LiDAL: Inter-frame Uncertainty Based Active Learning for 3D LiDAR Semantic Segmentation

Zeyu Hu[1]*, Xuyang Bai[1], Runze Zhang[2], Xin Wang[2], Guangyuan Sun[2], Hongbo Fu[3], and Chiew-Lan Tai[1]

[1] Hong Kong University of Science and Technology
{zhuam,xbaiad,taicl}@cse.ust.hk
[2] Lightspeed & Quantum Studios, Tencent
{ryanrzzhang,alexinwang,gerrysun}@tencent.com
[3] City University of Hong Kong
hongbofu@cityu.edu.hk

Abstract. We propose LiDAL, a novel active learning method for 3D LiDAR semantic segmentation by exploiting inter-frame uncertainty among LiDAR frames. Our core idea is that a well-trained model should generate robust results irrespective of viewpoints for scene scanning and thus the inconsistencies in model predictions across frames provide a very reliable measure of uncertainty for active sample selection. To implement this uncertainty measure, we introduce new inter-frame divergence and entropy formulations, which serve as the metrics for active selection. Moreover, we demonstrate additional performance gains by predicting and incorporating pseudo-labels, which are also selected using the proposed inter-frame uncertainty measure. Experimental results validate the effectiveness of LiDAL: we achieve 95% of the performance of fully supervised learning with less than 5% of annotations on the SemanticKITTI and nuScenes datasets, outperforming state-of-the-art active learning methods. Code release: https://github.com/hzykent/LiDAL

Keywords: Active Learning, 3D LiDAR Semantic Segmentation

1 Introduction

Light detection and ranging (LiDAR) sensors capture more precise and farther-away distance measurements than conventional visual cameras, and have become a necessity for an accurate perception system of outdoor scenes. These sensors generate rich 3D geometry of real-world scenes as 3D point clouds to facilitate a thorough scene understanding, in which 3D LiDAR semantic segmentation serves as a cornerstone. The semantic segmentation task is to parse a scene and assign an object class label to each point in 3D point clouds, thus providing point-wise perception information for numerous downstream applications like robotics [43] and autonomous vehicles [20].

* intern at Tencent Lightspeed & Quantum Studios

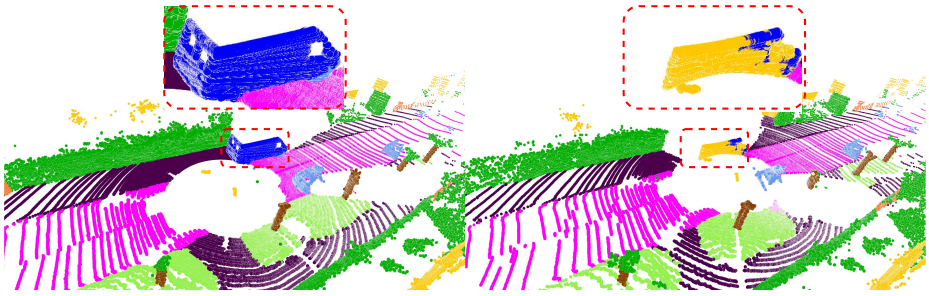

Fig. 1: **Illustration of inter-frame uncertainty.** While in one frame (Left) an object is correctly predicted as "vehicle" (highlighted in a red box), in the subsequent frame (Right) a large part of this object is mistakenly predicted as "fence" when scanned from a different viewpoint.

Thanks to the large-scale LiDAR datasets [1,4] made publicly available in recent years, the state of the art in 3D LiDAR semantic segmentation has been significantly pushed forward [42,12,59]. However, the requirement of fully labeled point clouds for existing segmentation methods has become a major obstacle to scaling up the perception system or extending it to new scenarios. Typically, since a LiDAR sensor may perceive millions of points per second, exhaustively labeling all points is extremely laborious and time-consuming. It poses demands on developing label-efficient approaches for 3D LiDAR semantic segmentation.

Active learning provides a promising solution to reduce the costs associated with labeling. Its core idea is to design a learning algorithm that can interactively query a user to label new data samples according to a certain policy, leading to models trained with only a fraction of the data while yielding similar performances. Inspired by 2D counterparts [44,8,29,21,37], some previous works have explored active learning for 3D LiDAR semantic segmentation [28,23,51,36]. However, these methods almost exclusively operate on single LiDAR frames. Such a strategy is surprising since, unlike most 2D datasets, in which images are captured as independent samples, 3D LiDAR datasets are generally scanned as continuous point cloud frames. As a consequence, inter-frame constraints naturally embedded in the LiDAR scene sequences are largely ignored. We believe such constraints are particularly interesting for examining the quality of network predictions; i.e., the same object in a LiDAR scene should receive the same label when scanned from different viewpoints (see Fig. 1).

In this work, we propose to exploit inter-frame constraints in a novel view-consistency-based uncertainty measure. More specifically, we propose new inter-frame divergence and entropy formulations based on the variance of predicted score functions across continuous LiDAR frames. For a given (unlabeled) object (e.g., the one in a red box in Fig. 1), if its predicted labels differ across frames, we assume faulty network predictions and then strive to obtain user-specified labels for the most uncertain regions. In addition to the main active learning formulation, we also explore further improvements to the labeling efficiency with

self-training by utilizing the proposed uncertainty measure in an inverse way. During each active learning iteration, we augment the user-specified labels with pseudo-labels generated from the most certain regions across frames to further boost performance without extra annotations or much computational cost.

To summarize, our contributions are threefold:

1. We propose a novel active learning strategy for 3D LiDAR semantic segmentation by estimating model uncertainty based on the inconsistency of predictions across frames.
2. We explore self-training in the proposed active learning framework and show that further gains can be realized by including pseudo-labels.
3. Through extensive experiments, we show that the proposed active learning strategy and self-training technique significantly improve labeling efficiency over baselines, and establish the state of the art in active learning for 3D LiDAR semantic segmentation.

2 Related Work

Compared to fully supervised methods [42,14,59,39,6,13], label-efficient 3D semantic segmentation is a relatively open research problem. Previous explorations can be roughly divided into five categories: transfer learning, unsupervised and self-supervised learning, weakly-supervised learning, active learning, and self-training. LiDAL falls into both the active learning and self-training categories.

Transfer Learning. Taking advantage of existing fully labeled datasets, transfer learning has been introduced to 3D semantic segmentation for reducing the annotation costs. Various domain adaptation approaches have been developed to make them perform well in novel scenarios given only labeled data from other domains [24,56,18] or even synthetic training sets [50]. They achieve fairly decent results but still require fully labeled data from a source domain and fail to generalize to new scenarios that are highly different from the source.

Unsupervised and Self-supervised Learning. Leveraging the colossal amount of unlabeled data, pre-trained models can be fine-tuned on a small set of labeled data to alleviate the over-dependence on labels and thus achieve satisfactory performances [9,41,38,53,15]. Pseudo tasks used for pre-training include reconstructing space [32], contrast learning [52,26,10], ball cover prediction [35], instance discrimination [58], and point completion [46], etc. Compared to other label-efficient counterparts, these methods require more labeled data and most of them only apply to object-level point clouds.

Weakly-supervised Learning. Instead of point-by-point labeling in fully supervised learning, weak labels take various forms like scene-level or sub-cloud-level labels [30,48], 2D supervision [45], fewer point labels [55,5,57,11,49], seg-level labels [27,40], and box-level labels [25], etc. These methods can reduce the number of labeled samples, but either require intricate labeling processes or produce much more inferior results than the fully-supervised counterparts.

