecture with VGG16 as the backbone. With a fixed budget for acquiring labeled images, KECOR demonstrates superior performance over AL-MDN in the early cycles as shown in Figure 4c. As the green dotted line indicates, KECOR requires only 1,187 box annotations to achieve the same level of mAP scores, while AL-MDN requires 1,913 annotations, resulting in approximately 38% savings in labeling costs. These results evidence that KECOR effectively trades off between annotation costs and detection performance.

6. Conclusion

This paper studies a novel informative-theoretic acquisition criterion for the active 3D detection task, which well balances a trade-off between the quantity of selected bounding boxes and the yielded detection performance. By maximizing the kernel coding rate, the informative point clouds

are identified and selected, which bring in unique and novel knowledge for both 3D box classification and regression. The proposed KECOR is proven to be versatile to one-stage and two-stage detectors and also applicable to 2D object detection tasks. The proposed strategy achieves superior performance on benchmark datasets and also significantly reduces the running time and labeling costs simultaneously.

A. More Discussions on Laplace RBF Kernel

Recall that in Section 4.1, we have discussed that the linear kernel K_{Linear} can be a useful starting point, it may be necessary to consider other PSD kernels that are better suited to the specific characteristics of the point cloud data at hand. The Laplace Radial Basis Function (RBF) kernel, also known as the Laplacian kernel, is a popular choice of kernel in machine learning algorithms. The Laplace RBF kernel maps the input features into a higher-dimensional feature space, where non-linear relationships can be more easily captured. This kernel function for two latent features $\mathfrak{m}_i$ and $\mathfrak{m}_j$ can be mathematically represented as follows:

$$K_{\text{RBF}}(\mathfrak{m}_i, \mathfrak{m}_j) = \exp(-\frac{\|\mathfrak{m}_i - \mathfrak{m}_j\|}{\sigma}), \qquad (11)$$

where σ indicates a hyperparameter that controls the width of the kernel. σ is empirically set to 1.0. The Laplace RBF kernel has a sharp cutoff beyond a distance of σ, which makes it less sensitive to outliers than the Gaussian RBF kernel. More experimental results and analysis can be found in Section E.

B. The Algorithm of KECOR

In this section, we elaborate on the entire workflow of the proposed KECOR approach for active 3D detection. As illustrated in Algorithm 1, the training and selection process includes three stages: (I) detection pre-training with the labeled set (Line 4), (II) active selection (Line 9) from the unlabeled pool, and (III) detection re-training with the updated labeled set (Line 19). Notably, in the pretraining stage, the proxy network is jointly learned to predict the outputs from the detector head h. The outputs can be ROI (forground confidence) only for SECOND [63] or with box regression for PV-RCNN [58]. The training of the proxy network is iterated by 10 and 20 epochs for KITTI and Waymo Open datasets. When the training of the detection model and proxy network converges, we move to the next active selection stage in which n informative point clouds will be selected based on the kernel coding rate maximization criterion presented in Equation (10). The selected point clouds are expected to bring novel and unique knowledge for the following re-training of the detector. The whole process will be gone through multiple times, until the number of selected point clouds reaches the pre-defined budget B.

Algorithm 1 THE PSEUDOCODE OF KECOR.

```
1: Inputs:
        D_L: a set of labeled point clouds
        D_U: a set of unlabeled point clouds
        Ω: annotators
        B: a total budget for selection
        g(·; θ_g): a feature extractor
        h(·; θ_h): a detector head
        f(·; θ): a proxy network of detector head
        count: a counter of point clouds selected
2: Outputs:
        g(·; θ_g): the trained feature extractor
        h(·; θ_h): the trained detector head
3: count ← 0
4: procedure PRE-TRAIN DETECTOR(g, h, f, D_L)
5:     Train g and h with detection loss
6:     Train f with regression loss L
7: end procedure
8: while count < B do
9:     procedure ACTIVE SELECTION(g, f, D_U)
10:        Extract features and gradients from g and f
11:        Extract classification entropy for P ∈ D_U
12:        Calculate K_NTK for any subset D ⊂ D_U
13:        Select the optimal subset D_r*  ▷ refer to Eq. (10)
14:    end procedure
15:    D_U ← D_U \ D_r*         ▷ remove the selected subset
16:    D_S ← Ω(D_r*)         ▷ query labels from annotators
17:    D_L ← D_L ∪ D_S
18:    count += n   ▷ number of selected data n = |D_r*|
19:    procedure RE-TRAIN DETECTOR(g, h, f, D_L)
20:        Train g and h with detection loss
21:        Train f with regression loss L
22:    end procedure
23: end while
```

