

Label-Efficient Point Cloud Segmentation with Active Learning

Johannes Meyer^{1*}, Jasper Hoffmann^{1*}, Felix Schulz¹, Dominik Merkle^{2,3}, Daniel Buescher¹, Alexander Reiterer^{2,3}, Joschka Boedecker¹, Wolfram Burgard⁴

Abstract—Semantic segmentation of 3D point cloud data often comes with high annotation costs. Active learning automates the process of selecting which data to annotate, reducing the total amount of annotation needed to achieve satisfactory performance. Recent approaches to active learning for 3D point clouds are often based on sophisticated heuristics for both, splitting point clouds into annotatable regions and selecting the most beneficial for further neural network training. In this work, we propose a novel and easy-to-implement strategy to separate the point cloud into annotatable regions. In our approach, we utilize a 2D grid to subdivide the point cloud into columns. To identify the next data to be annotated, we employ a network ensemble to estimate the uncertainty in the network output. We evaluate our method on the S3DIS dataset, the Toronto-3D dataset, and a large-scale urban 3D point cloud of the city of Freiburg, which we labeled in parts manually. The extensive evaluation shows that our method yields performance on par with, or even better than, complex state-of-the-art methods on all datasets. Furthermore, we provide results suggesting that in the context of point clouds the annotated area can be a more meaningful measure for active learning algorithms than the number of annotated points.

I. INTRODUCTION

Semantic point cloud segmentation is pivotal for many applications including robotics, urban planning, and environmental monitoring. The semantic segmentation of urban point cloud data is particularly important as a basis for wind, water, and heat simulations [1]. This can aid in identifying vulnerable areas within cities, thereby enhancing their resilience to climate change. The simulations require semantic information to differentiate between various surface types, such as sealed or open surfaces, which can impact water seepage. The distinction between fir and leaf trees due to the different capabilities of water storage and leaf fall is important to correctly simulate heavy rain and wind events, and simulate the heat load in cities at different seasons. The size and diversity of cities require a substantial amount of labeled data to sufficiently train state-of-the-art neural networks. Unfortunately, the annotation process for urban 3D point cloud data is especially costly [2], [3]. In practice, to precisely segment an object in 3D requires drawing many different 2D polygons from various perspectives.

* Equal contribution.

¹ Department of Computer Science, University of Freiburg, Germany

² Fraunhofer IPM, Freiburg, Germany

³ Institute for Sustainable Systems Engineering INATECH, University of Freiburg, Germany

⁴ Department of Computer Science and Artificial Intelligence, University of Technology Nuremberg, Germany.

This research was funded by the German Federal Ministry for the Environment, Nature Conservation and Nuclear Safety (BMU) on the basis of a resolution of the German Bundestag as part of the ‘KI-Leuchtturm’ project ‘Intelligence for Cities’ (I4C).

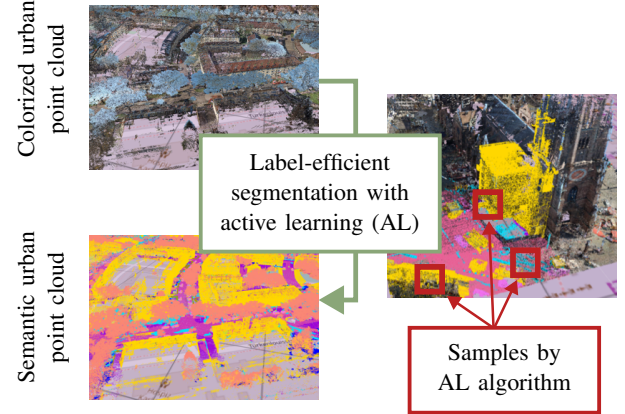


Fig. 1: The goal of this work is to reduce the annotation cost of semantic segmentation for unlabeled urban point clouds. By simplifying existing methods, we aim to reduce the entry barrier to apply active learning for point clouds.

One possible solution is active learning (AL), illustrated in Figure 1. AL can drastically reduce labeling costs by only “requesting” the labels for the most informative unlabeled samples. In practice, the AL algorithm starts to train a model with only a small labelled portion of the data. After each training cycle, the AL algorithm aims to find the most informative part of the data and obtains an annotation from a human or oracle. This process is repeated until the desired performance is reached.

Importantly, the problem of 3D AL is not just about finding individual points in the point cloud, but also finding regions that can be efficiently annotated by a human. Thus, there are two major challenges in AL for 3D, namely region separation: splitting the point cloud into annotatable candidate regions, and region selection: selecting the most beneficial regions to be annotated.

Previous works in 3D AL handled the region separation and region selection step by incorporating sophisticated heuristics. However, these approaches often require cumbersome pre-processing steps and lead to proposals that can be harder to annotate. The approach proposed in this paper separates the point cloud into easy to implement spatial columns and bootstraps the AL pipeline by requiring fewer pre-processing steps. Our proposed AL cycle is illustrated in Figure 2.

Our contributions can be summarized as follows:

- 1) For region separation, we find that our straightforward approach in finding annotatable regions is competitive with respect to state-of-the-art AL methods.