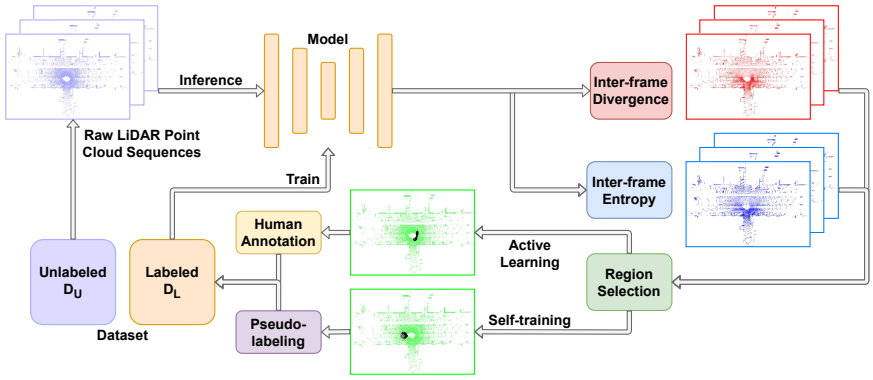

Fig. 2: **Pipeline of LiDAL.** In each round of active learning, we first train a 3D LiDAR semantic segmentation network in supervision with labeled dataset D_L. Second, we use the trained network to compute an inter-frame divergence score and an inter-frame entropy score for all regions from the unlabeled dataset D_U. We then select a batch of regions based on these scores for active learning and self-training, and finally request their respective labels from the human annotation and the pseudo-labeling. The process is repeated until the labeling budget is exhausted or all training data is labeled.

Active Learning. During network training, active learning methods iteratively select the most valuable data for label acquisition. The very few existing methods have explored uncertainty measurements like segment entropy [22], color discontinuity [51], and structural complexity [51,36]. The existing methods take LiDAR data as separated frames and only consider intra-frame information. Inspired by a 2D work operating on multi-view images [37], we take advantage of the inter-frame constraints for active learning in this work. Different from this 2D work, due to the distinct natures of 2D images and 3D point clouds, we design novel uncertainty formulations and selection strategies. Moreover, we propose a joint active learning and self-training framework to further exploit the inter-frame constraints.

Self-training. Building on the principle of knowledge distillation, previous methods generate pseudo labels to expand sparse labels [54,27] or to facilitate the network training using only scene-level supervision [30]. In this work, we develop a pseudo-labeling method applied in conjunction with our active learning framework to achieve even greater gains in efficiency.

3 Method

3.1 Overview

The goal of LiDAL is to train a well-performing 3D LiDAR semantic segmentation model with a constrained annotation budget. Specifically, we assume the

availability of data $D = \{D_L, D_U\}$. The data consists of sequences of LiDAR frames, each provided with an ego-pose of the scanning device. Initially, D_L is a small set of LiDAR frames randomly selected from D with each frame having its label annotation, and D_U is a large unlabeled set without any annotations. Following previous works [37,51], we use a sub-scene region as a fundamental query unit to focus on the most informative parts of the LiDAR frames.

As illustrated in Fig. 2, our LiDAL method consists of four main steps: 1. Train the network to convergence using the currently labeled dataset D_L. 2. Calculate the model uncertainty scores for each region of D_U with two indicators: inter-frame divergence and inter-frame entropy (Section 3.2). 3. Select regions based on the uncertainty measures for active learning and self-training (Section 3.3). 4. Obtain labels from human annotation and pseudo-labeling. These steps will be repeated until the labeling budget is exhausted or all training data is labeled.

3.2 Uncertainty Scoring

At each iteration, after the first step of training the network on D_L, our active learning method LiDAL then aims at predicting which samples from D_U are the most informative to the network at the current state. To this end, we introduce two novel uncertainty scoring metrics named *inter-frame divergence* and *inter-frame entropy*. Fig. 3 provides an overview of the scoring process.

Inter-frame Divergence. In a nutshell, the proposed inter-frame divergence score aims at estimating which objects are consistently predicted the same way, irrespective of the scanning viewpoints.

For each frame, we first calculate its point-wise class probability maps using the current trained segmentation network. To attain robust probability predictions, we perform data augmentations with random translation, rotation, scaling, and jittering. The probability P for a point p in frame F_i to belong to class c is given by:

$$P_i^p(c) = \frac{1}{D} \sum_{d=1}^{D} P_{i,d}^p(c), \tag{1}$$

where D is the number of augmented inference runs of the segmentation network, and $P_{i,d}^p(c)$ is the softmax probability of point p belonging to class c in the augmented inference run d.

Next, using the provided ego-pose, we register each frame in the world coordinate system. For each point in a given frame, we find its corresponding points in the neighboring frames and assign to it their associated probability distributions. Implementation details can be found in **Supplementary Section A**. Each point p in frame F_i is now associated with a set of probability distributions Ω_i^p, each coming from a neighboring frame:

$$\Omega_i^p = \{P_j^p, j | F_j^p \text{ corresponds to } F_i^p\}, \tag{2}$$

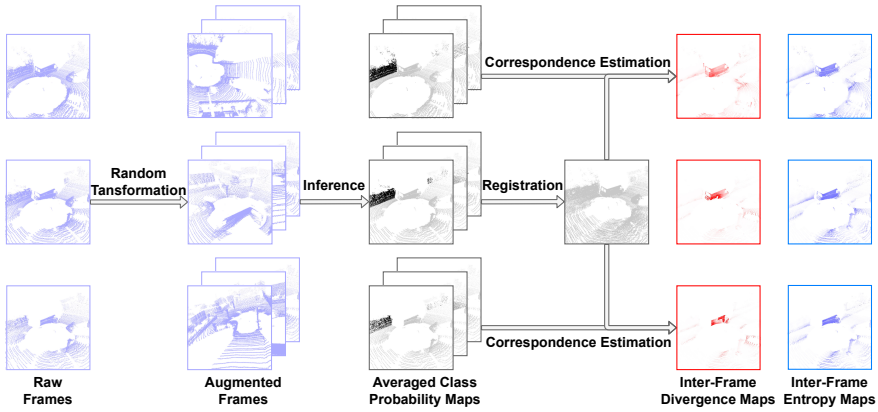

Fig. 3: **Illustration of uncertainty scoring.** For each unlabeled LiDAR frame in the dataset, we first obtain its averaged class probability predictions from augmented inference runs. Next, we register each frame with its provided ego-pose. For each point in the frame, we then find its corresponding points in neighboring frames and assign to it their associated class probability predictions. With the aggregated multiple class probability predictions per point, we compute the inter-frame divergence and entropy scores. We assign these two scores to each unlabeled region by averaging the scores of all the points contained in it.

where F_j^p denotes the point p in frame F_j, and F_j represents one of the neighboring frames of F_i.

In our setting, when estimating the point correspondences between neighboring frames, we assume that the objects in the scene are static and thus the points in the same registered position represent the same object. The moving objects are not specially treated for two reasons. First, they contribute only a small portion of the dataset. Second, when estimating correspondences after registration, the prediction disagreements introduced by the 3D motions can be seen as inter-frame inconsistency and help the system select these informative regions.

The inter-frame divergence corresponds to the average pairwise KL divergence between the class distribution at any given point and the class distributions assigned to that point from the neighboring frames. It effectively captures the degree of agreement between the prediction in the current frame with the predictions coming from the neighboring frames. Specifically, we define the inter-frame divergence score FD for a point p in frame F_i as follows:

$$FD_i^p = \frac{1}{|\Omega_i^p|} \sum_{P_j^p \in \Omega_i^p} D_{KL}(P_i^p || P_j^p), \tag{3}$$

where $D_{KL}(P_i^p || P_j^p)$ is the KL Divergence between distributions P_i^p and P_j^p.

Inter-frame Entropy. After measuring how inconsistent the predictions are across frames, we then define the inter-frame entropy score, which indicates the amount of uncertainty for the network to process a certain point. For a point p in frame F_i, with the aggregated probability distributions Ω_i^p, the mean distribution M_i^p can be calculated as:

$$M_i^p = \frac{1}{|\Omega_i^p|} \sum_{P_j^p \in \Omega_i^p} P_j^p, \tag{4}$$

which can be seen as the marginalization of the prediction probabilities over the scanning viewpoints.

