SQN: Weakly-Supervised Semantic Segmentation of Large-Scale 3D Point Clouds with 1000× Fewer Labels

Qingyong Hu[1], Bo Yang[2*], Guangchi Fang[3], Yulan Guo[3], Ales Leonardis[4],
Niki Trigoni[1], Andrew Markham[1]
[1]University of Oxford, [2]The Hong Kong Polytechnic University,
[3]Sun Yat-sen University, [4]Huawei Noah's Ark Lab
qingyong.hu@cs.ox.ac.uk, bo.yang@polyu.edu.hk, andrew.markham@cs.ox.ac.uk

Abstract

We study the problem of labelling effort for semantic segmentation of large-scale 3D point clouds. Existing works usually rely on densely annotated point-level semantic labels to provide supervision for network training. However, in real-world scenarios that contain billions of points, it is impractical and extremely costly to manually annotate every single point. In this paper, we first investigate whether dense 3D labels are truly required for learning meaningful semantic representations. Interestingly, we find that the segmentation performance of existing works only drops slightly given as few as 1% of the annotations. However, beyond this point (e.g., 1‰ and below) existing techniques fail catastrophically. To this end, we propose a new weak supervision method to implicitly augment the total amount of available supervision signals, by leveraging the semantic similarity between neighboring points. Extensive experiments demonstrate that the proposed Semantic Query Network (SQN) achieves state-of-the-art performance on six large-scale open datasets under weak supervision schemes, while requiring only 1‰ labeled points for training. The code is available at https://github.com/QingyongHu/SQN.

1. Introduction

Learning the precise semantic meanings of large-scale point clouds is crucial for intelligent machines to truly understand complex 3D scenes in the real world. This is a key enabler for autonomous vehicles, augmented reality devices, etc., to quickly interpret the surrounding environment for better navigation and planning.

With the availability of large amounts of labeled 3D data for fully-supervised learning, the task of 3D semantic segmentation has made significant progress in the past four years. Following the seminal works PointNet [41] and SparseConv [13], a series of sophisticated neural architecture [42, 31, 9, 22, 60, 34, 86, 8] have been proposed in the literature, greatly improving the accuracy and efficiency of semantic estimation on raw point clouds. The performance of these fully-supervised methods can be further boosted with the aid of self-supervised pre-training representation learning in recent studies [73, 33, 64, 7, 81, 58]. The success of these approaches primarily relies on densely annotated per-point semantic labels to train the deep neural networks. However, it is extremely costly to fully annotate 3D point clouds due to the unordered, unstructured, and non-uniform data format (*e.g.*, over 1700 person-hours to annotate a typical dataset [3] and around 22.3 minutes for a single indoor scene (5m×5m×2m) [12]). In fact, for very large-scale scenarios *e.g.*, an entire city, it becomes infeasible to manually

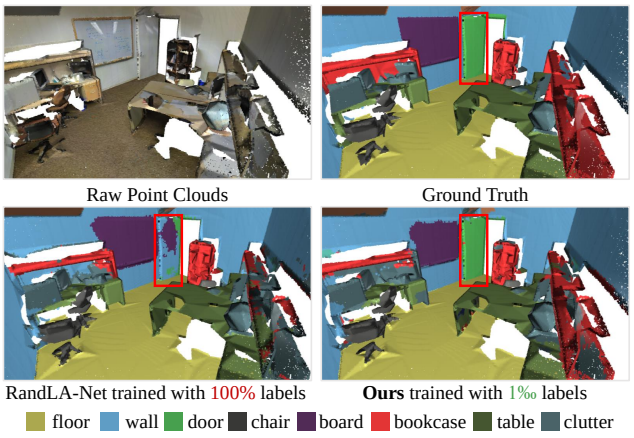

Figure 1: Qualitative results of RandLA-Net [22] and our SQN on the S3DIS dataset. Trained with only 1‰ annotations, SQN achieves comparable or even better results than the fully-supervised RandLA-Net. Red bounding boxes highlight the superior segmentation accuracy of our SQN.

*Corresponding author

label every point in practice.

Being inspired by the success of weakly-supervised learning techniques in 2D images, a few recent works have started to tackle 3D semantic segmentation using fewer point labels to train neural networks. These methods can be generally divided into four categories: 1) using 2D image labels for training [65]; 2) approximating gradients with fewer 3D labels [75]; 3) generating pseudo labels from limited annotations [54, 69]; 4) contrastive pretraining followed by fine-tuning with fewer labels [20, 73]. Although they achieve encouraging results on multiple datasets, there are a number of limitations still to be resolved.

Firstly, existing approaches usually use custom ways to annotate different amounts of data for training and evaluation, and thus it is unclear what proportion of raw points should be annotated, such that a fair comparison is impossible. **Secondly**, to fully utilize the scarce annotations, these pipelines usually involve multiple stages including careful data augmentation, self-pretraining, fine-tuning, and/or post-processing such as the use of dense CRF [25]. As a consequence, it tends to be more difficult to tune the parameters and deploy them in practical applications, compared with the standard end-to-end training scheme. **Thirdly**, these techniques do not adequately consider the semantic properties of point neighbors for large-scale scenarios, or do so ineffectively, resulting in the limited yet valuable annotations being under-exploited.

Motivated by these critical issues, this paper proposes a new paradigm for weakly-supervised semantic segmentation on large-scale point clouds, addressing the above shortcomings. In particular, we first set up a benchmark for weak-supervision schemes purely based on existing fully-supervised methods, and then introduce an effective approach to learn accurate semantics given extremely limited point annotations.

To setup our benchmark, we take into account two key questions: 1) whether, and how, do existing fully-supervised methods deteriorate given different amounts of annotated data for training? 2) given fewer and fewer labels, is there a critical point beyond which existing approaches completely collapse? Fundamentally, by doing so, we aim to explore the limit of current fully-supervised methods. This allows us to draw insights on the use of mature architectures when addressing this challenging task, instead of naïvely borrowing off-the-shelf techniques developed in 2D images. Interestingly, our extensive experiments in Section 3 show that the established fully-supervised methods excel at learning high-quality semantics given as few as 1% annotations, without integrating any extra modules. Unfortunately, however, they catastrophically fail when there are less than 1‰ annotations for training.

With this insight, we further propose a novel yet intuitively simple **S**emantic **Q**uery **N**etwork, named **SQN**, for semantic segmentation given as few as 1‰ labeled points for training. Our SQN firstly encodes the entire raw point cloud into a set of hierarchical latent representations via an existing feature extractor, and then takes an arbitrary 3D point position as input to query a subset of latent representations within a reasonable neighborhood. These queried representations are summarized into a compact vector and then fed into a series of multilayer perceptrons (MLPs) to predict the final semantic label. Fundamentally, our SQN explicitly and effectively considers the semantic similarity between neighboring 3D points, allowing the extremely sparse training signals to be back-propagated to a much wider spatial context, thereby achieving superior performance under weak supervision.

Overall, this paper takes the initiative to bridge the successful fully-supervised methods to the emerging weakly-supervised schemes. However, unlike the existing weak-supervision methods, our SQN does not require any self-supervised pretraining, hand-crafted constraints, or complicated post-processing steps, while obtaining up to fully-supervised accuracy using as few as 1‰ training labels on multiple large-scale open datasets. Figure 1 shows the qualitative results of our method. Our key contributions are:

- We present a weak-supervision benchmark using existing fully-supervised methods, identifying that dense 3D annotations are actually redundant.
- We propose a new weakly supervised method that leverages a point neighbourhood query to fully utilize the sparse training signals.
- We demonstrate a significant improvement over baselines in our benchmark, and surpass the state-of-the-art weak-supervision methods by large margins.

2. Related Work

2.1. Learning with Full Supervision

End-to-End Full Supervision. With the availability of densely-annotated point cloud datasets [21, 2, 15, 3, 46, 62, 52], deep learning-based approaches have achieved unprecedented development in semantic segmentation in recent years. The majority of existing approaches follow the standard end-to-end training strategy. They can be roughly divided into three categories according to the representation of 3D point clouds [14]: **1) Voxel-based methods.** They [8, 76, 13, 38] usually voxelize the irregular 3D point clouds into regular cubes [57, 9], cylinders [86], or spheres [30]. **2) 2D Projection-based methods.** This pipeline projects the unstructured 3D points into 2D images through multi-view [6, 26], bird-eye-view [1], or spherical projections [39, 11, 70, 71, 74], and then uses the mature 2D architectures [35, 19] for semantic learning. **3) Point-based methods.** These methods [22, 41, 42, 60, 31, 72, 84] directly op-

erate on raw point clouds using shared MLPs. Hybrid representations, such as point-voxel representation [53, 34, 43], 2D-3D representation [79, 24], are also studied.

Self-supervised Pretraining + Full Finetuning. Inspired by the success of self-supervised pre-training representation learning in 2D images [7, 18], a number of very recent studies [73, 33, 64, 81, 58, 47] apply contrastive techniques for 3D semantic segmentation. These methods usually pretrain the networks on additional 3D source datasets to learn initial per-point representations via self-supervised contrastive losses, after which the networks are carefully finetuned on the target datasets with full labels. This noticeably improves the overall accuracy.

Although these methods have achieved remarkable results on existing datasets, they rely on a large amount of labeled data for training, which is costly and prohibitive in real applications. By contrast, this paper aims to learn semantics from a small fraction of annotations, which is cheaper and more realistic in practice.