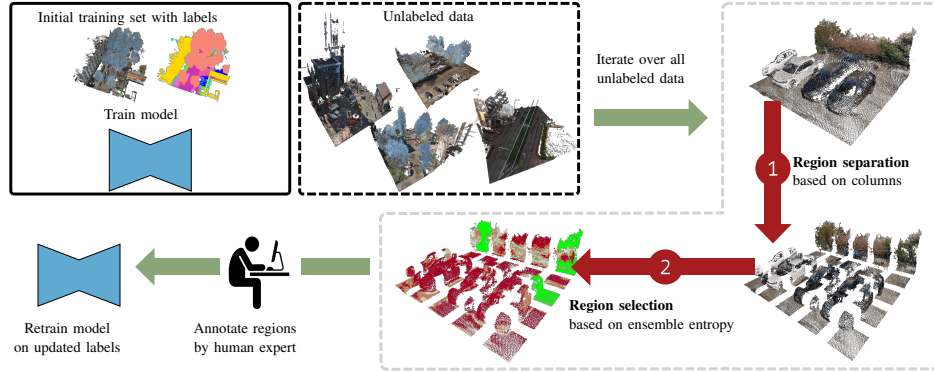


Fig. 2: Our proposed active learning pipeline. The initial dataset consists of unlabeled and labeled parts. The AL algorithm first separates the point cloud into columns and then selects regions with the highest ensemble entropy. These are presented to a human expert for extending the labeled dataset. We iteratively repeat the procedure.

- 2) For region selection, we analyze different common AL metrics based on ensembles, showing better or at least comparable performance when compared with specialized metrics used in current state-of-the-art point cloud segmentation works.
- 3) We show that the number of labeled points can be a misleading measure and propose an alternative metric based on the annotated area.
- 4) We show the applicability of our approach on a large-scale urban 3D dataset of the city of Freiburg.

II. RELATED WORK

To reduce labeling costs of large point cloud datasets, various approaches have been proposed like using scribble annotation to sparsely annotate data [2] or pre-trained models from the 2D image domain [4]. For the related work, we will focus on active learning especially in the context of point cloud segmentation.

A. Active Learning

In the field of AL, a lot of work focuses on the selection of the next batch of data to query a human annotator or an oracle. Most approaches in the field of AL fall into one of three categories [5]. Firstly, approaches that try to maximize the diversity within one batch of queried data. Secondly, approaches that try to select data points that the current model is uncertain about. And thirdly, those which try to estimate which data points will result in the biggest changes to the current learned model.

However, the literature shows that only following one approach of AL is problematic for 2D data [6] as well as for 3D data [7]. Only selecting samples based on the diversity strategy is prone to yield low-information samples which are only selected for diversity reasons and do not contribute valuable information to the training. On the other hand, approaches based on the uncertainty of the network often rely on the softmax probabilities of the network which are known to be overconfident [8]. For that reason, in deep AL often hybrid querying strategies based on the network uncertainty as well as diversity estimates, derived from heuristics, are used [5].

In our work, we investigate whether a pure ensemble

uncertainty-based approach can outperform current state-of-the-art hybrid approaches specialized for point clouds.

B. Active Learning for Point Cloud Segmentation

Several works propose using AL in the context of 3D point cloud segmentation, mostly using a hybrid AL approach, considering uncertainty and diversity for selecting the next batch of data. Importantly, they use specialized heuristics and require additional pre-processing steps. In [7], the authors propose the *Region-based and Diversity-aware Active Learning for Point Cloud Semantic Segmentation* (ReDAL) method. First, the point cloud is separated into regions, called supervoxels, by using an unsupervised segmentation method. For selecting the supervoxel, ReDAL considers heuristics based on color discontinuities between each point and its k -neighbors as well as structural complexity, based on the surface variation. Combined with the softmax entropy of the current semantic segmentation model a region information score is obtained. To ensure diversity, the backbone model features for each point in each supervoxel are averaged, and a k -means clustering across all supervoxels is computed. The information score for the regions is lowered for each region belonging to the same cluster that has a higher information score, ensuring that the information score considers diversity.

A similar hybrid strategy was proposed by Shao et al. [9] and is called Active Learning for point cloud semantic segmentation via Spatial-Structural Diversity Reasoning (SSDR-AL). Similar to ReDAL in SSDR-AL, the point cloud is separated into a set of superpixels, which we call supervoxel for simplicity. The region selection policy operates in two steps: In the first step, proposal supervoxels are sampled based on the average network uncertainty for each supervoxel, combined with a weighting, that based on the currently predicted classes, boosts the probability of underrepresented classes. In the second step, each supervoxel is projected into a so-called diversity space. The projection into the diversity space is done by averaging the features predicted by the backbone model for each point contained in the supervoxel and its surrounding neighboring supervoxels. To select from the regions sampled in the first step, we use a furthest-point sampling in the diversity space.

Samet et al. [4] find that the initial annotated set of regions needed for warm-starting the AL can have a significant influence on the final performance. Generating images from different views of the 3D scene, they use features from a pre-trained DINO model [10] to generate a diverse initial data set, significantly improving the performance of several AL methods. Xu et al. [11] presented a successful combination of AL with self-supervised learning, in which the annotator was queried to annotate individual points, instead of regions. However, as we argue in Section III-C annotating a lot of individual points still can come with a high labeling effort. A sparse point cloud is able to cover a large area of the unlabeled point cloud, even if the percentage of queried points is low, increasing the workload for the human annotator.

III. APPROACH

The two major challenges in AL for point cloud segmentation are region separation, where we separate the point cloud into region proposal regions that can be efficiently annotated, and region selection, where we want to find regions with the highest impact on segmentation performance.

A. Region Separation

In the following, we discuss our region separation mechanism to split the point cloud into 3D columns and also discuss previous state-of-the-art region separation techniques based on supervoxels.

1) Columns

This work employs a straightforward method of dividing the point cloud into easily and efficiently annotated regions, specifically columns. We divide the point cloud into spatial columns of a given grid resolution r using a 2D grid on the XY-plane. This ensures that each query of the AL algorithm does not overburden the human annotator, as the number of points that can be queried is limited by the size of the column. Additionally, each column can be described using straightforward x - y -coordinates, without the need for to store clusters of point indices.