The inter-frame entropy score FE is defined as the entropy of the mean class probability distribution M_i^p:

$$FE_i^p = -\sum_c M_i^p(c) \log(M_i^p(c)). \tag{5}$$

A high inter-frame entropy score implies that on average, the prediction of the current network for this point is significantly uncertain. Since the mean class probability distribution is the average result from both the augmented inference runs and the aggregation of corresponding points, the inter-frame entropy score estimates both the intra-frame uncertainty under random affine transformations and inter-frame uncertainty under viewpoint changes.

3.3 Region Selection

To select the most informative parts of the unlabeled dataset, we opt for using sub-scene regions as the fundamental label querying units, following previous works [37,51]. Our implementation uses the constrained K-means clustering [2] algorithm for region division. An ideal sub-scene region consists of one or several object classes and is lightweight to label for the annotator.

For each region r, the two scores FD_i^r and FE_i^r are computed as the average of the inter-frame divergence and inter-frame entropy scores of all the points contained in r:

$$FD_i^r = \frac{1}{|r|} \sum_{p \in r} FD_i^p, \tag{6}$$

$$FE_i^r = \frac{1}{|r|} \sum_{p \in r} FE_i^p, \tag{7}$$

where $|r|$ is the number of points contained in region r.

Active Learning. We now discuss our active learning strategy utilizing the proposed inter-frame uncertainty scores. Our strategy to select the next region for labeling consists of two steps. First, we look for the region r from frame F_i that has the highest inter-frame divergence score in D_U:

$$(i, r) = \underset{(j,s) \in D_U}{\arg\max}\, FD_j^s, \tag{8}$$

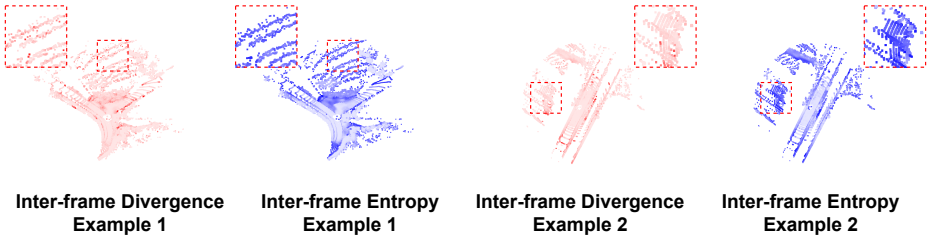

Fig. 4: **Examples of uncertainty scores.** Red and blue indicate inter-frame divergence and entropy, respectively. The darker the color, the higher the value. Due to the sparsity and varying-density property of LiDAR point clouds, neural networks tend to generate class distributions that are more uniform for farther away sparse points. As highlighted by the dotted red boxes, this property results in misleadingly high values for far away sparse points in terms of inter-frame entropy but affects less on the inter-frame divergence scores.

where (j, s) refers to region s from frame F_j.

Since the inter-frame divergence indicates that for each region how inconsistent the predictions are across frames, the scores are similar for all the regions that are in correspondence. To determine which one of the regions in correspondence contains the largest amount of beneficial information to improve the network, we retrieve the set of regions representing the same part of the outdoor scene and denote this set as S. We then look for the region from S with the highest inter-frame entropy score:

$$(k, t) = \underset{(j,s)\in S}{\arg\max}\{FE_j^s | (j, s) \text{ and } (i, r) \text{ overlap}\}, \tag{9}$$

where (k, t) refers to the selected region t from frame F_k. Implementation details can be found in **Supplementary Section A**.

The selected region is added to D_L and all regions in set S are then removed from D_U to avoid label redundancy. The process is repeated until reaching the labeling budget. The active learning algorithm is summarized in Alg. 1.

One possible alternative strategy is to first find the region with the highest inter-frame entropy score and then select the one with the highest inter-frame divergence score in the corresponding set. A similar strategy is implemented in a previous work operating on multi-view images [37]. However, unlike 2D images with dense and uniformly sampled pixels, 3D LiDAR frames have sparse and varying-density points. Specifically, the farther from the scanning viewpoint, the sparser the LiDAR points. As shown in Fig. 4, due to the sparsity, neural networks tend to predict uniform class distributions for peripheral points. This property will result in misleadingly high inter-frame entropy scores for points far away from the scanning viewpoint (Equation 5), while the inter-frame divergence scores remain stable (Equation 3). Considering the robustness of the system, we

opt for the proposed strategy instead of the possible alternative. A quantitative comparison can be found in Section 4.4.

Self-training. To further exploit the inter-frame constraints embedded in the LiDAR sequences, we leverage the fact that our measure of viewpoint inconsistency can also help us identify reliable regions with high-quality pseudo-labels, which can be directly injected into the training set.

On the contrary with respect to active learning, which selects samples with the most uncertain predictions, self-training aims at acquiring confident and accurate pseudo-labels. To this end, we conduct a reversed process of the proposed active learning strategy. Specifically, we first look for the region r from frame F_i that has the lowest inter-frame divergence score in D_U:

$$(i, r) = \underset{(j,s)\in D_U}{\arg\min}\, FD_j^s, \tag{10}$$

We then look for the region from the corresponding set S that has the lowest inter-frame entropy score:

$$(k, t) = \underset{(j,s)\in S}{\arg\min}\{FE_j^s | (j, s) \text{ and } (i, r) \text{ overlap}\}, \tag{11}$$

The pseudo-label of the selected region is retrieved from the network predictions and all regions in set S will not be used for further pseudo-labeling. The process is repeated until reaching the target number of pseudo-labels. In order to prevent label drifting [7], we reset the pseudo label set at each iteration and only select regions that are not already selected in the previous iteration. The self-training algorithm is summarized in Alg. 2.

Algorithm 1 Active Learning

Input:
Data set D, labeled set D_L,
annotation budget B, metric M
Init:
Added samples $A \leftarrow \{\}$
Unlabeled set $D_U \leftarrow D \setminus D_L$
repeat
$(i, r) \leftarrow \arg\max_{(j,s)\in D_U} M_{FD_j^s}$
Retrieve S ▷ Corresponding set
$(k, t) \leftarrow \arg\max_{(j,s)\in S} M_{FE_j^s}$
$A \leftarrow A \cup (k, t)$
$D_L \leftarrow D_L \cup (k, t)$
$D_U \leftarrow D_U \setminus S$
until $|A| = B$ or $|D_U| = 0$
return A

Algorithm 2 Self-training

Input:
Data set D, labeled set D_L,
previous pseudo set P,
target number T, metric M
Init:
New pseudo set $P' \leftarrow \{\}$
$D'_U \leftarrow (D \setminus D_L) \setminus P$ ▷ No re-labeling
repeat
$(i, r) \leftarrow \arg\min_{(j,s)\in D'_U} M_{FD_j^s}$
Retrieve S ▷ Corresponding set
$(k, t) \leftarrow \arg\min_{(j,s)\in S} M_{FE_j^s}$
$P' \leftarrow P' \cup (k, t)$
$D'_U \leftarrow D'_U \setminus S$
until $|P'| = T$ or $|D'_U| = 0$
return P'

4 Experiments

To demonstrate the effectiveness of our proposed method, we now present various experiments conducted on two large-scale 3D LiDAR semantic segmentation datasets, i.e., SemanticKITTI [1] and nuScenes [4]. We first introduce the datasets and evaluation metrics in Section 4.1, and then present the experimental settings in Section 4.2. We report the results on the SemanticKITTI and nuScenes datasets in Section 4.3, and the ablation studies in Section 4.4.