2.2. Learning with Weak Supervision

Limited Indirect Annotations. Instead of having some point-level semantic annotations, only sub-cloud level or seg-level labels are available. Wei et al. [69] firstly train a classifier with sub-cloud labels, and then generate point-level pseudo labels using class activation mapping technique [85] in 2D images. Tao et al. [54] present a segment grouping network to learn semantic and instance segmentation of 3D point clouds, with the seg-level labels generated by over-segmentation pre-processing.

Limited Point Annotations. Given a small fraction of points with accurate semantic labels for training, Xu and Lee [75] propose a weakly supervised point cloud segmentation method by approximating gradients and using handcrafted spatial and color smoothness constraints. It achieves comparable performance with the fully-supervised counterpart, whilst only requiring a tenth of the labels. Shi et al. [49] further investigate label-efficient learning by introducing a super-point-based active learning strategy. In addition, self-supervised pre-training methods [48, 73, 20, 81, 33] are also flexible to fine-tune the networks on limited annotations.

Our SQN is designed for limited point annotations which we believe has greater potential in practical applications. It does not require any pre-training, post-processing, or active labelling strategies, while achieving similar or even higher performance than the fully-supervised counterpart with only 1‰ randomly annotated points for training.

2.3. Learning with Self-Supervision

Saudar and Sievers [47] learn the point semantics by recovering the correct voxel position of every 3D point after the point cloud is randomly shuffled. Sun et al. propose Canonical Capsules [51] to decompose point clouds into object parts and elements via self-canonicalization and auto-encoding. Although they have obtained promising results, they are limited to simple objects and cannot process the complex large-scale point clouds.

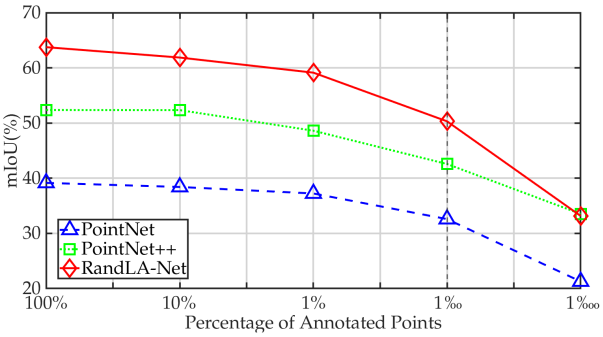

Figure 2: Benchmark results of three baselines in the *Area-5* of the S3DIS [2] dataset. Different amount of points are randomly annotated for weak supervision. (The horizontal axis uses a logarithmic scale).

3. Benchmarking Weak Supervision

Since weakly-supervised 3D semantic segmentation is still in its infancy, there is no consensus regarding two important factors: 1) what are the sensible formulations of weak training signals when collecting a particular dataset? such that a benchmark is possible; 2) given some well-defined training signals, how good or bad do the mature fully-supervised methods perform as baselines? Addressing these two questions could allow us to clearly understand whether dense annotations are indeed necessary, and if not, the benchmark and baselines could serve as the foundation for the development of advanced algorithms. To this end, we first set up a basic weak supervision benchmark and then evaluate several existing methods in this new context.

Preparing Datasets with Weak Training Signals. The fundamental objective of weakly-supervised segmentation is to obtain accurate estimations with as little as annotation cost. However, it is non-trivial to compare the cost of different annotation methods in practice. Existing annotation options include 1) randomly annotating sparse point labels [75], 2) actively annotating sparse point labels [20, 49], 2) annotating per-object labels [54] and 3) annotating sub-cloud labels [69]. Admittedly, all methods have merits. For the purpose of benchmarking, we opt for the random point annotation strategy, considering the practical simplicity of building such an annotation tool[1]. With this, we choose the well-known S3DIS dataset [2] as the testbed. The Areas

[1]A simple tool that randomly pops up a center point together with a local area as the reference for manually annotating.

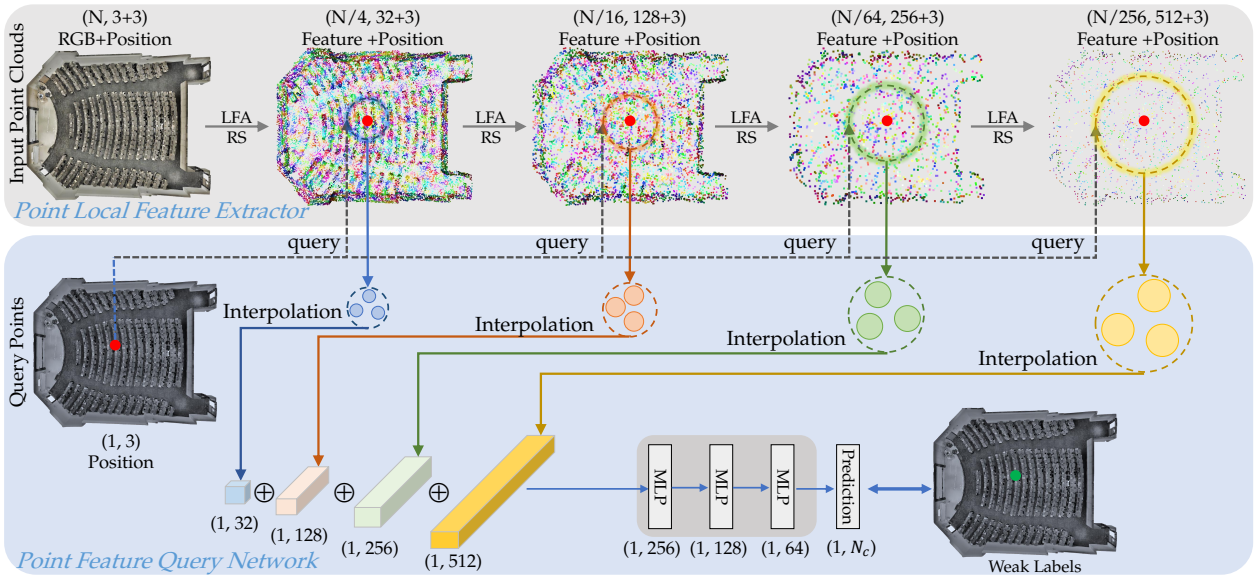

Figure 3: The pipeline of our SQN at the training stage with weak supervision. We only show one query point for simplicity.

$\{1/2/3/4/6\}$ are selected as the training point clouds, the Area 5 is fully annotated for testing only. We set up the following four groups of weak signals for training.

- Group 1: For each room in all training areas, the randomly selected 10% 3D points are annotated;
- Group 2/3/4: Similar to Group 1, but only 1%, 1‰, 1‱ points are annotated;

Using Fully-supervised Methods as Baselines. We select the seminal works PointNet/PointNet++ [41, 42] and the recent large-scale-point-cloud friendly RandLA-Net [22] as baselines. These methods are end-to-end trained on the four groups of weakly annotated data without using any additional modules. During training, only the labeled points are used to compute the loss for back-propagation. In total, 12 models (3 models/group $\times$ 4 groups) are trained for evaluation on the full Area 5.

Results and Findings. Figure 2 shows the mIoU scores of all models for segmenting the total 13 classes. The results under full supervision (100% annotations for all training data) are included for comparison. It can be seen that:

- The performance of all baselines only decreases marginally (less than 3%) even though the proportion of point annotations drops significantly from 100% to 1%. This clearly shows that the dense annotations are actually unnecessary to obtain a comparable and favorable segmentation accuracy.
- The performance of all baselines drops significantly once the annotated points are less than 1‰. This critical point indicates that keeping a certain amount of training signals is also essential for weak supervision.

Above all, we may conclude that for segmenting large-scale point clouds which are usually dominated by major classes and have numerous repeatable local patterns, it is more appealing to develop weakly-supervised methods which have an excellent trade-off between annotation costs and estimation accuracy. With this motivation, we propose SQN which achieves up to fully-supervised accuracy using only 1‰ labels for training.

4. SQN

4.1. Overview

Given extremely limited point annotations, the fundamental challenge for weakly-supervised learning is how to fully utilize the sparse yet valuable training signals to update the network parameters, such that more geometrically meaningful local patterns can be learned. To resolve this, we design a simple SQN which consists of two major components: 1) a point local feature extractor to learn diverse visual patterns; 2) a flexible point feature query network to collect as many as possible relevant semantic features for weakly-supervised training. As shown in Figure 3, our two sub-networks are illustrated by the stacked blocks.