2) Supervoxels

Region-based AL often uses unsupervised segmentation methods to separate the point cloud into coherent regions, called supervoxels. ReDAL [7] uses the Voxel Cloud Connectivity Segmentation (VCCS) method [12], which aims to produce over-segmentation masks that are fully consistent with the spatial geometry within each mask. Alternatives for generating such supervoxels are DBSCAN [13] or HDBSCAN [14]. It is also possible to combine such techniques with a prior ground segmentation as discussed in previous work [15]. However, such approaches sometimes struggle to find coherent supervoxels that can be easily annotated on real-world data. Similarly, in a more recent work [9] the point cloud is also split into supervoxels by using an unsupervised segmentation method based on a global energy model [16].

B. Region Selection

In the following, we discuss different region selection methods. Firstly, we discuss random region selection, which

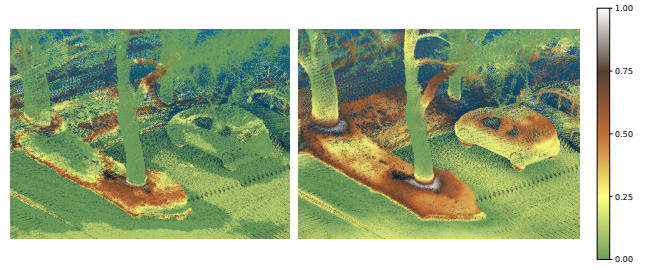


Fig. 3: VaR (left) and ensemble entropy (right) for the Freiburg dataset. Green corresponds to small, and white to large uncertainty. This image shows that both uncertainty metrics indicate a high uncertainty in the areas which are lower-vegetation. In contrast, the entropy indicates a higher uncertainty in the upper parts of the car than the VaR.

is an important AL baseline [5], and previous state-of-the-art region selection techniques that are based on point cloud heuristics. Secondly, we introduce two well established uncertainty metrics that are based on ensembles, the variation ratio and the ensemble entropy. Both are agnostic to point cloud segmentation and can lead to significantly improved uncertainty estimates for AL [5].

1) Random

Random selection of regions is often used as a baseline in the AL community. It serves as an indicator of whether the proposed selection metric is better than a completely uninformed method. Notable, performance estimates of random selection policies often heavily vary between different works, which also has been discussed by Ren et al. [5] and Samet et al. [4].

2) Heuristics-based Approaches

As discussed in Section II-B, current state-of-the-art methods like ReDAL [7] and SSDR-AL [9] are hybrid approaches that use heuristics, such as surface variation or color gradients, to get a reliable diversity estimation. However, since these metrics are often pre-computed they often only serve as a prior to circumvent the problem of overconfident neural networks [8]. Using deep ensembles and averaging the probabilities like in equation (2), can significantly improve the calibration [17], leading to a more meaningful uncertainty metric for AL [5].

3) Variation ratio (VaR)

The variation ratio (VaR) is based on deep ensembles, where N different neural networks are initialized with different parameters but are trained on the same dataset, to select regions for annotation. The variance in the predictions of the ensemble members can be interpreted as the model uncertainty for each point cloud region. This can be used as guidance for which samples, or regions of the point cloud in our case, should be labeled next [18]. For a given point x , the VaR measures the fraction of ensemble members diverging from the majority class as

$$\text{VaR}(x) := 1 - \frac{f_m(x)}{N},$$

where $f_m(x)$ is the frequency of the most selected class for the point x . The lower $f_m(x)$ is, the higher the discrepancy in network predictions of the ensemble and the higher VaR will be. If all members agree on a prediction, the VaR will be 0. To derive a VaR estimate for a given region S , we define the average VaR as

$$\overline{\text{VaR}}(S) := \frac{1}{|S|} \sum_{x \in S} \text{VaR}(x). \quad (1)$$

4) Ensemble entropy (Ent)

Another common metric used in AL based on ensembles is the average entropy [18]. It is derived by the averaged predicted probabilities of the ensemble members, namely

$$\hat{p}(y = c | x) := \frac{1}{N} \sum_{n=1}^N p(y = c | x, \theta_n), \quad (2)$$

where $p(y = c | x, \theta_n)$ denotes the predicted probability of the n th member of the ensemble of predicting class c for an input x . The metric for a single point x , is then defined as the entropy of the average prediction

$$\text{Ent}(x) := - \sum_{c=1}^C \hat{p}(y = c | x) \log(\hat{p}(y = c | x)),$$

where C is the number of classes. Similar to (1), we define then the average entropy for a region S as

$$\overline{\text{Ent}}(S) := \frac{1}{|S|} \sum_{x \in S} \text{Ent}(x). \quad (3)$$

In Figure 3, we show the examples for street-scene computed using the network output for four different random seeds. For this visualization, we use SPVCNN as a segmentation model. Both the VaR and the entropy are normalized between 0 and 1. Both metrics indicate high uncertainty around the edges of the low vegetation areas and for regions in close proximity to other classes. The entropy metric indicates a high uncertainty around the upper part of the vehicle.

C. Measuring Annotation Effort

The main goal of the region separation and selection pipeline in the active learning scheme is to provide proposals for efficient annotation by humans. However, we argue that the expected annotation effort is not well estimated in existing studies: it is measured in terms of the fraction of Lidar points to be annotated [7], [5], [11], [19]. This measure can be very misleading, in particular when the selected points are sparsely scattered. In this case a small fraction of points can cover the surfaces of many objects. Since a human is not efficient in annotating single points, we argue that the covered surface area needs to be considered to estimate the annotation effort.