4.1 Datasets and Metrics

SemanticKITTI [1]. SemanticKITTI is a large-scale driving-scene dataset derived from the KITTI Vision Odometry Benchmark and was collected in Germany with the Velodyne-HDLE64 LiDAR. The dataset consists of 22 sequences containing 43,552 point cloud scans. We perform all our experiments using the official training (seq 00-07 and 09-10) and validation (seq 08) split. 19 classes are used for segmentation.
nuScenes [4]. nuScenes was collected in Boston and Singapore with 32-beam LiDAR sensors. It contains 1,000 scenes of 20s duration annotated with 2Hz frequency. Following the official train/val splits, we perform all label acquisition strategies on the 700 training sequences (28k scans) and evaluate them on 150 validation sequences (6k scans). 16 classes are used for segmentation.
Metrics. For evaluation, we report mean class intersection over union (mIoU) results for both the SemanticKITTI and nuScenes datasets following the official guidance.

4.2 Experimental Settings

Network Architectures. To verify the effectiveness of the proposed active learning strategy on various network architectures, we adopt MinkowskiNet [6] based on sparse convolution, and SPVCNN [39] based on point-voxel CNN, as our backbone networks for their great performance and high efficiency. We make the same choices for the network architectures as ReDAL [51], a recent state-of-the-art active learning method of 3D semantic segmentation, for better comparison.

Baseline Active Learning Methods. We select eight baseline methods for comparison, including random frame selection (RAND_{fr}), random region selection (RAND_{re}), segment-entropy (SEGENT) [22], softmax margin (MAR) [17,31,47], softmax confidence (CONF) [34,47], softmax entropy (ENT) [16,47], core-set selection (CSET) [33], and ReDAL [51]. The implementation details for all the methods are explained in **Supplementary Section B**.

Learning Protocol. Following the same protocol as ReDAL [51], the model is initialized by training on $x_{init}\%$ of randomly selected LiDAR frames with full annotations. The active learning process consists of K rounds of the following

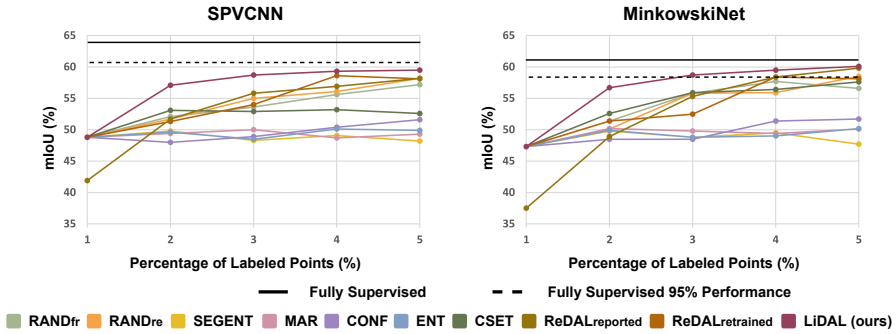

Fig. 5: **Mean intersection over union scores on SemanticKITTI Val [1].** Detailed results can be found in **Supplementary Section C**.

actions: 1. Finetune the model on the current labeled set D_L. 2. Select $x_{active}\%$ of data from the current unlabeled set D_U for human annotation according to different active selection strategies. 3. Update D_L and D_U.

The labeling budget is measured by the percentage of labeled points. For both SemanticKITTI and nuScenes datasets, we use $x_{init} = 1\%$, $K = 4$, and $x_{active} = 1\%$. For self-training, the target number of pseudo-labels in terms of percentage of labeled points $T = 1\%$. To ensure the reliability of the results, all the experiments are performed three times and the average results are reported. More training details can be found in **Supplementary Section A**.

4.3 Results and Analysis

In this section, we present the performance of our approach compared to the baseline methods on the SemanticKITTI and nuScenes datasets. Fig. 5 and Fig. 6 show the comparative results. In each subplot, the x-axis represents the percentage of labeled points and the y-axis indicates the mIoU score achieved by the respective networks, which are trained with data selected through different active learning strategies.

Since most of the baseline methods are not designed for LiDAR point clouds, we re-implement these methods for LiDAR data based on their official codes. For ReDAL [51], in its published paper, it is evaluated on the SemanticKITTI dataset but not on the nuScenes dataset. For the SemanticKITTI dataset, we find that the reported scores of its initial networks (trained with 1% of randomly selected frames) are way lower than our implementations (41.9 vs 48.8 for SPVCNN and 37.5 vs 47.3 for MinkowskiNet). We retrained its networks and got better results using its official code but with a finer training schedule (details can be found in **Supplementary Section A**). Both the retrained results and the reported results are presented in Fig. 5. For nuScenes, we adapt its official code and report the results.

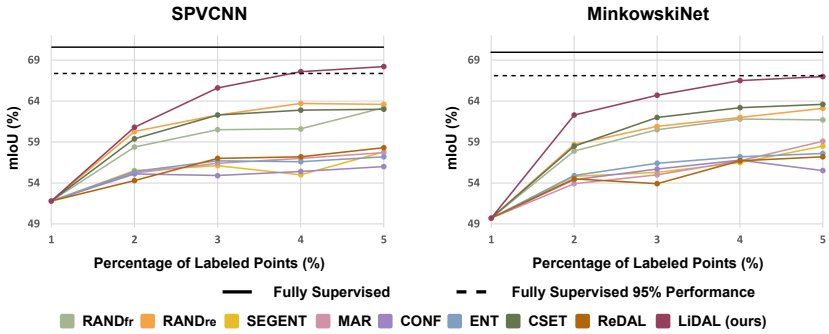

Fig. 6: **Mean intersection over union scores on nuScenes Val [4].** Detailed results can be found in **Supplementary Section C**.

SemanticKITTI. As shown in Fig. 5, our proposed LiDAL significantly surpasses the existing active learning strategies using the same percentages of labeled data. Specifically, our method is able to reach nearly 95% of the fully supervised performance with only 5% of labeled data for the SPVCNN network and even achieve about 98% of the fully supervised result for the MinkowskiNet network.

In addition, we notice that many active learning strategies perform worse than the random baseline and some even bring negative effects on the network performance (e.g., the mIoU scores may drop after adding the data samples selected by SEGENT). For uncertainty-based methods, such as SEGENT and CONF, since the model uncertainty values are calculated only within each frame and are biased by the peripheral points due to the scanning property of LiDAR, their performances are degraded. Even for pure diversity-based approaches, such as CSET, since the LiDAR datasets are captured as continuous frames and have plenty of redundant information among neighboring frames, simply clustering features may fail to produce diverse label acquisition.

Moreover, we observe that the performance gap between RAND_{re} and RAND_{fr} is trivial. It showcases that if not combined with effective uncertainty measures, region-based training brings little benefit to the network performance.

nuScenes. We also evaluate our algorithm on the novel nuScenes dataset and report the results in Fig. 6. As shown in the figure, our method outperforms all the competitors in terms of mIoU under all the experimental settings. Specifically, for both SPVCNN and MinkowskiNet, our method achieves more than 95% of the fully supervised performances with only 5% of labeled data.

Compared to the results of SemanticKITTI, similar phenomena can be witnessed that many active learning strategies perform worse than the random baseline. However, the negative effects are alleviated and the mIoU scores consistently increase by adding data samples selected by most strategies. A possible explanation is that, since the nuScenes dataset contains 1,000 scenes of tens of

Table 1: **Ablation study: building components.** FD: inter-frame divergence score; NMS: non-maximum suppression, i.e., select the region with the highest score in the corresponding set; FE: inter-frame entropy score.

FD	Frame-level	Region-level	NMS	FE	Pseudo	mIoU(%)
✓	✓					51.8
✓		✓				52.5
✓		✓	✓			55.5
✓		✓		✓		56.4
✓		✓		✓	✓	57.1

frames while the SemanticKITTI dataset contains only 22 scenes of thousands of frames, the network is less likely to be biased by the data selected from nuScenes than that from SemanticKITTI.