4.2. Point Local Feature Extractor

This component aims to extract local features for all points. As discussed in Section 2.1, there are many excellent backbone networks in the literature that are able to extract per-point features. In general, these networks stack multiple encoding layers together with downsampling operations to extract hierarchical local features. In this paper,

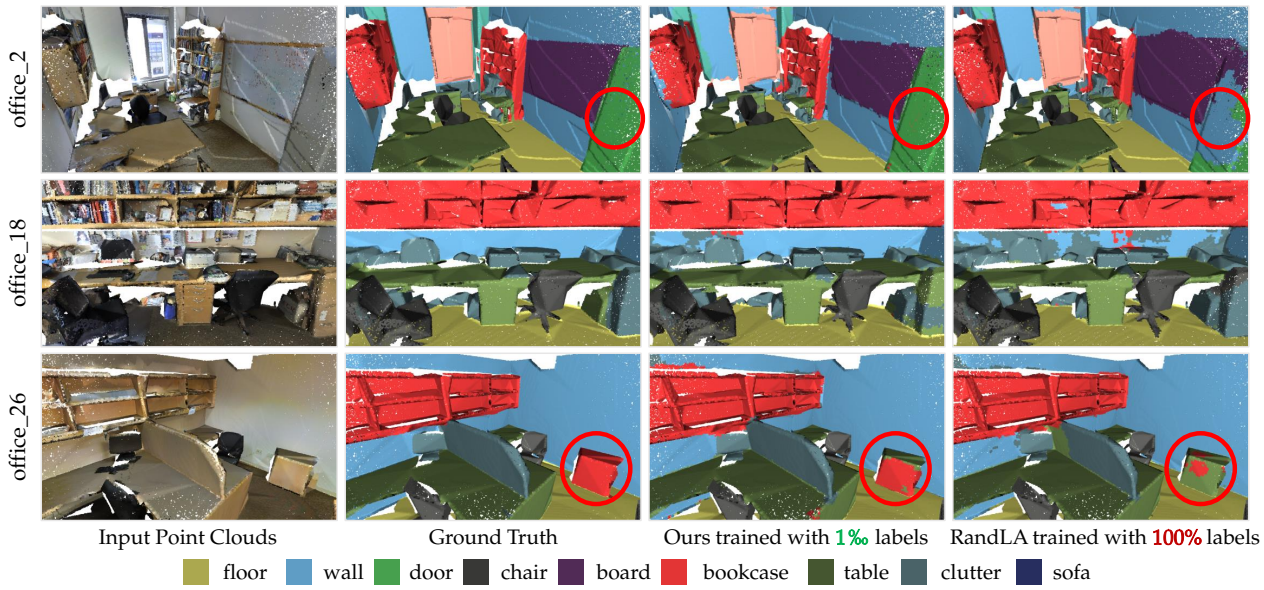

Figure 4: Qualitative results achieved by our SQN and RandLA-Net [22] on the *Area-5* of the S3DIS dataset.

we use the encoder of RandLA-Net [22] as our feature extractor thanks to its efficiency on large-scale point clouds, although our SQN is not restricted to any particular backbone network.

As shown in the top block of Figure 3, the encoder includes four layers of Local Feature Aggregation (LFA) followed by a Random Sampling (RS) operation. Details refer to RandLA-Net [22]. Given an input point cloud $\mathcal{P}$ with N points, four levels of hierarchical point features are extracted after each encoding layer, *i.e.*, 1) $\frac{N}{4} \times 32$, 2) $\frac{N}{16} \times 128$, 3) $\frac{N}{64} \times 256$, and 4) $\frac{N}{256} \times 512$. To facilitate the subsequent query network, the corresponding point location xyz are always preserved for each hierarchical feature vector.

4.3. Point Feature Query Network

Given the extracted point feature vectors, this query network is designed to collect as many relevant features, to be trained using the available sparse signals. In particular, as shown in the bottom block of Figure 3, it takes a specific 3D query point as input and then acquires a set of learned point features relevant to that point. Fundamentally, this is assumed that the query point shares similar semantic information with the collected point features, such that the training signals from the query points can be shared and back-propagated for the relevant points. The network consists of three modules: 1) Searching Spatial Neighbouring Point Features, 2) Interpolating Query Point Features, 3) Inferring Query Point Semantics.

Searching Spatial Neighbouring Point Features. Given a 3D query point p with its location xyz, this module is to simply search the nearest K points in each of the previous 4-level encoded feature vectors, according to the point-wise Euclidean distance. For example, as to the first level of extracted point features, the most relevant K points are selected, acquiring the raw feature vectors $\{\boldsymbol{F}_p^1, \dots \boldsymbol{F}_p^K\}$.

Interpolating Query Point Features. For each level of features, the queried K vectors are compressed into a compact representation for the query point p. For simplicity, we apply the trilinear interpolation method to compute a feature vector for p, according to the Euclidean distance between p and each of K points. Eventually, four hierarchical feature vectors are concatenated together, representing all relevant point features from the entire 3D point cloud.

Inferring Query Point Semantics. After obtaining the unique and representative feature vector for the query point p, we feed it into a series of MLPs, directly inferring the point semantic category.

Overall, given a sparse number of annotated points, we parallelly query their neighbouring point features for training. This allows the valuable training signals to be back-propagated to a much wider spatial context. During testing, all 3D points are fed into the two sub-networks for semantic estimation. In fact, our simple query mechanism greatly enables the network to infer the point semantic category from a significantly larger receptive field.

5. Experiments

5.1. Comparison with SOTA Approaches

Experimental Setup. We choose the hyperparameter K as 3 in all experiments, and our SQN is trained end-to-end with 1‰ randomly annotated points. Note that, we evaluate the

Settings	Methods	**mIoU(%)**	ceil.	floor	wall	beam	col.	win.	door	table	chair	sofa	book.	board	clutter
Full supervision (100%)	PointNet [41]	41.1	88.8	97.3	69.8	0.1	3.9	46.3	10.8	58.9	52.6	5.9	40.3	26.4	33.2
	PointCNN [31]	57.3	92.3	98.2	79.4	0.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	56.7
	SPGraph [28]	58.0	89.4	96.9	78.1	0.0	42.8	48.9	61.6	84.7	75.4	69.8	52.6	2.1	52.2
	SPH3D [30]	59.5	93.3	97.1	81.1	0.0	33.2	45.8	43.8	79.7	86.9	33.2	71.5	54.1	53.7
	PointWeb [83]	60.3	92.0	98.5	79.4	0.0	21.1	59.7	34.8	76.3	88.3	46.9	69.3	64.9	52.5
	RandLA-Net [22]	63.0	92.4	96.7	80.6	0.0	18.3	61.3	43.3	77.2	85.2	71.5	71.0	69.2	52.3
	KPConv rigid [60]	65.4	92.6	97.3	81.4	0.0	16.5	54.5	69.5	80.2	90.1	66.4	74.6	63.7	58.1
Weak supervision (10%)	Π Model [27]	46.3	91.8	97.1	73.8	0.0	5.1	42.0	19.6	67.2	66.7	47.9	19.1	30.6	41.3
	MT [55]	47.9	**92.2**	96.8	74.1	0.0	10.4	46.2	17.7	70.7	67.0	50.2	24.4	30.7	42.2
	Xu [75]	48.0	90.9	**97.3**	74.8	0.0	8.4	49.3	27.3	**71.7**	69.0	53.2	16.5	23.3	42.8
Weak supervision (1pt, 2‰†)	Π Model [27]	44.3	89.1	97.0	71.5	0.0	3.6	43.2	27.4	63.1	62.1	43.7	14.7	24.0	36.7
	MT [55]	44.4	88.9	96.8	70.1	**0.1**	3.0	44.3	28.8	63.7	63.6	43.7	15.5	23.0	35.8
	Xu [75]	44.5	90.1	97.1	71.9	0.0	1.9	47.2	29.3	64.0	62.9	42.2	15.9	18.9	37.5
	Ours (1‰)	**61.4**	91.7	95.6	**78.7**	0.0	**24.2**	**55.8**	**63.1**	70.5	**83.1**	**60.6**	**67.8**	**56.1**	**50.6**

Table 1: Quantitative results of different methods on the *Area-5* of S3DIS dataset. Mean IoU (mIoU, %), and per-class IoU (%) scores are reported. Note that, †there is only 1 point annotated for each category within each 1m×1m point cloud block, which is about 2‰ points annotated. Bold represents the best result in weakly setting and underlined represents the best in fully setting.

final performance on all points of the original test set for a fair comparison. Following [22], we use the Overall Accuracy (OA) and mean Intersection-over-Union (mIoU) as the main evaluation metrics. All experiments are conducted on a PC with an Intel Core™ i9-10900X CPU and an NVIDIA RTX Titan GPU with 24G memory.

Evaluation on S3DIS. The indoor S3DIS [2] dataset has 273 million 3D points belonging to 13 classes. Following [75], we report the results on Area-5. Note that, the method [75] is trained with randomly annotated 10% labels, while our SQN only uses 1‰ labels for training. Table 1 compares our SQN with three groups of approaches: 1) Fully-supervised methods including PointNet [41], SPGraph [28], and KPConv [60] with 100% training labels; 2) Weakly-supervised methods [75, 55, 27] with 10% training annotations; 3) Weakly-supervised methods [75, 55, 27] with 1 point annotation which is about 2‰ annotations.

It can be seen that our SQN outperforms all weakly-supervised methods [55, 75, 27] by large margins with only 1‰ annotations used. Surprisingly, our SQN is comparable to the fully-supervised RandLA-Net [22]. Figure 4 shows qualitative comparisons of RandLA-Net and our SQN.

Evaluation on ScanNet. The ScanNet [12] dataset consists of 1613 indoor scans (1201 for training, 312 for validation, and 100 for online testing). It has nearly 242 million points sampled from the densely reconstructed 3D meshes.

Table 2 shows the quantitative results of different approaches on the hidden test set. It can be seen that our SQN achieves much higher mIoU scores with only 1‰ training labels, compared with MPRM [69] which is trained with sub-cloud labels.

Settings	Methods	**mIoU(%)**
Full supervision	PointNet++ [42]	33.9
	SPLATNet [50]	39.3
	TangentConv [56]	43.8
	PointCNN [31]	45.8
	PointConv [72]	55.6
	SPH3D-GCN [30]	61.0
	KPConv [60]	68.4
	SparseConvNet [13]	72.5
	MinkowskiNet [9]	73.6
	Virtual MVFusion [26]	74.9
	OccuSeg [17]	76.4
	RandLA-Net [22]	64.5
Weak supervision	MPRM* [69]	41.1
	Ours (1‰)	**53.5**

Table 2: Quantitative results of different approaches on ScanNet (online test set). *MPRM [69] takes sub-cloud labels as supervision signal.