The performance of the AL pipeline, when using supervoxels calculated by VCCS, measured as function of the fraction of selected points is very competitive. However, as the illustration in Figure 4 shows the supervoxels created by VCCS cover large but sparse areas in the point cloud. In an AL pipeline which utilizes an oracle for label retrieval, such voxels can be efficiently annotated. However, a human might

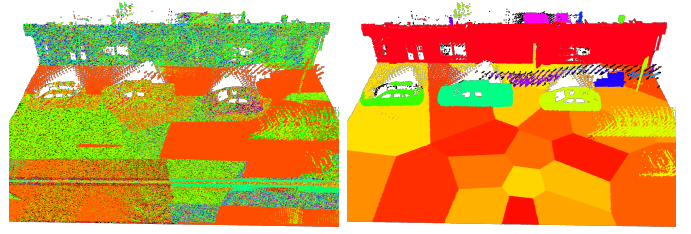


Fig. 4: Region separation on the Toronto data with VCCS (left) and HDBScan (right). Each set of points (supervoxels) is drawn in a different color. The very noisy representation of the clusters on the left depicts the failure of VCCS on this dataset. In contrast on the right the combination of HDBScan with ground-plane removal gives very concise clusters.

not be able to efficiently annotate large sparse point clouds. Despite our best efforts, we found that VCCS, which is the clustering method used in ReDAL, was not able to produce reliable clusters.

In order to compare ourself on Toronto-3D, we propose an alternative supervoxel-based region separation mechanism for that dataset. We segment a ground plane in the data (if available) and cluster the rest with HDBScan. This was also proposed by Nunes *et al.* in the context of contrastive learning [15]. Furthermore, we divide the ground plane into several smaller regions by using K-means clustering. Using this approach we are able to retrieve coherent supervoxels as visualized in Figure 4.

Furthermore, one of our central contributions is the following metric to estimate the annotation effort. As argued above, we base our metric on the area covered by the selected points. An exact estimate of this area is not easy to obtain, but we simplify it by using (half of) the surface A of the cuboid containing the selected points, where $A = \Delta x \Delta y + \Delta x \Delta z + \Delta y \Delta z$. Here, each $\Delta \zeta$ is defined by the extend of the selected cluster along the ζ -axis. This measure will grow with increased sparsity of the selection points, and we think it is a more accurate way of representing the amount of work required for human annotators.

IV. EXPERIMENTS

In the following, we lay out the experimental part of this work starting with the different metrics used, datasets, results and ablation studies.

A. Metrics

mIoU The mean Intersection-over-Union (mIoU) metric is commonly used to quantify the accuracy of semantic segmentation masks. It provides an average overlap between the predicted and ground-truth point labels.

mIoU@90 The mIoU@90 is used in the AL community as the target performance. It is 90% of the mIoU performance that can be achieved using supervised training on the whole dataset.

Area We propose an alternative measure to approximate the annotation effort of a queried region. It is defined as the sum over the cluster areas as described in Section III-C.

Dataset	Points	Semantics	Classes
S3DIS	273,546,486	100 %	13
Toronto-3D	78,320,210	100 %	9
Freiburg	57,995,691,249	~ 0.15 %	13

TABLE I: Overview of the datasets.

B. Datasets

We use the Stanford 3D Indoor Scene Dataset (S3DIS) [20] and Toronto-3D [21] to evaluate our approach. Additionally, we show the applicability of our approach on a private dataset from the city of Freiburg, Germany. All datasets include semantic labels and colorized points. We utilize the latter together with the XYZ locations for classification. Table I summarizes the datasets.

S3DIS dataset [20], is a large indoor point cloud dataset. It is divided into six large areas and has a total of 271 rooms. For each room, a dense point cloud with color and position information is provided. We use the 'Area5' validation set for all our performance evaluations.

Toronto-3D dataset [21], is a large-scale urban outdoor point cloud dataset from Toronto, Canada, covering 1 km of road.

Freiburg dataset [22], is a private dataset from the city of Freiburg. The LiDAR information is accompanied by RGB and intensity information. It covers about 78 km² with a spatial resolution in the centimeter range. This dataset does not come with semantic labels, instead, we manually annotated a small fraction, amounting to about 42,500 m². The intended usage of the data for environmental climate modeling [1] motivates our set classes: building, wall, car, cobblestone surface, street, leaf tree, fir tree, grass, open soil, bush, hedge, vegetation, and unknown. The annotation took about 85 working hours plus additional time for quality assurance, including review and correction. This illustrates the importance of AL in reducing the amount of manual labeling. For this dataset, the percentages of labeled data in the following sections are given with respect to the annotated subset reported in Table I.

C. Network Architecture, Hyperparameters and Codebase

We evaluate all AL methods and region separation and selection strategies on two different network architectures: SPVCNN [23] and the Minkunet [24]. For comparability, we extended the codebase of [7]. Similar to them we use the Adam optimizer with a learning rate of 0.001. We use a batch size per GPU of 4 and an ensemble size of $N = 4$ (where applicable). Each model was trained on a single NVIDIA RTX A6000 GPU.

D. Results

1) Performance with respect to covered area

We first present our results in terms of area which is required to be labeled, which we believe is a more representative estimate of the labelling effort. It is apparent in Figure 5a and Figure 5b that our column-based separation requires way less annotated area (more than an order of magnitude) compared to VCCS region separation, and still about a factor of two less area than HDBScan, for a similar mIoU.

For the indoor dataset S3DIS both region separation techniques show a comparable performance as function of the annotated area, as shown in Figure 5c. In this setting the VCCS based approaches are slightly better. We suspect that this is mostly caused by the fact that the columns always include floor and ceiling. For that reason, in indoor environments the annotated area is inflated for our approach, which could easily be disregarded by a human annotator by not considering the ceiling. Despite this inflation, we can see that after the first AL cycles both approaches are competitive in terms of required annotated area.

In order to make our analysis comparable to other works in the community of AL, we will present the forthcoming results in terms of the percentage of labeled data.

2) Region separation

As first component, we evaluate the region separation mechanism. For a fair comparison, we use the ReDAL region selection but vary only the method by which the point cloud is partitioned into regions. The complete results are shown in Figure 6. Here, we use the ReDAL region selection policy for comparability and only change how we split the point cloud into regions. While the results for the S3DIS dataset are fairly consistent between the two network architectures, indicating low variance, there is more variation found in the Freiburg results. Hence, we will focus on the S3DIS (and Toronto) datasets for our interpretation, while the Freiburg results are deemed for validation.