4.4 Ablation Study

In this section, we conduct a number of controlled experiments that demonstrate the effectiveness of the building modules in LiDAL, and also examine some specific decisions in our LiDAL design. All the experiments are conducted on the SemanticKITTI validation set evaluating the performance of the SPVCNN network trained on the data selected in the first active learning round, keeping all the hyper-parameters the same. More ablation studies can be found in **Supplementary Section D**.

Building Components. In Table 1, we evaluate the effectiveness of each component of our method. **1. Effect of region-level labeling.** "FD + Frame-level" represents the baseline, which is to select frames with the highest average inter-frame divergence scores for training. By changing from "FD + Frame-level" to "FD + Region-level" (selecting regions), we can improve the performance by 0.7%. This improvement is brought by focusing on the most informative parts of the scenes. **2. Effect of active selection strategy.** "FD + Region-level + NMS" refers to selecting only the region with the highest inter-frame divergence score in the corresponding set. By avoiding the label redundancy, we can gain about 3% of improvement. "FD + Region-level + FE" refers to the proposed selection strategy described in Section 3.3. From the proposed inter-frame entropy measure, we further improve about 0.9%. **3. Effect of pseudo-labels.** "FD + Region-level + FE + Pseudo" denotes the complete strategy of LiDAL. The introduction of pseudo-labels brings around 0.7% of performance improvement.

Region Selection Strategies. In Section 3.3, we discuss two possible region selection strategies for both active learning and self-training. We advocate the proposed one that first finds corresponding sets using the inter-frame divergence

Table 2: **Ablation study**: **(Left)** Region selection strategy; **(Right)** Target number of pseudo-label.

Strategy	mIoU(%)
FE + FD	55.7
FD + FE	57.1

Target Number(%)	mIoU(%)
0.0	56.4
0.5	56.8
1.0	57.1
2.0	56.4

scores and then selects regions with the inter-frame entropy scores. To justify our choice, we implement both strategies and report the results in Table 2 (Left). "FD + FE" refers to the proposed strategy and "FE + FD" refers to the possible alternative strategy that first finds corresponding sets using the entropy scores and then selects regions with the divergence scores. As shown in the table, the proposed strategy significantly outperforms the possible alternative strategy. It may be caused by the misleadingly high entropy values of peripheral points, as illustrated in Fig. 4.

Target Number of Pseudo-labels. In Section 3.3, we explore self-training in the proposed active learning framework and show that further gains can be realized by including pseudo-labels in Table 1. To investigate the impact of pseudo-labels, we inject different numbers of pseudo-labels into the training set and report the results in Table 2 (Right). We observe that with the increasing number of pseudo-labels, the gain of network performance first increases and then decreases. We speculate that adding pseudo-labels with a reasonable number will improve the network performance but superfluous pseudo-labels may bring unhelpful training biases and label noises. A further study on pseudo-labels can be found in **Supplementary Section D**.

5 Conclusion

In this paper, we have presented a novel active learning strategy for 3D LiDAR semantic segmentation, named LiDAL. Aiming at exploiting the inter-frame constraints embedded in LiDAR sequences, we propose two uncertainty measures estimating the inconsistencies of network predictions among frames. We design a unified framework of both active learning and self-training by utilizing the proposed measures. Extensive experiments show that LiDAL achieves state-of-the-art results on the challenging SemanticKITTI and nuScenes datasets, significantly improving over strong baselines. For future works, one straightforward direction is to explore the potential of inter-frame constraints for RGB-D sequences of indoor scenes. Moreover, we believe that future works with special treatments for moving objects will further improve the performance.

Acknowledgements. This work is supported by Hong Kong RGC GRF 16206722 and a grant from City University of Hong Kong (Project No. 7005729).

References

1. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: Semantickitti: A dataset for semantic scene understanding of lidar sequences. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9297–9307 (2019)
2. Bradley, P.S., Bennett, K.P., Demiriz, A.: Constrained k-means clustering. Microsoft Research, Redmond **20**(0), 0 (2000)
3. Butt, M.A., Maragos, P.: Optimum design of chamfer distance transforms. IEEE Transactions on Image Processing **7**(10), 1477–1484 (1998)
4. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11621–11631 (2020)
5. Cheng, M., Hui, L., Xie, J., Yang, J.: Sspc-net: Semi-supervised semantic 3d point cloud segmentation network. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 1140–1147 (2021)
6. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019)
7. Feng, Q., He, K., Wen, H., Keskin, C., Ye, Y.: Active learning with pseudo-labels for multi-view 3d pose estimation. arXiv preprint arXiv:2112.13709 (2021)
8. Górriz, M., Giró Nieto, X., Carlier, A., Faure, E.: Cost-effective active learning for melanoma segmentation. In: ML4H: Machine Learning for Health NIPS, Workshop at NIPS 2017. pp. 1–5 (2017)
9. Hassani, K., Haley, M.: Unsupervised multi-task feature learning on point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8160–8171 (2019)
10. Hou, J., Graham, B., Nießner, M., Xie, S.: Exploring data-efficient 3d scene understanding with contrastive scene contexts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15587–15597 (2021)
11. Hu, Q., Yang, B., Fang, G., Guo, Y., Leonardis, A., Trigoni, N., Markham, A.: Sqn: Weakly-supervised semantic segmentation of large-scale 3d point clouds with 1000x fewer labels. arXiv preprint arXiv:2104.04891 (2021)
12. Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A.: Randla-net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11108–11117 (2020)
13. Hu, Z., Bai, X., Shang, J., Zhang, R., Dong, J., Wang, X., Sun, G., Fu, H., Tai, C.L.: Vmnet: Voxel-mesh network for geodesic-aware 3d semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15488–15498 (2021)
14. Hu, Z., Zhen, M., Bai, X., Fu, H., Tai, C.l.: Jsenet: Joint semantic segmentation and edge detection network for 3d point clouds. In: European Conference on Computer Vision. pp. 222–239. Springer (2020)
15. Huang, S., Xie, Y., Zhu, S.C., Zhu, Y.: Spatio-temporal self-supervised representation learning for 3d point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6535–6545 (2021)
16. Hwa, R.: Sample selection for statistical parsing. Computational linguistics **30**(3), 253–276 (2004)