C. Implementation Details

Following the same setting in [41], 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 D_L. With the annotated data, the 3D detector is trained with E epochs, which is then freezed to select n candidates from 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 $E = 30$, $R = 6$
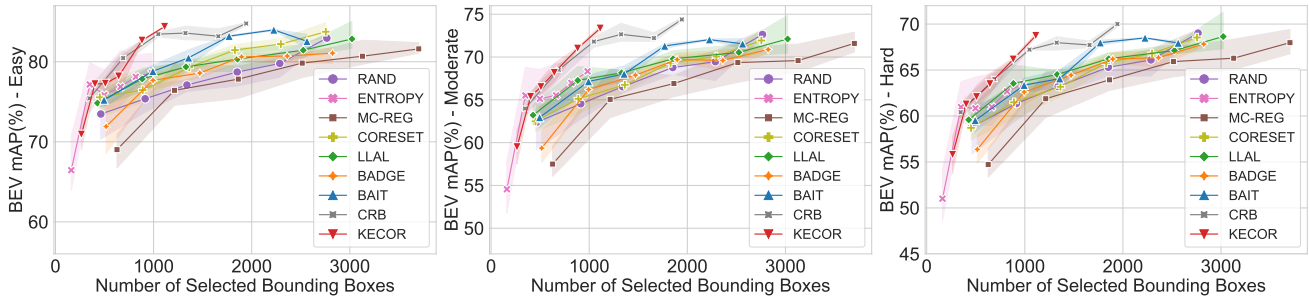
Figure 6: 3D mAP (%) of KECOR and AL baselines on the KITTI *val* split with PV-RCNN.

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.

D. 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 6. 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.

E. 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 7, 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.

F. 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 8. Similar 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.

G. 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 regard to different evaluation metrics in Figure 9. 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 $\sigma_{\mathrm{ent}} = 0.7$ 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] Sharat Agarwal, Himanshu Arora, Saket Anand, and Chetan Arora. Contextual diversity for active learning. In *Proc.*
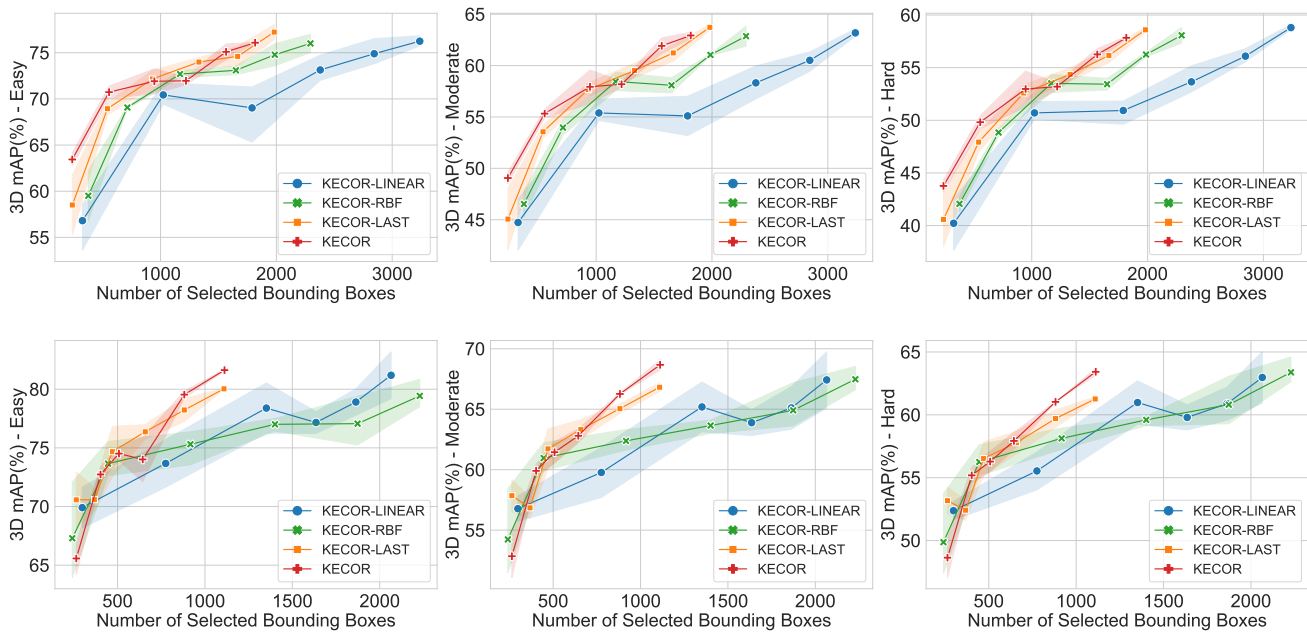
Figure 7: Ablation study on the different choices of kernels on the KITTI *val* split with SECOND (Top row) and PV-RCNN (Bottom row) across a variety of difficulty levels.
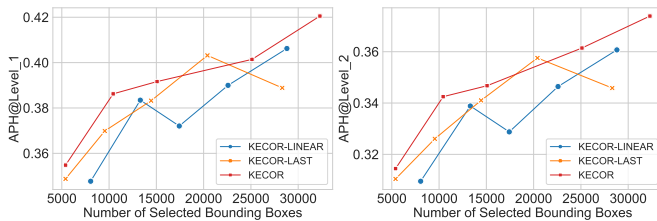