5.2. Evaluation on Large-Scale 3D Benchmarks

We further evaluate our SQN on five outdoor large-scale point cloud datasets, including Semantic3D [15], SensatUrban [21], Toronto3D [52], DALES [62], and SemanticKITTI [3] dataset. Note that, we only compare our approach with existing supervised methods in this section, since there is no other weakly-supervised method reported on these datasets.

Evaluation on Semantic3D. This dataset consists of 30 urban and rural street-scenarios (15 for training and 15 for online testing). There are 4 billion points in total acquired by the terrestrial laser. In particular, we also train our SQN with only 1‱ randomly annotated points, considering the extremely large amount of 3D points scanned.

Table 3 compares our results with a number of fully-supervised methods. It can be seen that our SQN trained

with 1‰ labels achieves competitive performance with fully-supervised baselines on both *Semantic8* and *Reduced8* subsets. This clearly demonstrates the effectiveness of our semantic query framework, which takes full advantage of the limited annotations. Our SQN trained with 1‱ labels also achieves satisfactory accuracy, though there is space to be improved in the future.

		Semantic8		*Reduced8*	
	Methods	OA(%)	**mIoU(%)**	OA(%)	**mIoU(%)**
Full sup.	SnapNet [6]	91.0	67.4	88.6	59.1
	PointNet++ [42]	85.7	63.1	-	-
	ShellNet [82]	-	-	93.2	69.3
	GACNet [66]	-	-	91.9	70.8
	RGNet [61]	90.6	72.0	94.5	74.7
	SPG [28]	92.9	76.2	94.0	73.2
	KPConv [60]	-	-	92.9	74.6
	ConvPoint [5]	93.4	76.5	-	-
	WreathProdNet [67]	94.6	77.1	-	-
	RandLA-Net [22]	95.0	75.8	94.8	77.4
Weak sup.	**Ours (1‰)**	**94.8**	**72.3**	**93.7**	**74.7**
	Ours (1‱)	91.9	58.8	90.3	65.6

Table 3: Quantitative results of different approaches on Semantic3D [15]. The scores are obtained from the recent publications. Accessed on 17 March 2021.

Settings	Methods	OA(%)	mAcc(%)	**mIoU(%)**
Full supervision	PointNet [41]	80.78	30.32	23.71
	PointNet++ [42]	84.30	39.97	32.92
	TagentConv [56]	76.97	43.71	33.30
	SPGraph [28]	25.27	44.39	37.29
	SparseConv [13]	88.66	63.28	42.66
	KPConv [60]	93.20	63.76	57.58
	RandLA-Net [22]	89.78	69.64	52.69
Weak supervision	**Ours (1‰)**	**90.97**	**70.84**	**53.97**
	Ours (1‱)	85.57	49.40	37.17

Table 4: Quantitative results of different approaches on the urban-scale SensatUrban [21] dataset. Overall Accuracy (OA, %), mean class Accuracy (mAcc, %), and mean IoU (mIoU, %) are reported.

Evaluation on SensatUrban. This is a new urban-scale photogrammetry point cloud dataset covering over 7.6 square kilometers of urban areas in the UK. It has nearly 3 billion points in total. Note that, this dataset is extremely challenging due to the unbalanced class distributions.

As shown in Table 4, the performance of our SQN is on par or even better than the fully-supervised methods KPConv and RandLA-Net, whilst the model is only supplied with 1‰ labels for training. This shows the great potential of our method, especially for extremely large-scale point clouds with billions of points, where the manual annotation is unrealistic and impractical. The detailed per-class IoU scores can be found in the supplementary material.

Evaluation on Toronto3D. This dataset consists of 1KM urban road point clouds acquired by vehicle-mounted mobile laser systems. It has 78.3 million points belonging to 8 semantic categories.

Settings	Methods	OA(%)	**mIoU(%)**
Full supervision	PointNet++ [42]	84.88	41.81
	PointNet++ (MSG) [42]	92.56	59.47
	DGCNN [68]	94.24	61.79
	KPFCNN [60]	95.39	69.11
	MS-PCNN [36]	90.03	65.89
	TGNet [32]	94.08	61.34
	MS-TGNet [52]	95.71	70.50
	RandLA-Net (w/ RGB) [22]	97.15	81.88
	RandLA-Net (w/o RGB) [22]	95.63	77.72
Weak supervision	**Ours (w/ RGB, 1‰)**	**96.67**	**77.75**
	Ours (w/o RGB, 1‰)	92.84	69.35
	Ours (w/ RGB, 1‱)	94.19	68.17
	Ours (w/o RGB, 1‱)	90.47	57.57

Table 5: Quantitative results of different approaches on the Toronto3D [52] dataset.

We provide the quantitative comparison of our SQN and several fully-supervised methods in Table 5. Following [22], we also additionally report the performance of our method with and without color information. It can be seen that our SQN outperforms several fully-supervised methods such as the strong KPConv, with merely 1‰ of point annotations for training. We notice that the usage of color information closes the gap between our method and the top-performing RandLA-Net [22]. This implies that it could be helpful to introduce auxiliary information under the setting of weak supervision.

Evaluation on DALES. This dataset consists of large-scale earth scans acquired by an aerial LiDAR. It covers over 10 km^2 spatial ranges with 5 million points belonging to 8 semantic categories. We compare our SQN with strong fully-supervised approaches.

As shown in Table 6, our method achieves higher mIoU scores than PointNet++ [42], ConvPoint [5], SPGraph [28], PointCNN [31] and ShellNet [82], with only 1‰ labels for training. However, there is still a performance gap compared with the the leading fully-supervised counterparts, primarily due to our weak performance on minor categories such as *trucks* and *cars*. It is very likely that the simple random annotation strategy may happen to ignore the minor classes. More details are provided in the Appendix.

Evaluation on SemanticKITTI. This large-scale dataset consists of point cloud sequences captured by LiDAR for autonomous driving. In particular, it has 22 sequences, 43552 sparse scans, and nearly 4 billion points. Note that, RGB is not available in this dataset.

We compare our SQN and fully-supervised techniques on the online test set in Table 7. It can be seen that our approach achieves a satisfactory mIoU score, outperforming several strong baselines with only 1‰ labels for training. In addition, our model only has 1.05 million trainable parameters, and is extremely lightweight and suitable for real-

Settings	Methods	OA(%)	**mIoU(%)**
Full supervision	ShellNet [82]	96.4	57.4
	PointCNN [31]	97.2	58.4
	SPGraph [28]	95.5	60.6
	ConvPoint [4]	97.2	67.4
	PointNet++ [42]	95.7	68.3
	KPConv [60]	97.8	81.1
	Pyramid Point [63]	98.3	83.6
	RandLA-Net [22]	97.1	80.0
Weak supervision	**Ours (1‰)**	**97.0**	**72.0**
	Ours (1‰‰)	95.9	60.4

Table 6: Quantitative results of different approaches on the DALES [62] dataset.

Settings	Methods	Params(M)	**mIoU(%)**
Full supervision	PointNet [41]	3	14.6
	PointNet++ [42]	6	20.1
	SPGraph [28]	0.25	17.4
	TagentConv [56]	0.4	40.9
	KPConv [60]	14.3	58.8
	FusionNet [79]	-	61.3
	SPVNAS [86]	12.5	66.4
	Cylinder3D [86]	55.85	67.8
	$(AF)^2$-S3Net [8]	-	69.7
	RandLA-Net [22]	1.24	53.9
Weak supervision	**Ours (1‰)**	**1.05**	**50.8**
	Ours (1‰‰)	1.05	39.1

Table 7: Quantitative results of different approaches on the SemanticKITTI dataset. M: million.

world applications.

5.3. Ablation Study

To evaluate the effectiveness of each module in our framework, we conduct the following ablation studies. All ablated networks are trained on Areas$\{1/2/3/4/6\}$ with 1‰ labels, and tested on the *Area-5* of the S3DIS dataset.

(1) Variants of Semantic Queries. The hierarchical point feature query mechanism for semantic estimation is the major component of our SQN. To evaluate this component, we perform semantic query at different encoding layers. In particular, we train four additional models, each of which has a different combination of queried neighbouring point features.

From Table 8 we can see that the segmentation performance drops significantly if we only collect the relevant point features at a single layer (*e.g.*, the first or the last layer), whilst querying at the last layer can achieve much better results than in the first layer. This is because the points in the last encoding layer are quite sparse but representative, aggregating a large number of neighboring points. Additionally, querying at different encoding layers and combining them is likely to achieve better segmentation results, mainly because it integrates different spatial levels of semantic content and considers more neighboring points.

(2) Varying Number of Queried Neighbours. Intuitively, querying a larger neighborhood is more likely to achieve better results. However, an overly large neighborhood may include points with very different semantics, deteriorating overall performance. To investigate the impact of the number of neighboring points used in our semantic query, we conduct experiments by varying the number of neighboring points from 1 to 25. As shown in Figure 5, our choice of K = 3 achieves the highest score. However, the overall performance with different numbers of neighboring points does not change significantly, showing that our simple query mechanism is robust to the size of the neighboring patch. Instead, the mixture of different levels of feature plays a more important role as demonstrated in Table 8.

Model	1st	2nd	3rd	4st	OA(%)	mIoU(%)
A	✓				48.66	22.89
B				✓	75.54	46.02
C	✓	✓			70.76	38.18
D	✓	✓	✓		82.37	54.21
E	✓	✓	✓	✓	86.26	61.37

Table 8: Ablations of different levels of semantic query.