17. Joshi, A.J., Porikli, F., Papanikolopoulos, N.: Multi-class active learning for image classification. In: 2009 ieee conference on computer vision and pattern recognition. pp. 2372–2379. IEEE (2009)
18. Langer, F., Milioto, A., Haag, A., Behley, J., Stachniss, C.: Domain transfer for semantic segmentation of lidar data using deep neural networks. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 8263–8270. IEEE (2020)
19. Levina, E., Bickel, P.: The earth mover's distance is the mallows distance: Some insights from statistics. In: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001. vol. 2, pp. 251–256. IEEE (2001)
20. Li, B., Zhang, T., Xia, T.: Vehicle detection from 3d lidar using fully convolutional network. arXiv preprint arXiv:1608.07916 (2016)
21. Li, H., Yin, Z.: Attention, suggestion and annotation: a deep active learning framework for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 3–13. Springer (2020)
22. Lin, Y., Vosselman, G., Cao, Y., Yang, M.: Efficient training of semantic point cloud segmentation via active learning. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences **5**(2) (2020)
23. Lin, Y., Vosselman, G., Cao, Y., Yang, M.Y.: Active and incremental learning for semantic als point cloud segmentation. ISPRS journal of photogrammetry and remote sensing **169**, 73–92 (2020)
24. Liu, W., Luo, Z., Cai, Y., Yu, Y., Ke, Y., Junior, J.M., Gonçalves, W.N., Li, J.: Adversarial unsupervised domain adaptation for 3d semantic segmentation with multi-modal learning. ISPRS Journal of Photogrammetry and Remote Sensing **176**, 211–221 (2021)
25. Liu, Y., Hu, Q., Lei, Y., Xu, K., Li, J., Guo, Y.: Box2seg: Learning semantics of 3d point clouds with box-level supervision. arXiv preprint arXiv:2201.02963 (2022)
26. Liu, Y., Yi, L., Zhang, S., Fan, Q., Funkhouser, T., Dong, H.: P4contrast: Contrastive learning with pairs of point-pixel pairs for rgb-d scene understanding. arXiv e-prints pp. arXiv–2012 (2020)
27. Liu, Z., Qi, X., Fu, C.W.: One thing one click: A self-training approach for weakly supervised 3d semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1726–1736 (2021)
28. Luo, H., Wang, C., Wen, C., Chen, Z., Zai, D., Yu, Y., Li, J.: Semantic labeling of mobile lidar point clouds via active learning and higher order mrf. IEEE Transactions on Geoscience and Remote Sensing **56**(7), 3631–3644 (2018)
29. Mackowiak, R., Lenz, P., Ghori, O., Diego, F., Lange, O., Rother, C.: Cereals-cost-effective region-based active learning for semantic segmentation. In: BMVC (2018)
30. Ren, Z., Misra, I., Schwing, A.G., Girdhar, R.: 3d spatial recognition without spatially labeled 3d. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13204–13213 (2021)
31. Roth, D., Small, K.: Margin-based active learning for structured output spaces. In: European Conference on Machine Learning. pp. 413–424. Springer (2006)
32. Sauder, J., Sievers, B.: Self-supervised deep learning on point clouds by reconstructing space. Advances in Neural Information Processing Systems **32** (2019)
33. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. arXiv preprint arXiv:1708.00489 (2017)
34. Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: proceedings of the 2008 conference on empirical methods in natural language processing. pp. 1070–1079 (2008)

35. Sharma, C., Kaul, M.: Self-supervised few-shot learning on point clouds. Advances in Neural Information Processing Systems **33**, 7212–7221 (2020)
36. Shi, X., Xu, X., Chen, K., Cai, L., Foo, C.S., Jia, K.: Label-efficient point cloud semantic segmentation: An active learning approach. arXiv preprint arXiv:2101.06931 (2021)
37. Siddiqui, Y., Valentin, J., Nießner, M.: Viewal: Active learning with viewpoint entropy for semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9433–9443 (2020)
38. Sun, W., Tagliasacchi, A., Deng, B., Sabour, S., Yazdani, S., Hinton, G., Yi, K.M.: Canonical capsules: Unsupervised capsules in canonical pose. arXiv preprint arXiv:2012.04718 (2020)
39. Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., Han, S.: Searching efficient 3d architectures with sparse point-voxel convolution. In: European conference on computer vision. pp. 685–702. Springer (2020)
40. Tao, A., Duan, Y., Wei, Y., Lu, J., Zhou, J.: Seggroup: Seg-level supervision for 3d instance and semantic segmentation. arXiv preprint arXiv:2012.10217 (2020)
41. Thabet, A.K., Alwassel, H., Ghanem, B.: Mortonnet: Self-supervised learning of local features in 3d point clouds (2019)
42. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6411–6420 (2019)
43. Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., et al.: Stanley: The robot that won the darpa grand challenge. Journal of field Robotics **23**(9), 661–692 (2006)
44. Vezhnevets, A., Buhmann, J.M., Ferrari, V.: Active learning for semantic segmentation with expected change. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 3162–3169. IEEE (2012)
45. Wang, H., Rong, X., Yang, L., Feng, J., Xiao, J., Tian, Y.: Weakly supervised semantic segmentation in 3d graph-structured point clouds of wild scenes. arXiv preprint arXiv:2004.12498 (2020)
46. Wang, H., Liu, Q., Yue, X., Lasenby, J., Kusner, M.J.: Unsupervised point cloud pre-training via occlusion completion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9782–9792 (2021)
47. Wang, K., Zhang, D., Li, Y., Zhang, R., Lin, L.: Cost-effective active learning for deep image classification. IEEE Transactions on Circuits and Systems for Video Technology **27**(12), 2591–2600 (2016)
48. Wei, J., Lin, G., Yap, K.H., Hung, T.Y., Xie, L.: Multi-path region mining for weakly supervised 3d semantic segmentation on point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4384–4393 (2020)
49. Wei, J., Lin, G., Yap, K.H., Liu, F., Hung, T.Y.: Dense supervision propagation for weakly supervised semantic segmentation on 3d point clouds. arXiv preprint arXiv:2107.11267 (2021)
50. Wu, B., Zhou, X., Zhao, S., Yue, X., Keutzer, K.: Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 4376–4382. IEEE (2019)
51. Wu, T.H., Liu, Y.C., Huang, Y.K., Lee, H.Y., Su, H.T., Huang, P.C., Hsu, W.H.: Redal: Region-based and diversity-aware active learning for point cloud semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15510–15519 (2021)

52. Xie, S., Gu, J., Guo, D., Qi, C.R., Guibas, L., Litany, O.: Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In: European conference on computer vision. pp. 574–591. Springer (2020)
53. Xu, C., Yang, S., Zhai, B., Wu, B., Yue, X., Zhan, W., Vajda, P., Keutzer, K., Tomizuka, M.: Image2point: 3d point-cloud understanding with pretrained 2d convnets. arXiv preprint arXiv:2106.04180 (2021)
54. Xu, K., Yao, Y., Murasaki, K., Ando, S., Sagata, A.: Semantic segmentation of sparsely annotated 3d point clouds by pseudo-labelling. In: 2019 International Conference on 3D Vision (3DV). pp. 463–471. IEEE (2019)
55. Xu, X., Lee, G.H.: Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13706–13715 (2020)
56. Yi, L., Gong, B., Funkhouser, T.: Complete & label: A domain adaptation approach to semantic segmentation of lidar point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15363–15373 (2021)
57. Zhang, Y., Li, Z., Xie, Y., Qu, Y., Li, C., Mei, T.: Weakly supervised semantic segmentation for large-scale point cloud. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 3421–3429 (2021)
58. Zhang, Z., Girdhar, R., Joulin, A., Misra, I.: Self-supervised pretraining of 3d features on any point-cloud. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10252–10263 (2021)
59. Zhu, X., Zhou, H., Wang, T., Hong, F., Ma, Y., Li, W., Li, H., Lin, D.: Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9939–9948 (2021)

Supplementary Material for LiDAL: Inter-frame Uncertainty Based Active Learning for 3D LiDAR Semantic Segmentation

Anonymous ECCV submission

Paper ID 893

Abstract. This supplementary document is organized as follows:

- Section A explains in more detail about the LiDAL implementation.
- Section B describes the baseline active learning methods.
- Section C enumerates detailed semantic segmentation results of the line charts in the main paper.
- Section D provides more ablation studies on the self-training strategy, class distribution of actively selected samples, and pseudo-label accuracy.

A Implementation Details

As explained in Section 3.1 of the main paper, our LiDAL method consists of four steps: 1. Train the network to convergence using the currently labeled dataset D_L. 2. Calculate the model uncertainty scores for each region of D_U. 3. Select regions based on the uncertainty measures for active learning and self-training. 4. Obtain labels from human annotation and pseudo-labeling. In this section, we supplement implementation details to these steps. Note that the used symbols are the same as those in Section 3 of the main paper.

A.1 Network Training

All the experiments are conducted on a PC with 8 NVIDIA Tesla V100 GPUs. The batch sizes are set to 30 and 90 for the SemanticKITTI [1] and nuScenes [4] datasets, respectively.