Figure 8: Ablation study on the different choices of kernels on the Waymo Open dataset with PV-RCNN.
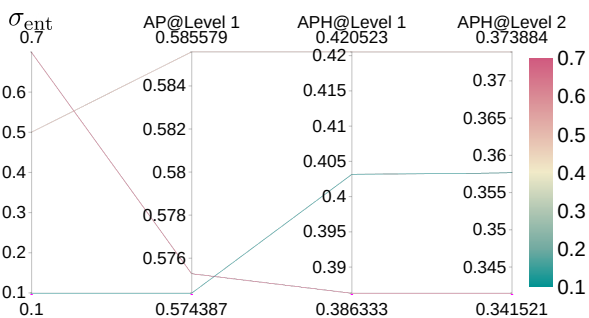

Figure 9: The parallel plot of the impact of σ_{ent} on Waymo.

European Conference on Computer Vision (ECCV), volume 12361, pages 137–153, 2020. 2

[2] Oisin Mac Aodha, Neill D. F. Campbell, Jan Kautz, and Gabriel J. Brostow. Hierarchical subquery evaluation for active learning on a graph. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 564–571, 2014. 2

[3] Sanjeev Arora, Simon S. Du, Zhiyuan Li, Ruslan Salakhutdinov, Ruosong Wang, and Dingli Yu. Harnessing the power of infinitely wide deep nets on small-data tasks. In *Proc. International Conference on Learning Representations (ICLR)*, 2020. 3

[4] Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Sham M. Kakade. Gone fishing: Neural active learning with fisher embeddings. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, pages 8927–8939, 2021. 2, 7

[5] Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *Proc. International Conference on Learning Representations (ICLR)*, 2020. 1, 2, 6, 7

[6] William H. Beluch, Tim Genewein, Andreas Nürnberger, and Jan M. Köhler. The power of ensembles for active learning in image classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9368–9377, 2018. 2

[7] Shubhang Bhatnagar, Sachin Goyal, Darshan Tank, and Amit Sethi. PAL : Pretext-based active learning. In *Proc. British Machine Vision Conference (BMVC)*, page 195. BMVA Press, 2021. 2

[8] Jacob Binia, Moshe Zakai, and Jacob Ziv. On the epsilon -entropy and the rate-distortion function of certain non-

gaussian processes. *IEEE Transactions on Information Theory*, 20(4):517–524, 1974. 2, 3

[9] Jiwoong Choi, Ismail Elezi, Hyuk-Jae Lee, Clément Farabet, and Jose M. Alvarez. Active learning for deep object detection via probabilistic modeling. In *Proc. International Conference on Computer Vision (ICCV)*, pages 10244–10253, 2021. 2, 5, 6, 8

[10] C. Mario Christoudias, Raquel Urtasun, and Trevor Darrell. Unsupervised feature selection via distributed coding for multi-view object recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. 4

[11] Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. Batch active learning at scale. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, pages 11933–11944, 2021. 2

[12] Thomas M. Cover and Joy A. Thomas. *Elements of information theory (2. ed.)*. 2006. 2, 3, 4

[13] 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 *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, pages 1201–1209, 2021. 1, 3

[14] Shengheng Deng, Zhihao Liang, Lin Sun, and Kui Jia. VISTA: boosting 3d object detection via dual cross-view spatial attention. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8438–8447. IEEE, 2022. 1

[15] Pan Du, Suyun Zhao, Hui Chen, Shuwen Chai, Hong Chen, and Cuiping Li. Contrastive coding for active learning under class distribution mismatch. In *Proc. International Conference on Computer Vision (ICCV)*, pages 8907–8916, 2021. 2

[16] Ehsan Elhamifar, Guillermo Sapiro, Allen Y. Yang, and S. Shankar Sastry. A convex optimization framework for active learning. In *Proc. International Conference on Computer Vision (ICCV)*, pages 209–216, 2013. 1, 2