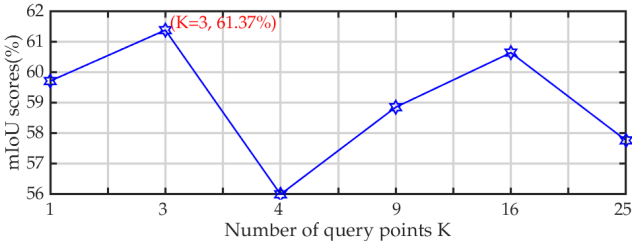

Figure 5: The results of our SQN with different number of querying points on the *Area-5* of the S3DIS dataset.

6. Discussions

Whilst the proposed SQN achieves remarkable results in semantic segmentation with extremely weak supervision, it also has limitations. First, the assumption about the consistency of neighborhood semantics does not hold at the boundary areas with different semantics. Future works will further explore how to achieve more precise segmentation performance at boundaries. Second, our feature extractor is still limited to capture precise point features, more advanced encoding architecture will be used in future work. We believe this is the major reason why our SQN still lags behind some state-of-the-art fully-supervised methods.

7. Conclusion

In this paper, we propose SQN, a conceptually simple, elegant, and novel framework to achieve semantic learning of large-scale point clouds, with only 1‰ labels for training.

We first point out the redundancy of dense 3D annotations through extensive experiments, and then propose an effective semantic query framework based on the assumption of semantic similarity of neighboring points in 3D space. The proposed SQN shows great potential for weakly-supervised semantic segmentation of large-scale point clouds. It would be interesting to extend this method for weakly-supervised instance segmentation, panoptic segmentation, and interactive annotation based on active learning.

References

[1] Eren Erdal Aksoy, Saimir Baci, and Selcuk Cavdar. Salsanet: Fast road and vehicle segmentation in LiDAR point clouds for autonomous driving. In IV, pages 926–932, 2019. 2

[2] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2D-3D-semantic data for indoor scene understanding. In ICCV, 2017. 2, 3, 6, 13, 14

[3] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences. In ICCV, pages 9297–9307, 2019. 1, 2, 6, 13, 17, 19

[4] Alexandre Boulch. Generalizing discrete convolutions for unstructured point clouds. In 3DOR, pages 71–78, 2019. 8, 15, 17

[5] Alexandre Boulch, Gilles Puy, and Renaud Marlet. Fkaconv: Feature-kernel alignment for point cloud convolution. In ACCV, 2020. 7, 15

[6] A Boulch, B Le Saux, and N Audebert. Unstructured point cloud semantic labeling using deep segmentation networks. In 3DOR, pages 17–24, 2017. 2, 7, 15

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In ICML, pages 1597–1607, 2020. 1, 3

[8] Ran Cheng, Ryan Razani, Ehsan Taghavi, Enxu Li, and Bingbing Liu. 2-S3Net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. arXiv preprint arXiv:2102.04530, 2021. 1, 2, 8

[9] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4D spatio-temporal convnets: Minkowski convolutional neural networks. In CVPR, pages 3075–3084, 2019. 1, 2, 6, 15

[10] Jhonatan Contreras and Joachim Denzler. Edge-convolution point net for semantic segmentation of large-scale point clouds. In IGARSS, pages 5236–5239, 2019. 15

[11] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. SalsaNext: Fast semantic segmentation of LiDAR point clouds for autonomous driving. In ISVC, 2020. 2, 17

[12] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In CVPR, pages 5828–5839, 2017. 1, 6, 15

[13] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3D semantic segmentation with submanifold sparse convolutional networks. In CVPR, 2018. 1, 2, 6, 7, 15, 16

[14] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3D point clouds: A survey. IEEE TPAMI, 2020. 2

[15] Timo Hackel, Nikolay Savinov, Lubor Ladicky, Jan D Wegner, Konrad Schindler, and Marc Pollefeys. Semantic3D.Net: A new large-scale point cloud classification benchmark. ISPRS, 2017. 2, 6, 7, 13, 15

[16] Timo Hackel, Jan D Wegner, and Konrad Schindler. Fast semantic segmentation of 3D point clouds with strongly varying density. ISPRS, 3:177–184, 2016. 15

[17] Lei Han, Tian Zheng, Lan Xu, and Lu Fang. Occuseg: Occupancy-aware 3d instance segmentation. In CVPR, pages 2940–2949, 2020. 6, 15

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In CVPR, pages 9729–9738, 2020. 3

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2016. 2

[20] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3D scene understanding with contrastive scene contexts. In CVPR, 2021. 2, 3

[21] Qingyong Hu, Bo Yang, Sheikh Khalid, Wen Xiao, Niki Trigoni, and Andrew Markham. Towards semantic segmentation of urban-scale 3D point clouds: A dataset, benchmarks and challenges. In CVPR, 2021. 2, 6, 7, 13, 18

[22] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. RandLA-Net: Efficient semantic segmentation of large-scale point clouds. In CVPR, 2020. 1, 2, 4, 5, 6, 7, 8, 14, 15, 16, 17

[23] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3D segmentation of point clouds. In ICCV, 2018. 14

[24] Maximilian Jaritz, Jiayuan Gu, and Hao Su. Multi-view pointnet for 3D scene understanding. In ICCVW, pages 0–0, 2019. 3

[25] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected CRFs with gaussian edge potentials. In NeurIPS, pages 109–117, 2011. 2

[26] Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David Ross, Brian Brewington, Thomas Funkhouser, and Caroline Pantofaru. Virtual multi-view fusion for 3D semantic segmentation. In ECCV, pages 518–535. Springer, 2020. 2, 6, 15

[27] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In ICLR, 2017. 6

[28] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In CVPR, pages 4558–4567, 2018. 6, 7, 8, 14, 15, 16, 17

[29] Huan Lei, Naveed Akhtar, and Ajmal Mian. SegGCN: Efficient 3D point cloud segmentation with fuzzy spherical kernel. In CVPR, 2020. 15

[30] Huan Lei, Naveed Akhtar, and Ajmal Mian. Spherical kernel for efficient graph convolution on 3D point clouds. IEEE TPAMI, 2020. 2, 6, 15

[31] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. PointCNN: Convolution on X-transformed points. In NeurIPS, 2018. 1, 2, 6, 7, 8, 14, 15, 17

[32] Ying Li, Lingfei Ma, Zilong Zhong, Dongpu Cao, and Jonathan Li. TGNet: Geometric graph cnn on 3D point cloud segmentation. IEEE TGRS, 2019. 7, 16

[33] Yunze Liu, Li Yi, Shanghang Zhang, Qingnan Fan, Thomas Funkhouser, and Hao Dong. P4contrast: Contrastive learning with pairs of point-pixel pairs for rgb-d scene understanding. arXiv preprint arXiv:2012.13089, 2020. 1, 3

[34] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3D deep learning. In NeurIPS, 2019. 1, 3

[35] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In CVPR, pages 3431–3440, 2015. 2

[36] Lingfei Ma, Ying Li, Jonathan Li, Weikai Tan, Yongtao Yu, and Michael A Chapman. Multi-scale point-wise convolutional neural networks for 3D object segmentation from LiDAR point clouds in large-scale environments. IEEE TITS, 2019. 7, 16

[37] Yanni Ma, Yulan Guo, Hao Liu, Yinjie Lei, and Gongjian Wen. Global context reasoning for semantic segmentation of 3D point clouds. WACV, 2020. 15

[38] Hsien-Yu Meng, Lin Gao, Yu-Kun Lai, and Dinesh Manocha. VV-Net: Voxel vae net with group convolutions for point cloud segmentation. In ICCV, 2019. 2

[39] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate LiDAR semantic segmentation. In IROS, pages 4213–4220, 2019. 2, 17

[40] Javier A Montoya-Zegarra, Jan D Wegner, L'ubor Ladickỳ, and Konrad Schindler. Mind the gap: modeling local and global context in (road) networks. In GCPR, 2014. 15

[41] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In CVPR, pages 652–660, 2017. 1, 2, 4, 6, 7, 8, 14, 16, 17

[42] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In NeurIPS, 2017. 1, 2, 4, 6, 7, 8, 14, 15, 16, 17

[43] Dario Rethage, Johanna Wald, Jurgen Sturm, Nassir Navab, and Federico Tombari. Fully-convolutional point networks for large-scale point clouds. In ECCV, 2018. 3

[44] Radu Alexandru Rosu, Peer Schütt, Jan Quenzel, and Sven Behnke. LatticeNet: Fast point cloud segmentation using permutohedral lattices. In RSS, 2020. 17

[45] Xavier Roynard, Jean-Emmanuel Deschaud, and François Goulette. Classification of point cloud for road scene understanding with multiscale voxel deep network. In PPNIV, 2018. 15

[46] Xavier Roynard, Jean-Emmanuel Deschaud, and François Goulette. Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. IJRR, 37(6):545–557, 2018. 2

[47] Jonathan Sauder and Bjarne Sievers. Self-supervised deep learning on point clouds by reconstructing space. In NeurIPS, pages 12962–12972, 2019. 3

[48] Charu Sharma and Manohar Kaul. Self-supervised few-shot learning on point clouds. In NeurIPS, 2020. 3

[49] Xian Shi, Xun Xu, Ke Chen, Lile Cai, Chuan Sheng Foo, and Kui Jia. Label-efficient point cloud semantic segmentation: An active learning approach. arXiv preprint arXiv:2101.06931, 2021. 3