For both datasets, we train the networks by minimizing the cross-entropy loss using Adam optimizer with an initial learning rate 1e-3. For fully-supervised baselines, the networks are trained for 80,000 iterations. For each round of active learning (including the initial round), the networks are trained or fine-tuned for 20,000 iterations.

The training settings are the same for SPVCNN [39] and MinkowskiNet [6] network architectures.

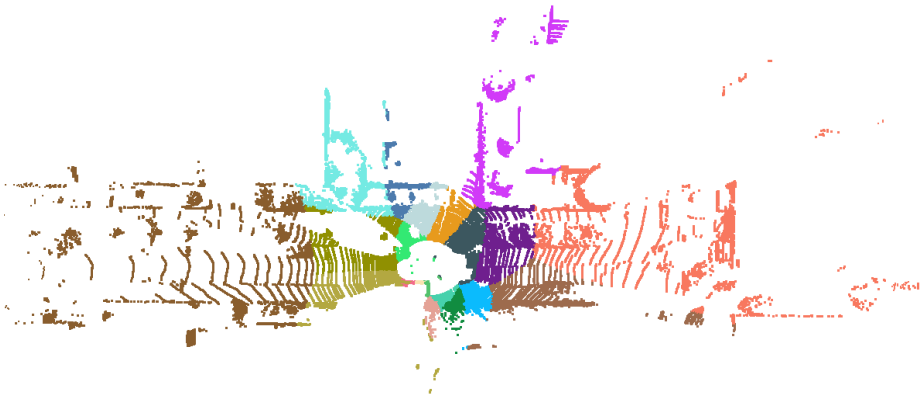

Fig. 1: **An example of divided sub-scene regions in the SemanticKITTI dataset.** Points of the same regions are painted with the same colors.

A.2 Correspondence Estimation

In Section 3.2 of the main paper, after the registration of each frame, we then find for each point its corresponding points in the neighboring frames to calculate inter-frame uncertainty measures. Since there are hundreds of thousands of points in a LiDAR frame, it is impractical to register all the LiDAR frames at the same time and then estimate correspondences for each point.

To address this issue, for each frame F_i, we retrieve its neighboring N_{nei} frames for correspondence estimation. After registration, for each point p of the frame F_i, we find its nearest point in each of the neighboring N_{nei} frames as the initial corresponding points. Since a certain position may be scanned in not all the frames due to occlusion and the movement of the scanning device, point p may not have proper corresponding points in some neighboring frames. We then filter out the corresponding points whose distances to p are larger than a threshold T_p. For both the SemanticKITTI and nuScenes datasets, we set $N_{nei} = 24$ and $T_p = 0.1m$.

A.3 Region Division and Overlap Determining

We utilize the constrained K-means clustering [2] algorithm to divide a LiDAR frame F into multiple sub-scene regions. As an extension of the classical K-means algorithm, this algorithm forces the number of points in each of the K clusters in (N_{min}, N_{max}). For both the SemanticKITTI and nuScenes datasets, we set $K = 20$, $N_{min} = 0.95 * \frac{|F|}{K}$, and $N_{max} = 1.05 * \frac{|F|}{K}$, where $|F|$ is the number of points contained in frame F. An example of divided sub-scene regions of the SemanticKITTI dataset is shown in Fig 1.

In Section 3.3 of the main paper, for a specific region r, we need to retrieve the set of regions overlapping with r for further processing. To determine if two

regions overlap, we may check the Earth Mover's distance [19] or the Chamfer distance [3] between the two regions. However, we find that a simpler solution based on the distance between the weight centers of two regions yields similar results. Considering the efficiency of this simple solution, we determine that two regions overlap if the distance between their weight centers is less than T_r. For both the SemanticKITTI and nuScenes datasets, we set $T_r = 5m$.

A.4 Label Acquisition

For active learning, instead of using a human annotator, we simulate annotation by using the ground-truth annotation of the dataset as the annotation from a human annotator. For self-training, we use the network predictions averaged over 8 augmented inference runs as the pseudo-labels.

B Baseline Active Learning Methods

In this section, we describe the implementation of the baseline active learning methods used in our experiments (Section 4.2 of the main paper).

Random Selection ($\mathbf{RAND_{fr}}$ and $\mathbf{RAND_{re}}$). In each round of active learning, this baseline method randomly selects a portion of LiDAR frames or point cloud regions from the unlabeled dataset for label acquisition. It is a commonly used baseline strategy in the literature [51,37,33,7].

Segment-entropy (SEGENT). Based on the assumption that points within a region are supposed to share the same label, segment-entropy is proposed to serve as a metric for active selection [22]. In this method, the distribution of predicted labels within a region r is estimated by:

$$E_{seg} = -\sum_{c} q(c) \log q(c), \tag{1}$$

$$\hat{y}^p = \arg\max_{c} P^p, \tag{2}$$

$$q(c) = \frac{1}{|r|} \sum_{p \in r} f(\hat{y}^p, c), \tag{3}$$

$$f(\hat{y}^p, c) = \begin{cases} 1, & \text{if } \hat{y}^p = c \\ 0, & \text{otherwise} \end{cases}, \tag{4}$$

where E_{seg} is the proposed segment-entropy, P^p is the probability distribution of point p, $\hat{y}^p$ is the predicted label of point p, and $q(c)$ is the percentage of points predicted as class c. The segment-entropy score of a frame is the average of the scores of all the points inside this frame. The frames with the largest segment-entropy scores are selected for label acquisition. In the implementation of this method, we utilize the same region division results as our LiDAL for a fair comparison.

Softmax Margin (MAR). Some previous active learning methods [17,31,47] rank all the samples in order of the model decision margin, which is the difference of softmax probabilities between the most probable label and the second most probable label, and then select the samples with the least differences. For a point p, the softmax margin is calculated as:

$$MAR^p = P^p(\hat{y}^1) - P^p(\hat{y}^2), \tag{5}$$

where P^p is the probability distribution of point p, $\hat{y}^1$ is the most probable label class, and $\hat{y}^2$ is the second most probable label class.

The softmax margin of a frame is calculated by averaging the values of all the points inside it. The frames with the least softmax margin values are selected for label acquisition.

Softmax Confidence (CONF). Similar to MAR, the softmax probability of the most probable label is considered as a confidence score in some previous methods [34,47]. For point p, the softmax confidence is calculated as:

$$CONF^p = P^p(\hat{y}^1), \tag{6}$$

where P^p is the probability distribution of point p, and $\hat{y}^1$ is the most probable label class.

For a frame, the softmax confidence score is the average result of the scores of all the associated points. The frames with the least confidence scores are selected for label acquisition.

Softmax Entropy (ENT). Unlike MAR and CONF, which consider only the top two most probable classes, softmax entropy takes into account probabilities of all classes to measure the information of a probability distribution [16,47]. For point p, the softmax entropy score is calculated as:

$$ENT^p = -\sum_c P^p(c) \log(P^p(c)), \tag{7}$$

where $P^p(c)$ is the probability of point p belonging to class c.

For a frame, the softmax entropy score is the average result of the scores of all the associated points. The frames with the largest entropy scores are selected for label acquisition.

Core-set Selection (CSET). Core-set refers to a small subset that captures the diversity of the whole dataset [33], and thus a model trained on this subset yields similar performance to that trained on the whole dataset. This method first extracts features for each sample of the dataset using the currently trained network. Operating on the feature space, it then selects a small set of samples for labeling utilizing the furthest point sampling strategy. In the implementation, we use the intermediate results of the second-last layers of the networks as the features. The feature of a frame is averaged over all the associated points.

ReDAL. Region-based and diversity-aware active learning (ReDAL) [51] is a recent state-of-the-art method designed for 3D semantic segmentation of both indoor and outdoor scenes. This method first divides a 3D scene into sub-scene regions and then estimates the region information utilizing three metrics: softmax entropy, color discontinuity, and structural complexity. With the estimated region information scores, this method further designs a diversity-aware selection algorithm to avoid visually similar regions appearing in a querying batch for labeling. Since both the SemanticKITTI and nuScenes datasets do not provide colored point clouds, the color discontinuity metric is discarded in the implementation following the instruction of ReDAL's official code.