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 5

[18] Di Feng, Xiao Wei, Lars Rosenbaum, Atsuto Maki, and Klaus Dietmayer. Deep active learning for efficient training of a lidar 3d object detector. In *Proc. Intelligent Vehicles Symposium, (IV)*, pages 667–674, 2019. 1, 2, 6

[19] Alexander Freytag, Erik Rodner, and Joachim Denzler. Selecting influential examples: Active learning with expected model output changes. In *Proc. European Conference on Computer Vision (ECCV)*, pages 562–577, 2014. 2

[20] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proc. International Conference on Machine Learning (ICML)*, volume 48, pages 1050–1059, 2016. 2

[21] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012. 5

[22] Yuhong Guo. Active instance sampling via matrix partition. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, pages 802–810, 2010. 1, 2

[23] Ali Harakeh, Michael Smart, and Steven L. Waslander. Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In *Proc. International Conference on Robotics and Automation (ICRA)*, pages 87–93, 2020. 2

[24] Mahmudul Hasan and Amit K. Roy-Chowdhury. Context aware active learning of activity recognition models. In *Proc. International Conference on Computer Vision (ICCV)*, pages 4543–4551, 2015. 2

[25] Chenhang He, Ruihuang Li, Shuai Li, and Lei Zhang. Voxel set transformer: A set-to-set approach to 3d object detection from point clouds. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8407–8417. IEEE, 2022. 1

[26] Neil Houlsby, Ferenc Huszar, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *CoRR*, abs/1112.5745, 2011. 2

[27] Long-Kai Huang, Junzhou Huang, Yu Rong, Qiang Yang, and Ying Wei. Frustratingly easy transferability estimation. In *Proc. International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pages 9201–9225. PMLR, 2022. 2

[28] Arthur Jacot, Clément Hongler, and Franck Gabriel. Neural tangent kernel: Convergence and generalization in neural networks. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, pages 8580–8589, 2018. 3, 4

[29] Ajay J. Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2372–2379, 2009. 1, 2

[30] Chieh-Chi Kao, Teng-Yok Lee, Pradeep Sen, and Ming-Yu Liu. Localization-aware active learning for object detection. In *Proc. Asian Conference on Computer (ACCV)*, pages 506–522, 2018. 1, 2, 6

[31] Kwanyoung Kim, Dongwon Park, Kwang In Kim, and Se Young Chun. Task-aware variational adversarial active learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8166–8175, 2021. 2

[32] Yoon-Yeong Kim, Kyungwoo Song, JoonHo Jang, and Il-Chul Moon. LADA: look-ahead data acquisition via augmentation for deep active learning. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, pages 22919–22930, 2021. 2

[33] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, pages 7024–7035, 2019. 2

[34] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12697–12705. Computer Vision Foundation / IEEE, 2019. 1, 3

[35] Jaehoon Lee, Lechao Xiao, Samuel S. Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In *Neural Tangent Kernel: Convergence and Generalization in Neural Networks*, pages 8570–8581, 2019. 3

[36] David D. Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Proc. International Conference on Machine Learning (ICML)*, pages 148–156, 1994. 1, 2

[37] Peng Liu, Lizhe Wang, Rajiv Ranjan, Guojin He, and Lei Zhao. A survey on active deep learning: From model driven to data driven. *ACM Computing Surveys*, 54(10s):221:1–221:34, 2022. 1

[38] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *Proc. European Conference on Computer Vision (ECCV)*, volume 9905, pages 21–37. Springer, 2016. 5, 8

[39] Xin Liu, Zhongdao Wang, Yali Li, and Shengjin Wang. Self-supervised learning via maximum entropy coding. *CoRR*, abs/2210.11464, 2022. 2, 4

[40] Zhuoming Liu, Hao Ding, Huaping Zhong, Weijia Li, Jifeng Dai, and Conghui He. Influence selection for active learning. In *Proc. International Conference on Computer Vision (ICCV)*, pages 9254–9263, 2021. 2

[41] 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, 2, 5, 6, 7, 9

[42] Yi Ma, Harm Derksen, Wei Hong, and John Wright. Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE Transactions on Pattern Analysis and Machine Intelligence.*, 29(9):1546–1562, 2007. 2, 3, 4, 5

[43] David J. C. MacKay. Information-based objective functions for active data selection. *Journal of Neural Computation*, 4(4):590–604, 1992. 2

[44] Mohamad Amin Mohamadi, Wonho Bae, and Danica J. Sutherland. Making look-ahead active learning strategies feasible with neural tangent kernels. *CoRR*, abs/2206.12569, 2022. 1, 2