[50] Hang Su, Varun Jampani, Deqing Sun, Subhransu Maji, Evangelos Kalogerakis, Ming-Hsuan Yang, and Jan Kautz. SPLATNet: sparse lattice networks for point cloud processing. In CVPR, pages 2530–2539, 2018. 6, 15, 17

[51] Weiwei Sun, Andrea Tagliasacchi, Boyang Deng, Sara Sabour, Soroosh Yazdani, Geoffrey Hinton, and Kwang Moo Yi. Canonical capsules: Unsupervised capsules in canonical pose. arXiv preprint arXiv:2012.04718, 2020. 3

[52] Weikai Tan, Nannan Qin, Lingfei Ma, Ying Li, Jing Du, Guorong Cai, Ke Yang, and Jonathan Li. Toronto-3D: A large-scale mobile LiDAR dataset for semantic segmentation of urban roadways. In CVPRW, pages 202–203, 2020. 2, 6, 7, 16

[53] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3D architectures with sparse point-voxel convolution. In ECCV, pages 685–702, 2020. 3

[54] An Tao, Yueqi Duan, Yi Wei, Jiwen Lu, and Jie Zhou. Seggroup: Seg-level supervision for 3D instance and semantic segmentation. arXiv preprint arXiv:2012.10217, 2020. 2, 3

[55] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In NeurIPS, pages 1195–1204, 2017. 6

[56] Maxim Tatarchenko, Jaesik Park, Vladlen Koltun, and Qian-Yi Zhou. Tangent convolutions for dense prediction in 3D. In CVPR, pages 3887–3896, 2018. 6, 7, 8, 15, 16, 17

[57] Lyne Tchapmi, Christopher Choy, Iro Armeni, JunYoung Gwak, and Silvio Savarese. Segcloud: Semantic segmentation of 3D point clouds. In 3DV, pages 537–547, 2017. 2, 15

[58] Ali Thabet, Humam Alwassel, and Bernard Ghanem. Self-supervised learning of local features in 3D point clouds. In CVPRW, pages 938–939, 2020. 1, 3

[59] Hugues Thomas, François Goulette, Jean-Emmanuel Deschaud, Beatriz Marcotegui, and Yann LeGall. Semantic classification of 3D point clouds with multiscale spherical neighborhoods. In 3DV, pages 390–398, 2018. 15

[60] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. KPConv: Flexible and deformable convolution for point clouds. In ICCV, pages 6411–6420, 2019. 1, 2, 6, 7, 8, 14, 15, 16, 17

[61] Giang Truong, Syed Zulqarnain Gilani, Syed Mohammed Shamsul Islam, and David Suter. Fast point cloud registration using semantic segmentation. In DICTA, pages 1–8, 2019. 7, 15

[62] Nina Varney, Vijayan K Asari, and Quinn Graehling. DALES: A large-scale aerial LiDAR data set for semantic segmentation. In CVPRW, pages 186–187, 2020. 2, 6, 8, 13

[63] Nina Varney, Vijayan K Asari, and Quinn Graehling. Pyramid point: A multi-level focusing network for revisiting feature layers. arXiv preprint arXiv:2011.08692, 2020. 8, 17

[64] Hanchen Wang, Qi Liu, Xiangyu Yue, Joan Lasenby, and Matthew J Kusner. Pre-training by completing point clouds. arXiv preprint arXiv:2010.01089, 2020. 1, 3

[65] Haiyan Wang, Xuejian Rong, Liang Yang, Shuihua Wang, and Yingli Tian. Towards weakly supervised semantic segmentation in 3D graph-structured point clouds of wild scenes. In BMVC, page 284, 2019. 2

[66] Lei Wang, Yuchun Huang, Yaolin Hou, Shenman Zhang, and Jie Shan. Graph attention convolution for point cloud semantic segmentation. In CVPR, 2019. 7, 15

[67] Renhao Wang, Marjan Albooyeh, and Siamak Ravanbakhsh. Equivariant maps for hierarchical structures. arXiv preprint arXiv:2006.03627, 2020. 7, 15

[68] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. ACM TOG, 38(5):1–12, 2019. 7, 16

[69] Jiacheng Wei, Guosheng Lin, Kim-Hui Yap, Tzu-Yi Hung, and Lihua Xie. Multi-path region mining for weakly supervised 3D semantic segmentation on point clouds. In CVPR, pages 4384–4393, 2020. 2, 3, 6

[70] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud. In ICRA, pages 1887–1893, 2018. 2, 17

[71] Bichen Wu, Xuanyu Zhou, Sicheng Zhao, Xiangyu Yue, and Kurt Keutzer. SqueezeSegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud. In ICRA, pages 4376–4382, 2019. 2, 17

[72] Wenxuan Wu, Zhongang Qi, and Li Fuxin. PointConv: Deep convolutional networks on 3D point clouds. In CVPR, pages 9621–9630, 2018. 2, 6, 15

[73] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. PointContrast: Unsupervised pretraining for 3D point cloud understanding. In ECCV, pages 574–591, 2020. 1, 2, 3

[74] Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. SqueezeSegV3: Spatially-adaptive convolution for efficient point-cloud segmentation. In ECCV, pages 1–19, 2020. 2, 17

[75] Xun Xu and Gim Hee Lee. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In CVPR, pages 13706–13715, 2020. 2, 3, 6, 13

[76] Xu Yan, Jiantao Gao, Jie Li, Ruimao Zhang, Zhen Li, Rui Huang, and Shuguang Cui. Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. In AAAI, 2020. 2

[77] Xu Yan, Chaoda Zheng, Zhen Li, Sheng Wang, and Shuguang Cui. PointASNL: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In ICCV, pages 5589–5598, 2020. 14

[78] Xiaoqing Ye, Jiamao Li, Hexiao Huang, Liang Du, and Xiaolin Zhang. 3D recurrent neural networks with context fusion for point cloud semantic segmentation. In ECCV, 2018. 14

[79] Feihu Zhang, Jin Fang, Benjamin Wah, and Philip Torr. Deep fusionnet for point cloud semantic segmentation. In ECCV, volume 2, page 6, 2020. 3, 8

[80] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. PolarNet: An improved grid representation for online LiDAR point clouds semantic segmentation. In CVPR, pages 9601–9610, 2020. 17

[81] Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-supervised pretraining of 3D features on any point-cloud. arXiv preprint arXiv:2101.02691, 2021. 1, 3

[82] Zhiyuan Zhang, Binh-Son Hua, and Sai-Kit Yeung. ShellNet: Efficient point cloud convolutional neural networks using concentric shells statistics. In ICCV, pages 1607–1616, 2019. 7, 8, 14, 15, 17

[83] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. PointWeb: Enhancing local neighborhood features for point cloud processing. In CVPR, 2019. 6, 14

[84] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point Transformer. arXiv preprint arXiv:2012.09164, 2020. 2

[85] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In CVPR, pages 2921–2929, 2016. 3

[86] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation. In CVPR, 2021. 1, 2, 8

Appendix

A. Implementation Tricks

(1) Data augmentation. We follow [75] to apply different data augmentation techniques on the input point clouds during training, including random flipping, random rotation, and random noise.

(2) Re-training with generated pseudo labels. We observe that different datasets (*e.g.*, S3DIS [2] *vs*. Semantic3D [15]) have significantly different number of total points (273 million *vs*. 4000 million points). Therefore, the actual number of annotated points under our weak supervision setting (1‰) are also different (272 thousand *vs*. 4 million points). In the experiment, for the relatively small-scale S3DIS dataset which has extremely sparse supervision signals, we empirically find that retraining a new model with the generated pseudo labels can further increase the final segmentation performance. In particular, we firstly train our SQN with the limited annotated 1‰ points, and then we infer the semantics of the entire training set. These estimated semantics are regarded as pseudo labels. After that, we retrain a new model of our SQN from scratch with the generated pseudo labels. This retraining trick is able to fully utilize the extremely limited but valuable supervision signals. However, for the large-scale datasets including Semantic3D [15], SensatUrban [21], SemanticKITTI [3], DALES [62] in Section 5.2, our SQN can achieve satisfactory performance trained with 1‰ annotated points, while the retraining trick does not noticeably improve the performance.

B. Additional Experimental Results

(1) Detailed results of weak supervision benchmark. As mentioned in Section 3, we set up a basic weak supervision benchmark and evaluate several baseline methods under different forms of weak supervision. Here, we further provide the detailed benchmarking results on Table 9, with per-class IoU scores reported.

(2) How SQN perform under different numbers of labeled points? Here, we further examine the performance of SQN with different amounts of annotated points. As shown in Table 10, the proposed SQN can achieve satisfactory segmentation performance when there are only 1‰ labels available, but the performance drops significantly when there are only 1‱ labeled points available. This is primarily because the supervision signal is too sparse and limited in this case. We also noticed that our framework achieves the best performance when there is only 10% labels are used, indicating the supervision signal is sufficient in this case.

(3) How sensitive of SQN to different annotated points? As mentioned in Section 3, we randomly annotate 1‰ of points to provide supervision signals for training. To further verify the sensitivity of our SQN to different annotated points, we train our models five times with the exact same architectures, *i.e.*, the only change is that different randomly selected 1‰ of points are labeled. The experimental results are reported in Table 11.

It can be seen that there are indeed some differences between the results of different runs, but not significantly, indicating that the proposed framework is robust to randomly annotated points. We also noticed the major performance change lies in categories such as *door*, *sofa*, and *board*, showing that the minor categories in the dataset are more sensitive to the weakly-supervised settings such as 1‰ random labels.