We observe that columns with an edge length of 0.5 meters are competitive with the supervoxels used by ReDAL. These results show that the performance of ReDAL is not bound to the burden of supervoxel computation. Competitive or even better performance can be achieved by using our column-based approach. The granularity of the regions seems to be an important factor in the performance, since with a larger edge length of 3.0 meters the selected columns are not competitive anymore.

3) Region selection

As second component, we evaluate the region selection mechanism. In this part of the evaluation, we fix the region separation mechanism to supervoxel or columns with an edge length of 0.5 meters as they performed best with the ReDAL algorithm. We compare our ensemble-based entropy and VaR metrics, against ReDAL and the random policy. The complete results are shown in Figure 7.

For the S3DIS dataset (Figure 7a), we can see that all approaches show similar performance for the range of 3 – 7 % labeled data. However, in the later AL stages, the ensemble-based methods outperform ReDAL as well as the random selection policies. We find that the entropy-based selection of columns with SPVCNN is able to reach the mIoU@90 threshold with 11.3 % of labeled points. This also outperforms SSDR-AL [9] and other methods as shown in Table II, and highlights the capabilities of our proposed pipeline. Comparing our results with the original ones from ReDAL [7], we find some discrepancies. Our version of ReDAL, using the original

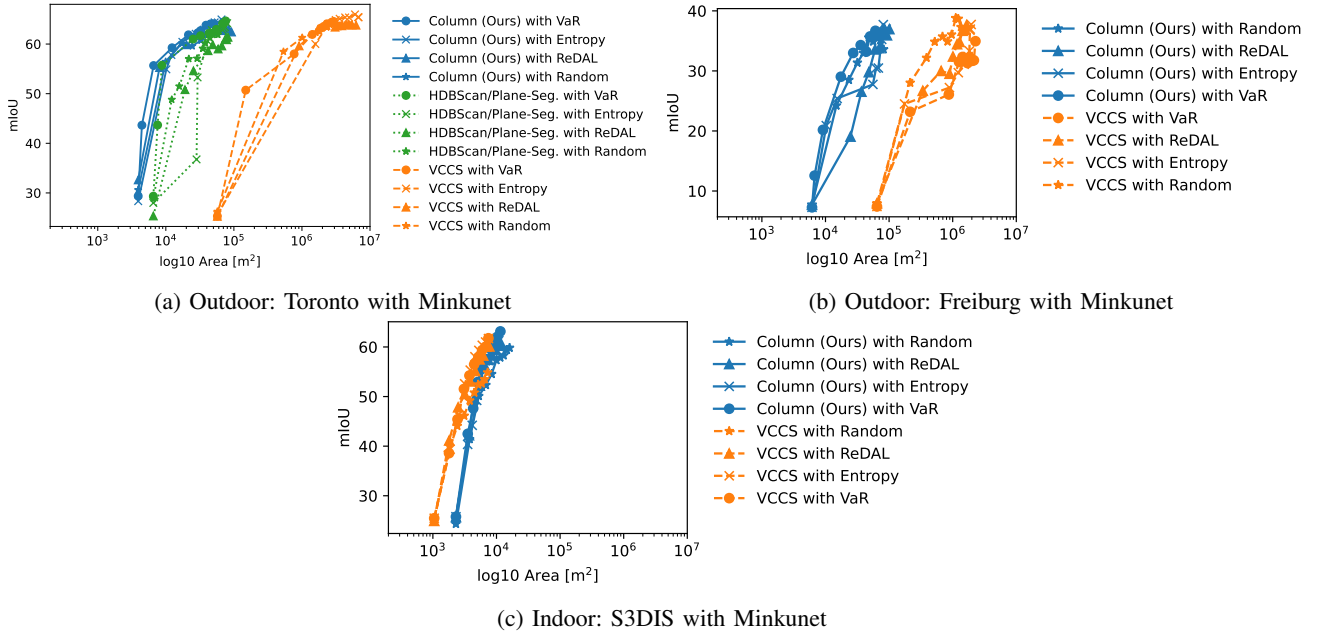


Fig. 5: Performance as a function of the annotated area for all datasets. The blue lines correspond to our proposed column separation, the orange line correspond to the region separation with VCCS and the green line correspond to the HDBScan-based separation.

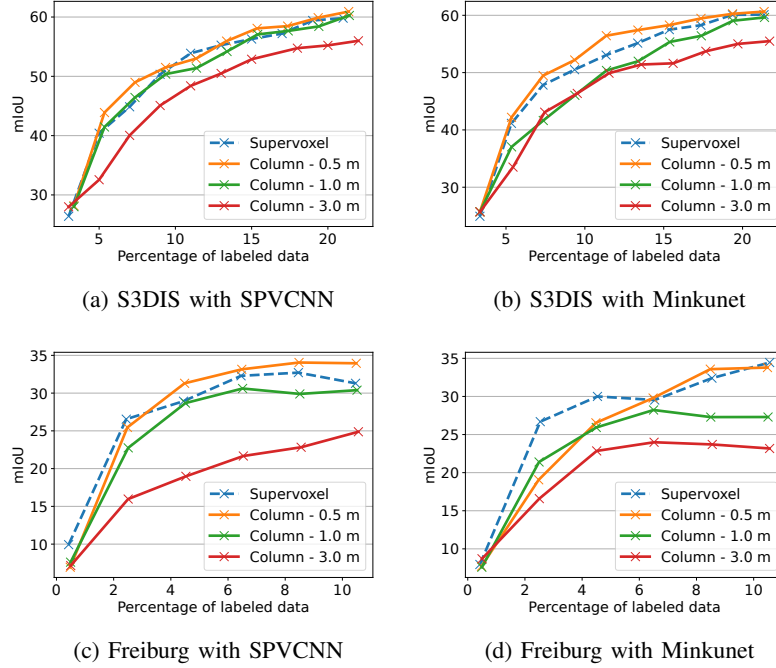


Fig. 6: Performance for the region separation methods supervoxels, created with VCCS, and columns with edge lengths of 0.5, 1.0, and 3.0 meters with ReDAL as region selection algorithm. We show results for the S3DIS and Freiburg datasets with SPVCNN or Minkunet as segmentation models.

codebase, crosses the mIoU@90 threshold at about 19% annotated data, while the authors report 13 to 15%.