C Detailed Experimental Results

In this section, we provide more details on our experimental results, for benchmarking purposes with future works. The results of fully-supervised networks are reported in Table 1. Detailed scores for Fig. 5 in the main paper are shown in Tables 2 and 3. For Fig. 6, the detailed scores are presented in Tables 4 and 5.

Table 1: **Mean intersection over union scores of fully-supervised networks.**

Network \ Dataset	SemanticKITTI	nuScenes
SPVCNN	64.5	71.7
MinkowskiNet	61.4	70.6

Table 2: **Mean intersection over union scores on SemanticKITTI Val with SPVCNN.**

Percentage of Labeled Points	Init (1%)	2%	3%	4%	5%
RAND_{fr}	48.8	52.1	53.6	55.6	57.2
RAND_{re}	48.8	51.7	55.0	56.1	58.2
SEGENT	48.8	49.8	48.3	49.1	48.2
MAR	48.8	49.4	50.0	48.7	49.3
CONF	48.8	48.0	48.9	50.4	51.6
ENT	48.8	49.6	48.5	50.1	49.9
CSET	48.8	53.1	52.9	53.2	52.6
$\text{ReDAL}_{\text{reported}}$	41.9	51.7	55.8	56.9	58.2
$\text{ReDAL}_{\text{retrained}}$	48.8	51.3	54.0	58.6	58.1
LiDAL (ours)	48.8	57.1	58.7	59.3	59.5

Table 3: **Mean intersection over union scores on SemanticKITTI Val with MinkowskiNet.**

Percentage of Labeled Points	Init (1%)	2%	3%	4%	5%
RAND_{fr}	47.3	51.4	55.8	57.7	56.6
RAND_{re}	47.3	50.1	55.8	55.9	58.5
SEGENT	47.3	49.8	48.8	49.5	47.7
MAR	47.3	50.2	49.8	49.4	50.1
CONF	47.3	48.5	48.5	51.4	51.7
ENT	47.3	49.9	48.8	49.0	50.2
CSET	47.3	52.6	55.9	56.4	57.6
$\text{ReDAL}_{\text{reported}}$	37.5	48.9	55.3	58.4	59.8
$\text{ReDAL}_{\text{retrained}}$	47.3	51.4	52.5	58.4	58.1
LiDAL (ours)	47.3	56.7	58.7	59.5	60.1

Table 4: **Mean intersection over union scores on nuScenes Val with SPVCNN.**

Percentage of Labeled Points	Init (1%)	2%	3%	4%	5%
RAND_{fr}	51.8	58.4	60.5	60.6	63.2
RAND_{re}	51.8	60.3	62.3	63.7	63.6
SEGENT	51.8	55.5	56.1	55	57.8
MAR	51.8	55.2	56.4	57.0	57.7
CONF	51.8	55.1	54.9	55.4	56.0
ENT	51.8	55.4	56.7	56.6	57.2
CSET	51.8	59.4	62.3	62.9	63.0
ReDAL	51.8	54.3	57.0	57.2	58.3
LiDAL (ours)	51.8	60.8	65.6	67.6	68.2

Table 5: **Mean intersection over union scores on nuScenes Val with MinkowskiNet.**

Percentage of Labeled Points	Init (1%)	2%	3%	4%	5%
RAND_{fr}	49.7	57.9	60.5	61.8	61.7
RAND_{re}	49.7	58.7	60.9	62.0	63.1
SEGENT	49.7	54.8	55.3	56.5	58.5
MAR	49.7	53.9	55.0	56.7	59.1
CONF	49.7	54.4	55.7	56.8	55.5
ENT	49.7	54.9	56.4	57.2	57.6
CSET	49.7	58.5	62.0	63.2	63.6
ReDAL	49.7	54.5	53.9	56.7	57.2
LiDAL (ours)	49.7	62.3	64.7	66.5	67.0

D More Ablation Studies

In this section, we provide more ablation studies to examine the design decision of our self-training strategy and to analyse the actively selected labels and pseudo-labels.

Self-training Strategy. In Section 3.3 of the main paper, we inject pseudo-labels to the training set at each active learning round to further boost the performance. We have considered three commonly used strategies for self-training:

- **S1:** Enlarge the pseudo-label set in each round with the newly selected regions. (The selection criterion is discussed in the main paper.)
- **S2:** Keep the size of the pseudo-label set constant, and replace in each round with the newly selected regions.
- **S3:** (Our design choice) Keep the size of the pseudo-label set constant, and replace in each round with the newly selected regions that are not already in the last pseudo-label set.

The results of these three self-training strategies on the SemanticKITTI dataset with SPVCNN are shown in Table 6. As shown in the table, both two alternative strategies generate more inferior results to our design choice. We assume that, for **S1**, it is easily susceptible to label drifting as its size of pseudo-label set increases over time. For **S2**, since the previous pseudo-label set used for training is also considered for the pseudo-labeling of the current round, it tends to select a stable set of regions that are less and less helpful during training.

Table 6: **Mean intersection over union scores of different self-training strategies on SemanticKITTI Val with SPVCNN.**

Percentage of Labeled Points	Init (1%)	2%	3%	4%	5%
S1	48.8	56.9	57.2	59.1	58.8
S2	48.8	57.2	58.5	58.9	59.0
S3 (Our design choice)	48.8	57.1	58.7	59.3	59.5

Class Distribution of Actively Selected Samples. To gain a better understanding of the property of inter-frame constraints, we count the class distribution of samples selected in all 4 rounds of LiDAL operating on the SemanticKITTI dataset with SPVCNN network. As shown in Table 7, LiDAL focuses more on less-represented but highly important classes like person and bicyclist. This is foreseeable since the networks struggle to generate consistent predictions for these hard samples. This is a valuable property that can benefit downstream tasks like autonomous driving, which poses great significance on safety issues.

Table 7: **Class distributions of labels(‰).** We present samples selected in all 4 rounds of LiDAL operating on the SemanticKITTI dataset with SPVCNN network.

Method	Total	car	bicycle	motorcycle	truck	other-vehicle	person	bicyclist	motorcyclist	road	parking	sidewalk	other-ground	building	fence	vegetation	trunk	terrain	pole	traffic-sign
Full	10^3	43.68	0.17	0.42	2.02	2.40	0.36	0.13	0.04	205.22	15.19	148.59	4.03	137.00	74.69	275.57	6.23	80.67	2.95	0.63
LiDAL	10^3	36.75	0.25	1.06	4.01	7.45	0.89	0.39	0.07	146.42	22.30	154.42	11.91	127.15	98.29	277.80	9.01	95.91	4.34	1.57

Accuracy of Pseudo-labels. The main challenge with pseudo-labels is to ensure their accuracy and to avoid drifting. In Section 4.4 of the main paper, we evaluate the effects of injecting different numbers of pseudo-labels into the training set. Here we quantitatively measure the accuracy of added pseudo-labels in Table 8. The study is conducted in the first training round of SPVCNN on the SemanticKITTI dataset. As shown in the table, the generated pseudo-labels maintain high accuracy in general, but the accuracy drops when more and more pseudo-labels are selected. This confirms our conjecture in the main paper that adding a reasonable number of pseudo-labels will improve network performance, but redundant pseudo-labels might introduce unhelpful training bias and label noise.

Table 8: **Accuracy of pseudo-labels.** Samples are selected in the first training round of SPVCNN on SemanticKITTI dataset.

Range of Added Pseudo-labels	Mean Accuracy
0-1%	97.58%
1-2%	97.04%
2-3%	93.05%