(4) Additional results on S3DIS. In Section 5.1, we provide the quantitative results achieved on the *Area-5* subset of the S3DIS dataset. Here, we further report the detailed 6-fold cross-validation results achieved by our SQN and other baselines on this dataset in Table 12.

(5) Additional results on ScanNet. We also provided the detailed per class IoU results on Table 13.

(6) Additional results on Semantic3D. The detailed experimental results achieved on the *Semantic8* and *Reduced8* subset of the Semantic3D dataset are reported in Table 15 and Table 14. In addition, we also show the qualitative results achieved by our SQN on the *Reduced-8* subset with 1‰ labels in Fig 6.

(7) Additional results on SensatUrban. The detailed experimental results achieved on the SensatUrban dataset are reported in Table 16. In addition, we also show the qualitative results achieved by our SQN trained with only 1‰ labels on this dataset in Fig. 7. It is noted that the segmentation errors are mainly located in the boundary of areas with different semantic categories, primarily because the assumption of semantic similarity does not hold in the boundary areas. We leave this issue for future exploration.

(8) Additional results on Toronto3D. We also report the detailed experimental results on the Toronto3D dataset in Table 17.

(9) Additional results on DALES. We also report the detailed experimental results on the DALES dataset in Table 18.

(10) Additional results on SemanticKITTI. We also report the detailed experimental results on the SemanticKITTI dataset in Table 19. In addition, we also visualize the segmentation results achieved by our SQN on the validation set of the SemanticKITTI dataset in Fig. 8.

Settings	Methods	**mIoU(%)**	*ceil.*	*floor*	*wall*	*beam*	*col.*	*win.*	*door*	*chair*	*table*	*book.*	*sofa*	*board*	*clutter*
100% (Full supervision)	PointNet [41]	39.15	89.65	93.37	70.32	0.00	0.85	36.22	3.03	57.29	44.40	0.02	56.21	19.65	37.95
	PointNet++ [42]	52.36	88.84	90.88	75.83	0.18	10.47	43.57	13.86	71.90	82.81	35.71	67.28	51.60	47.80
	RandLA-Net [22]	63.75	92.19	97.67	81.12	0.00	20.22	61.02	41.49	78.53	88.04	70.65	74.21	70.65	53.01
10% (Random)	PointNet [41]	38.41	88.65	94.20	71.11	0.00	0.15	27.16	4.28	58.34	45.28	0.05	54.58	18.89	36.63
	PointNet++ [42]	52.34	86.67	90.68	76.37	0.00	10.63	43.76	20.14	70.37	83.34	40.97	68.00	41.88	47.64
	RandLA-Net [22]	61.87	91.87	97.58	79.71	0.00	19.24	60.76	39.36	77.06	86.44	61.77	70.63	67.50	52.34
1% (Random)	PointNet [41]	37.23	88.93	94.90	68.94	0.00	0.18	21.76	3.22	56.44	44.29	0.06	52.08	17.78	35.47
	PointNet++ [42]	48.61	87.65	89.39	73.98	0.01	7.05	39.15	12.98	66.28	73.94	28.97	66.87	40.13	45.56
	RandLA-Net [22]	59.13	90.86	96.96	78.34	0.00	16.40	60.33	25.73	75.30	83.05	59.10	69.00	64.84	48.73
1‰ (Random)	PointNet [41]	33.26	83.48	89.40	61.66	0.00	0.01	20.85	3.82	48.57	31.80	3.77	41.08	21.99	25.96
	PointNet++ [42]	42.57	85.43	88.76	69.87	0.00	1.00	24.61	7.30	57.72	66.28	24.90	58.80	30.89	37.82
	RandLA-Net [22]	52.90	89.90	95.90	75.28	0.00	7.46	52.38	26.48	62.19	74.48	49.10	60.15	49.26	45.08
1‰ (Random)	PointNet [41]	21.28	72.13	81.79	53.48	0.00	0.00	7.03	4.66	24.40	8.39	0.00	8.51	0.00	16.30
	PointNet++ [42]	33.53	77.84	83.87	67.09	0.23	3.89	34.83	16.60	41.49	30.65	0.79	39.23	13.81	25.50
	RandLA-Net [22]	33.16	85.15	89.20	61.54	0.00	3.66	13.17	9.11	29.15	42.29	6.52	46.78	16.86	27.72

Table 9: Detailed benchmark results of three baselines in the *Area-5* of the S3DIS [2] dataset. Different amount of points are randomly annotated for weak supervision.

Settings	Methods	**mIoU(%)**	*ceil.*	*floor*	*wall*	*beam*	*col.*	*win.*	*door*	*chair*	*table*	*book.*	*sofa*	*board*	*clutter*
100%	SQN	63.73	92.76	96.92	**81.84**	0.00	25.93	50.53	65.88	79.52	85.31	55.66	**72.51**	65.78	55.85
10%	SQN	**64.67**	**93.04**	**97.45**	81.55	0.00	**28.01**	55.77	68.68	**80.11**	**87.67**	55.25	72.31	63.91	**57.02**
1%	SQN	63.65	92.03	96.41	81.32	0.00	21.42	53.71	**73.17**	77.80	85.95	56.72	69.91	**66.57**	52.49
1‰	SQN	61.41	91.72	95.63	78.71	0.00	24.23	**55.89**	63.14	70.50	83.13	**60.67**	67.82	56.14	50.63
1‱	SQN	45.30	89.16	93.49	71.28	0.00	4.14	34.67	41.02	54.88	66.85	25.68	55.37	12.80	39.57

Table 10: Quantitative results achieved by our SQN on *Area-5* of S3DIS under different amount of labeled points.

	OA(%)	**mIoU(%)**	*ceil.*	*floor*	*wall*	*beam*	*col.*	*wind.*	*door*	*table*	*chair*	*sofa*	*book.*	*board*	*clut.*
Iter1	**86.53**	60.97	**92.33**	96.70	**78.99**	0.00	25.01	**56.76**	58.99	**74.22**	79.06	58.41	67.73	53.29	51.08
Iter2	85.63	59.24	91.72	**97.01**	77.35	0.00	20.10	53.55	**65.28**	71.63	**83.61**	51.44	65.57	43.37	49.49
Iter3	86.39	60.93	91.96	96.02	78.88	0.00	**25.31**	55.80	63.43	70.71	82.80	51.18	**68.39**	56.53	51.05
Iter4	86.32	59.40	92.22	96.07	78.85	0.00	19.00	50.10	65.19	68.37	83.27	49.79	67.09	51.33	50.89
Iter5	86.40	**61.56**	91.88	95.97	78.89	0.00	24.95	55.88	63.73	70.75	83.20	**59.29**	68.25	**56.37**	**51.13**
Average	86.25	60.42	92.02	96.35	78.59	0.00	22.87	54.42	63.32	71.14	82.39	54.02	67.41	52.18	50.73
STD	0.32	0.93	0.22	0.42	0.62	0.00	2.74	2.41	2.29	1.88	1.68	3.99	1.03	4.82	0.62

Table 11: Sensitivity analysis of the proposed SQN on the S3DIS dataset (*Area 5*) by running 5 times. Overall Accuracy (OA, %), mean IoU (mIoU, %), and per-class IoU (%) are reported. Bold represents the best result.

	Methods	OA(%)	mAcc(%)	**mIoU(%)**	*ceil.*	*floor*	*wall*	*beam*	*col.*	*wind.*	*door*	*table*	*chair*	*sofa*	*book.*	*board*	*clut.*
Full supervision	PointNet [41]	78.6	66.2	47.6	88.0	88.7	69.3	42.4	23.1	47.5	51.6	54.1	42.0	9.6	38.2	29.4	35.2
	RSNet [23]	-	66.5	56.5	92.5	92.8	78.6	32.8	34.4	51.6	68.1	59.7	60.1	16.4	50.2	44.9	52.0
	3P-RNN [78]	86.9	-	56.3	92.9	93.8	73.1	42.5	25.9	47.6	59.2	60.4	66.7	24.8	57.0	36.7	51.6
	SPG [28]	86.4	73.0	62.1	89.9	95.1	76.4	62.8	47.1	55.3	68.4	73.5	69.2	63.2	45.9	8.7	52.9
	PointCNN [31]	88.1	75.6	65.4	94.8	97.3	75.8	63.3	51.7	58.4	57.2	71.6	69.1	39.1	61.2	52.2	58.6
	PointWeb [83]	87.3	76.2	66.7	93.5	94.2	80.8	52.4	41.3	64.9	68.1	71.4	67.1	50.3	62.7	62.2	58.5
	ShellNet [82]	87.1	-	66.8	90.2	93.6	79.9	60.4	44.1	64.9	52.9	71.6	84.7	53.8	64.6	48.6	59.4
	PointASNL [77]	88.8	79.0	68.7	95.3	97.9	81.9	47.0	48.0	67.3	70.5	71.3	77.8	50.7	60.4	63.0	62.8
	KPConv (*rigid*) [60]	-	78.1	69.6	93.7	92.0	82.5	62.5	49.5	65.7	77.3	57.8	64.0	68.8	71.7	60.1	59.6
	KPConv (*deform*) [60]	-	79.1	70.6	93.6	92.4	83.1	63.9	54.3	66.1	76.6	57.8	64.0	69.3	74.9	61.3	60.3
	RandLA-Net [22]	88.0	82.0	70.0	93.1	96.1	80.6	62.4	48.0	64.4	69.4	69.4	76.4	60.0	64.2	65.9	60.1
Weak sup.	**Ours (1‰)**	**85.3**	**76.3**	**63.7**	**92.5**	**95.4**	**77.1**	**50.8**	**43.6**	**58.5**	**67.0**	**67.7**	**54.1**	**54.9**	**61.0**	**53.0**	**52.7**

Table 12: Quantitative results of different approaches on S3DIS [2] (6-fold cross-validation). Overall Accuracy (OA, %), mean class Accuracy (mAcc, %), mean IoU (mIoU, %), and per-class IoU (%) are reported.