This might be attributed to the variance in the results, e.g. from random seeds. The mIoU@90 is particularly sensitive to this effect if the performance saturates around this value, which we observe on the Toronto-3D dataset. On Toronto-3D, we estimate the standard deviation from 5 random seeds on the mIoU to amount to about $\pm 1.4\%$ in shoulder region

of the curves (2-4% labeled points) and to about $\pm 0.5\%$ in the saturation region ($\geq 10\%$ labeled points). Hence, the mIoU@90 value differs by up to $\pm 3\%$ of labeled points. These findings sit well with the observations in Samet et al. [10] which investigate the differences in seed selection for AL.

On the Toronto-3D data with SPVCNN, the column-based region separation with Entropy and VaR reaches the mIoU@90 threshold with 12% and 14% of annotated data, respectively.

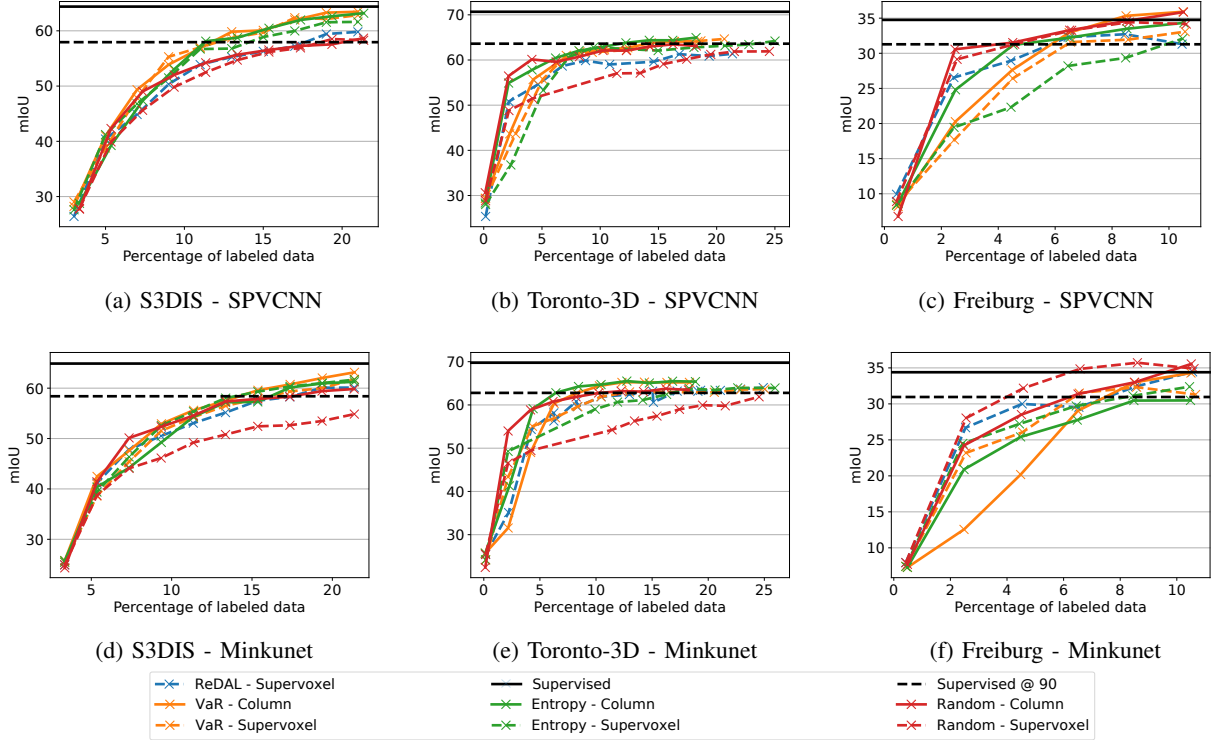


Fig. 7: Performance for various region separation (supervoxels and columns of size 0.5 m) and selection methods (ReDAL, VaR, ensemble entropy, and random). We show results for the S3DIS, Toronto, and Freiburg datasets with SPVCNN or Minkunet as segmentation models. We also report the performance for supervised learning on all labeled data, as well as 90 % of that value (mIoU@90 %).

In	Methods	Points	
		SPVCNN [25]	Minkunet [24]
Our	Columns + Random	21.4 %	19.4 %
	Columns + Entropy	11.4 %	17.4 %
	Columns + VaR	13.0 %	15.4 %
	ReDAL [7]	19.0 %	19.3 %
[7]	ReDAL [7]	13 %	15 %
[9]	Random [9]		40.9 %
	Entropy [26]		46.7 %
	BvSB [26]		43.0 %
	ClassBal [27]		13.3 %
	SSDR-AL [9]		11.7 %

TABLE II: Amount of points required for 90 % of the supervised training performance. Our results are compared to the ones reported in [7] and [9]. The latter uses the RandLA-Net [28] segmentation model. All results are evaluated on the S3DIS dataset.

The supervoxel-based counterparts require 16 % and 25 %, respectively, while random selection and ReDAL are unable to reach the threshold within ten AL cycles. The results with Minkunet show a similar overall picture.