[45] Hieu Tat Nguyen and Arnold W. M. Smeulders. Active learning using pre-clustering. In Carla E. Brodley, editor, *Proc. International Conference on Machine Learning (ICML)*, 2004. 2

[46] Roman Novak, Jascha Sohl-Dickstein, and Samuel S. Schoenholz. Fast finite width neural tangent kernel. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *Proc. International Conference on Machine Learning (ICML)*, volume 162, pages 17018–17044. PMLR, 2022. 3, 4

[47] Younghyun Park, Soyeong Kim, Wonjeong Choi, Dong-Jun Han, and Jaekyun Moon. Active learning for object detection with evidential deep learning and hierarchical uncertainty aggregation. In *Proc. International Conference on Learning Representations (ICLR)*, 2023. 2

[48] Amin Parvaneh, Ehsan Abbasnejad, Damien Teney, Gholam-reza (Reza) Haffari, Anton van den Hengel, and Javen Qinfeng Shi. Active learning by feature mixing. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12237–12246, 2022. 1, 2

[49] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B. Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM Computing Surveys*, 54(9):180:1–180:40, 2022. 1

[50] Dan Roth and Kevin Small. Margin-based active learning for structured output spaces. In *Proc. European Conference on Machine Learning (ECML)*, pages 413–424, 2006. 1, 2

[51] Nicholas Roy and Andrew McCallum. Toward optimal active learning through monte carlo estimation of error reduction. In *Proc. International Conference on Machine Learning (ICML)*, pages 441–448, 2001. 2

[52] Soumya Roy, Asim Unmesh, and Vinay P. Namboodiri. Deep active learning for object detection. In *Proc. British Machine Vision Conference (BMVC)*, page 91, 2018. 2

[53] David Schinagl, Georg Krispel, Horst Possegger, Peter M. Roth, and Horst Bischof. Occam's laser: Occlusion-based attribution maps for 3d object detectors on lidar data. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1131–1140. IEEE, 2022. 1

[54] Sebastian Schmidt, Qing Rao, Julian Tatsch, and Alois C. Knoll. Advanced active learning strategies for object detection. In *Proc. Intelligent Vehicles Symposium, (IV)*, pages 871–876, 2020. 1, 2, 6

[55] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *Proc. International Conference on Learning Representations (ICLR)*, 2018. 1, 2, 6, 7

[56] Burr Settles, Mark Craven, and Soumya Ray. Multiple-instance active learning. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, pages 1289–1296, 2007. 2

[57] Guangsheng Shi, Ruifeng Li, and Chao Ma. Pillarnet: Real-time and high-performance pillar-based 3d object detection. In *Proc. European Conference on Computer Vision (ECCV)*, volume 13670, pages 35–52, 2022. 1, 3

[58] 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 *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10526–10535, 2020. 1, 3, 5, 9

[59] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointr-cnn: 3d object proposal generation and detection from point cloud. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–779. Computer Vision Foundation / IEEE, 2019. 1, 3

[60] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo

open dataset. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2443–2451, 2020. 5

[61] Dan Wang and Yi Shang. A new active labeling method for deep learning. In *Proc. International Joint Conference on Neural Networks (IJCNN)*, pages 112–119, 2014. 2, 5

[62] Jiaxi Wu, Jiaxin Chen, and Di Huang. Entropy-based active learning for object detection with progressive diversity constraint. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9387–9396. IEEE, 2022. 1, 2

[63] Yan Yan, Yuxing Mao, and Bo Li. SECOND: sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 1, 3, 9

[64] Bin Yang, Wenjie Luo, and Raquel Urtasun. PIXOR: real-time 3d object detection from point clouds. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7652–7660. IEEE, 2018. 1, 3

[65] Yi Yang, Zhigang Ma, Feiping Nie, Xiaojun Chang, and Alexander G. Hauptmann. Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113:113–127, 2015. 1, 2

[66] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11037–11045. Computer Vision Foundation / IEEE, 2020. 1, 3

[67] Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. STD: sparse-to-dense 3d object detector for point cloud. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1951–1960. IEEE, 2019. 1, 3

[68] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 93–102, 2019. 1, 2, 6, 7

[69] Weiping Yu, Sijie Zhu, Taojiannan Yang, and Chen Chen. Consistency-based active learning for object detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3950–3959. IEEE, 2022. 2

[70] Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. In *Proc. Annual Conference on Neural Information Processing (NeurIPS)*, 2020. 2

[71] Tianning Yuan, Fang Wan, Mengying Fu, Jianzhuang Liu, Songcen Xu, Xiangyang Ji, and Qixiang Ye. Multiple instance active learning for object detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5330–5339, 2021. 2