In regards to the Freiburg data, though large variances in the results are observed, it is evident that the random policy outperforms the informed region selection methods. We attribute this to the small overall fraction of labels in the data being more widespread, covering a relatively large area. This is in contrast to Toronto-3D, which focuses on a single street. Therefore, we conclude that the label set remains noisy, which also explains the network’s ability to outperform the fully

supervised baseline on a smaller data budget.

It should be noted that, across most datasets and segmentation models, the random selection policy also performs very well. At first glance, this is in contradiction to some works that report very poor performances of random selection policies [7], [9]. However, often these poor scores stem from random point-selection policies instead of random region-selection policies.

In summary, our results demonstrate that our easy-to-implement AL pipeline using spatial columns as a region separation mechanism and ensemble-based region selection policies are competitive with, or better than, state-of-the-art approaches.

E. Ablation

In the following, we show that our approach is computationally cheaper and further analyze which augmentations are the most important during training. This is especially important when scaling to very large datasets.

1) Preprocessing time

In Figure 8 we compare the preprocessing time required for our column-based region separation technique and the more heuristic based method of ReDAL. In comparison all preprocessing steps individually take longer than the computation of the column-based separation. For a pointcloud with $50 \times 50\text{m}$ the computation of column separation takes ~ 0.3 sec, which would also allow for online computation, but not used for a fair comparison of the methods. Limiting the pre-processing to the

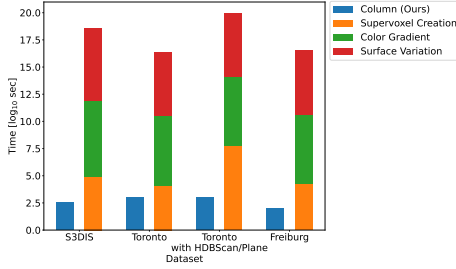


Fig. 8: Comparison of preprocessing time required for our method against the preprocessing time required by ReDAL. For a fair comparison, we use the same framework as ReDAL and precompute the column-based region separation beforehand. Note, that we report log sec as duration of individual steps for a better visual comparison. Time measurement done on a machine with Ryzen 9 pro 7945 and 64 GB RAM.

S/R/E/C	2 % labels		10 % labels		100 % labels	
	mIoU	Time	mIoU	Time	mIoU	Time
× × × ×	23.32	16.02	32.77	21.67	43.68	159.00
✓ × × ×	24.48	×0.95	38.03	×1.05	48.71	×1.38
× ✓ × ×	27.55	×1.16	43.23	×1.48	58.79	×1.76
× × ✓ ×	21.96	×1.94	36.83	×2.56	45.95	×3.33
× × × ✓	22.86	×1.03	32.77	×1.20	44.22	×1.60
✓ ✓ ✓ ✓	26.05	×2.37	41.76	×3.34	61.13	×5.92

TABLE III: Performance on the S3DIS dataset using data augmentation techniques: scale (S), rotation (R), elastic (E), and chromatic (C). The training time is given in minutes, or as scale factor in proportion to the first row.

preprocessing of spatial columns reduces the pre-processing time on S3DIS to 0.59% of the otherwise required time. On Toronto-3D to 1.9% or 0.64% with HDBScan and on the annotated portion of the Freiburg data to 0.70%.

2) Data Augmentation

is one of the key techniques to diversify the training data in order to achieve better training results. However, it is often very domain-dependent on which data augmentation techniques perform the best. As methods like RandAugment [29] or TrivialAugment [30] have shown, it is often the mixture of different augmentations that works the best. To improve the AL cycle in terms of performance and required time, we investigate the influence of the individual augmentation methods used in ReDAL [7].

Table III shows the influence of the different data augmentation methods for 2%, 10%, and 100% percent of data files used. Note, that the sub-100% data points are randomly sampled and not selected by any method. From the results, one can observe that generating novel views of the data through rotation is by far the most important augmentation method. In the lower-data regime, it outperforms the combination of all other data augmentation schemes. However, when training with all data, training with all augmentations is still the best.

The results show that rotation augmentation is by far the most important augmentation for the performance of the network. For the low-label regime, with 2% or 10% of

scenes sampled, it is often beneficial to only use the rotation augmentation both in terms of performance and used training time. When training with all data, it is beneficial to train with all augmentations. However, it is also the slowest training. Hence, we chose to deactivate the elastic distortion during the AL cycles of our method but to enable all other augmentations.

V. CONCLUSIONS

In this paper, we presented a novel active learning pipeline in the context of point cloud segmentation that provides comparable or better performance than state-of-the-art results, while employing easy-to-implement methods. We evaluated our approach in the context of large-scale urban point clouds, with classes directed at forecasting extreme weather events, but also on the common S3DIS indoor dataset.

In terms of region separation, we proposed to divide the point cloud into a 2D grid of columns. Columns can be easily optimized due to having a single parameter (the edge length) and efficiently stored without the need for point indices. Furthermore, columns are robust under domain changes, while we observed that the more involved VCCS method can fail. Concerning the region selection step, we propose to use common ensemble uncertainty metrics, with better or equally good results as more involved hybrid approaches. This reduces the number of cumbersome preprocessing steps.

Additionally, we proposed a novel metric to determine the annotation costs of different active learning approaches. This metric not only takes into account the number of points to be annotated but also the area that needs to be considered by the human annotator during the labeling process. As a result, we estimate that our active learning approach for point cloud data requires less work from human annotators.

Despite these encouraging results, there are several aspects that warrant future research. First, one could investigate whether foundation models can be employed to replace the human annotator. Second, one could utilize a location-dependent adjustment of the grid resolution.

Acknowledgements

We acknowledge the city of Freiburg for providing us with the colored LiDAR data from the city of Freiburg. This research was funded by the German Federal Ministry for the Environment, Nature Conservation and Nuclear Safety (BMU) on the basis of a resolution of the German Bundestag as part of the ‘KI-Leuchtturm’ project ‘Intelligence for Cities’ (I4C).

REFERENCES

- [1] F. Briegel, O. Makansi, A. Matzarakis, T. Brox, and A. Christen, “Modelling long-term thermal comfort conditions in urban environments using a deep convolutional encoder-decoder as a computational shortcut,” *Urban Climate*, vol. 47, Jan 2023.
- [2] O. Unal, D. Dai, and L. Van Gool, “Scribble-supervised lidar semantic segmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 2697–2707.
- [3] D. Merkle and A. Reiterer, “Overview of 3D point cloud annotation and segmentation techniques for smart city applications,” in *Remote Sensing Technologies and Applications in Urban Environments VII*, T. Erbertseder, N. Chrysoulakis, and Y. Zhang, Eds., vol. 12269, International Society for Optics and Photonics. SPIE, 2022, p. 1226903.

- [4] N. Samet, O. Siméoni, G. Puy, G. Ponimatkin, R. Marlet, and V. Lepetit, “You Never Get a Second Chance To Make a Good First Impression: Seeding Active Learning for 3D Semantic Segmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 18 445–18 457.
- [5] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang, “A survey of deep active learning,” *ACM computing surveys (CSUR)*, vol. 54, no. 9, pp. 1–40, 2021.
- [6] G. Wang, J.-N. Hwang, C. Rose, and F. Wallace, “Uncertainty sampling based active learning with diversity constraint by sparse selection,” in *IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, 2017, pp. 1–6.
- [7] T.-H. Wu, Y.-C. Liu, Y.-K. Huang, H.-Y. Lee, H.-T. Su, P.-C. Huang, and W. H. Hsu, “ReDAL: Region-based and diversity-aware active learning for point cloud semantic segmentation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2021, pp. 15 490–15 499.
- [8] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, “On calibration of modern neural networks,” in *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR, 2017, pp. 1321–1330.
- [9] F. Shao, Y. Luo, P. Liu, J. Chen, Y. Yang, Y. Lu, and J. Xiao, “Active learning for point cloud semantic segmentation via spatial-structural diversity reasoning,” *arXiv preprint arXiv:2202.12588*, 2022.
- [10] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, “Emerging properties in self-supervised vision transformers,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.
- [11] Z. Xu, B. Yuan, S. Zhao, Q. Zhang, and X. Gao, “Hierarchical Point-based Active Learning for Semi-supervised Point Cloud Semantic Segmentation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 18 098–18 108.
- [12] J. Papon, A. Abramov, M. Schoeler, and F. Wörgötter, “Voxel cloud connectivity segmentation-supervoxels for point clouds,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 2027–2034.
- [13] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al., “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *kdd*, vol. 96, no. 34, 1996, pp. 226–231.
- [14] R. J. Campello, D. Moulavi, and J. Sander, “Density-based clustering based on hierarchical density estimates,” in *Pacific-Asia conference on knowledge discovery and data mining*. Springer, 2013, pp. 160–172.
- [15] L. Nunes, R. Marcuzzi, X. Chen, J. Behley, and C. Stachniss, “SegContrast: 3D Point Cloud Feature Representation Learning Through Self-Supervised Segment Discrimination,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2116–2123, 2022.
- [16] S. Guinard and L. Landrieu, “Weakly supervised segmentation-aided classification of urban scenes from 3d lidar point clouds,” in *ISPRS Workshop 2017*, 2017.
- [17] B. Lakshminarayanan, A. Pritzel, and C. Blundell, “Simple and scalable predictive uncertainty estimation using deep ensembles,” *Advances in neural information processing systems*, vol. 30, 2017.
- [18] W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler, “The power of ensembles for active learning in image classification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 9368–9377.
- [19] F. Shao, Y. Luo, P. Liu, J. Chen, Y. Yang, Y. Lu, and J. Xiao, “Active Learning for Point Cloud Semantic Segmentation via Spatial-Structural Diversity Reasoning.” [Online]. Available: <http://arxiv.org/abs/2202.12588>
- [20] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese, “3d semantic parsing of large-scale indoor spaces,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [21] W. Tan, N. Qin, L. Ma, Y. Li, J. Du, G. Cai, K. Yang, and J. Li, “Toronto-3D: A large-scale mobile lidar dataset for semantic segmentation of urban roadways,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition workshops*, 2020, pp. 202–203.
- [22] Stadt Freiburg, “Fotofahrten für das Vermessungsamt,” *Amtsblatt Stadt Freiburg: M 8334 D – Nr. 763 – Jahrgang 33*, vol. 763, p. 9, Mar. 2020.
- [23] L. Tang, Y. Zhan, Z. Chen, B. Yu, and D. Tao, “Contrastive boundary learning for point cloud segmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 8489–8499.
- [24] C. Choy, J. Gwak, and S. Savarese, “4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3075–3084.
- [25] H. Tang, Z. Liu, S. Zhao, Y. Lin, J. Lin, H. Wang, and S. Han, “Searching efficient 3d architectures with sparse point-voxel convolution,” in *Proceedings of European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 685–702.
- [26] A. J. Joshi, F. Porikli, and N. Papanikolopoulos, “Multi-class active learning for image classification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 2372–2379.
- [27] L. Cai, X. Xu, J. H. Liew, and C. S. Foo, “Revisiting superpixels for active learning in semantic segmentation with realistic annotation costs,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 10 988–10 997.
- [28] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, and A. Markham, “Randla-net: Efficient semantic segmentation of large-scale point clouds,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 108–11 117.
- [29] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, “Randaugment: Practical automated data augmentation with a reduced search space,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition workshops*, 2020, pp. 702–703.
- [30] S. G. Müller and F. Hutter, “Trivialaugment: Tuning-free yet state-of-the-art data augmentation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 774–782.