Settings	Method	**mIoU(%)**	*bath*	*bed*	*bkshf*	*cab*	*chair*	*cntr*	*curt*	*desk*	*door*	*floor*	*other*	*pic*	*fridg*	*show*	*sink*	*sofa*	*table*	*toil*	*wall*	*wind*
Full supervision	ScanNet [12]	30.6	20.3	36.6	50.1	31.1	52.4	21.1	0.2	34.2	18.9	78.6	14.5	10.2	24.5	15.2	31.8	34.8	30.0	46.0	43.7	18.2
	PointNet++ [42]	33.9	58.4	47.8	45.8	25.6	36.0	25.0	24.7	27.8	26.1	67.7	18.3	11.7	21.2	14.5	36.4	34.6	23.2	54.8	52.3	25.2
	SPLATNET3D [50]	39.3	47.2	51.1	60.6	31.1	65.6	24.5	40.5	32.8	19.7	92.7	22.7	0.0	0.1	24.9	27.1	51.0	38.3	59.3	69.9	26.7
	Tangent-Conv [56]	43.8	43.7	64.6	47.4	36.9	64.5	35.3	25.8	28.2	27.9	91.8	29.8	14.7	28.3	29.4	48.7	56.2	42.7	61.9	63.3	35.2
	PointCNN [31]	45.8	57.7	61.1	35.6	32.1	71.5	29.9	37.6	32.8	31.9	94.4	28.5	16.4	21.6	22.9	48.4	54.5	45.6	75.5	70.9	47.5
	PointConv [72]	55.6	63.6	64.0	57.4	47.2	73.9	43.0	43.3	41.8	44.5	94.4	37.2	18.5	46.4	57.5	54.0	63.9	50.5	82.7	76.2	51.5
	SPH3D-GCN [30]	61.0	85.8	77.2	48.9	53.2	79.2	40.4	64.3	57.0	50.7	93.5	41.4	4.6	51.0	70.2	60.2	70.5	54.9	85.9	77.3	53.4
	KPConv [60]	68.4	84.7	75.8	78.4	64.7	81.4	47.3	77.2	60.5	59.4	93.5	45.0	18.1	58.7	80.5	69.0	78.5	61.4	88.2	81.9	63.2
	SparseConvNet [13]	72.5	64.7	82.1	84.6	72.1	86.9	53.3	75.4	60.3	61.4	95.5	57.2	32.5	71.0	87.0	72.4	82.3	62.8	93.4	86.5	68.3
	SegGCN [29]	58.9	83.3	73.1	53.9	51.4	78.9	44.8	46.7	57.3	48.4	93.6	39.6	6.1	50.1	50.7	59.4	70.0	56.3	87.4	77.1	49.3
	RandLA-Net [22]	64.5	77.8	73.1	69.9	57.7	82.9	44.6	73.6	47.7	52.3	94.5	45.4	26.9	48.4	74.9	61.8	73.8	59.9	82.7	79.2	62.1
	MinkowskiNet [9]	73.6	85.9	81.8	83.2	70.9	84.0	52.1	85.3	66.0	64.3	95.1	54.4	28.6	73.1	89.3	67.5	77.2	68.3	87.4	85.2	72.7
	Occuseg [17]	76.4	75.8	79.6	83.9	74.6	90.7	56.2	85.0	68.0	67.2	97.8	61.0	33.5	77.7	81.9	84.7	83.0	69.1	97.2	88.5	72.7
	Virtual MVFusion [26]	74.6	77.1	81.9	84.8	70.2	86.5	39.7	89.9	69.9	66.4	94.8	58.8	33.0	74.6	85.1	76.4	79.6	70.4	93.5	86.6	72.8
Weak supervision	**Ours (1‰)**†	**51.6**	**44.2**	**68.3**	**58.7**	**47.2**	**75.5**	**30.7**	**47.9**	**48.9**	**33.3**	**93.0**	**29.6**	**32.7**	**27.0**	**42.3**	**38.7**	**68.3**	**54.0**	**76.2**	**71.1**	**44.7**
	Ours (1‰)	35.9	35.5	59.0	53.6	21.4	62.8	25.8	40.4	34.0	19.9	91.8	24.2	14.5	01.5	16.6	09.4	53.4	36.7	33.3	58.1	25.8

Table 13: Quantitative results of different approaches on ScanNet (online test set). Mean IoU (mIoU, %), and per-class IoU (%) scores are reported. †To clarify, we report the results achieved on the validation set of the ScanNet in the main paper by mistake, this typo will be corrected in the next version of the paper for consistency.

Settings	Methods	**mIoU(%)**	OA(%)	*man-made.*	*natural.*	*high veg.*	*low veg.*	*buildings*	*hard scape*	*scanning art.*	*cars*
Full supervision	SnapNet_ [6]	59.1	88.6	82.0	77.3	79.7	22.9	91.1	18.4	37.3	64.4
	SEGCloud [57]	61.3	88.1	83.9	66.0	86.0	40.5	91.1	30.9	27.5	64.3
	RF_MSSF [59]	62.7	90.3	87.6	80.3	81.8	36.4	92.2	24.1	42.6	56.6
	MSDeepVoxNet [45]	65.3	88.4	83.0	67.2	83.8	36.7	92.4	31.3	50.0	78.2
	ShellNet [82]	69.3	93.2	96.3	90.4	83.9	41.0	94.2	34.7	43.9	70.2
	GACNet [66]	70.8	91.9	86.4	77.7	88.5	60.6	94.2	37.3	43.5	77.8
	SPG [28]	73.2	94.0	97.4	92.6	87.9	44.0	83.2	31.0	63.5	76.2
	KPConv [60]	74.6	92.9	90.9	82.2	84.2	47.9	94.9	40.0	77.3	79.9
	RGNet [61]	74.7	94.5	97.5	93.0	88.1	48.1	94.6	36.2	72.0	68.0
	RandLA-Net [22]	77.4	94.8	95.6	91.4	86.6	51.5	95.7	51.5	69.8	76.8
Weak supervision	**Ours (1‰)**	**74.7**	**93.7**	**97.1**	**90.8**	**84.7**	**48.5**	**93.9**	**37.4**	**71.0**	**74.5**
	Ours (1‰)	65.6	90.3	96.6	87.5	80.6	37.1	88.5	16.9	56.6	60.9

Table 14: Quantitative results of different approaches on Semantic3D (*reduced-8*) [15]. This test consists of 78,699,329 points. The scores are obtained from the recent publications. Bold represents the best result in weakly-supervised methods, and underlined represents the best results in fully-supervised methods. Accessed on 17 March 2021.

Settings	Methods	**mIoU(%)**	OA(%)	*man-made.*	*natural.*	*high veg.*	*low veg.*	*buildings*	*hard scape*	*scanning art.*	*cars*
Full supervision	TML-PC [40]	39.1	74.5	80.4	66.1	42.3	41.2	64.7	12.4	0.0	5.8
	TMLC-MS [16]	49.4	85.0	91.1	69.5	32.8	21.6	87.6	25.9	11.3	55.3
	PointNet++ [42]	63.1	85.7	81.9	78.1	64.3	51.7	75.9	36.4	43.7	72.6
	EdgeConv [10]	64.4	89.6	91.1	69.5	65.0	56.0	89.7	30.0	43.8	69.7
	SnapNet [6]	67.4	91.0	89.6	79.5	74.8	56.1	90.9	36.5	34.3	77.2
	PointGCR [37]	69.5	92.1	93.8	80.0	64.4	66.4	93.2	39.2	34.3	85.3
	RGNet [61]	72.0	90.6	86.4	70.3	69.5	68.0	96.9	43.4	52.3	89.5
	LCP [5]	74.6	94.1	94.7	85.2	77.4	70.4	94.0	52.9	29.4	92.6
	SPGraph [28]	76.2	92.9	91.5	75.6	78.3	71.7	94.4	56.8	52.9	88.4
	ConvPoint [4]	76.5	93.4	92.1	80.6	76.0	71.9	95.6	47.3	61.1	87.7
	RandLA-Net [22]	75.8	95.0	97.4	93.0	70.2	65.2	94.4	49.0	44.7	92.7
	WreathProdNet [67]	77.1	94.6	95.2	87.1	75.3	67.1	96.1	51.3	51.0	93.4
Weak supervision	**Ours (1‰)**	**72.3**	**94.8**	**97.9**	**93.2**	**65.5**	**63.4**	**94.9**	**44.9**	**47.4**	**70.9**
	Ours (1‰)	58.8	91.9	96.7	90.3	56.6	53.3	90.7	13.6	24.0	44.9

Table 15: Quantitative results of different approaches on Semantic3D (*semantic-8*) [15]. This test consists of 2,091,952,018 points. The scores are obtained from the recent publications. Bold represents the best result in weakly-supervised methods, and underlined represents the best results in fully-supervised methods. Accessed on 17 March 2021.

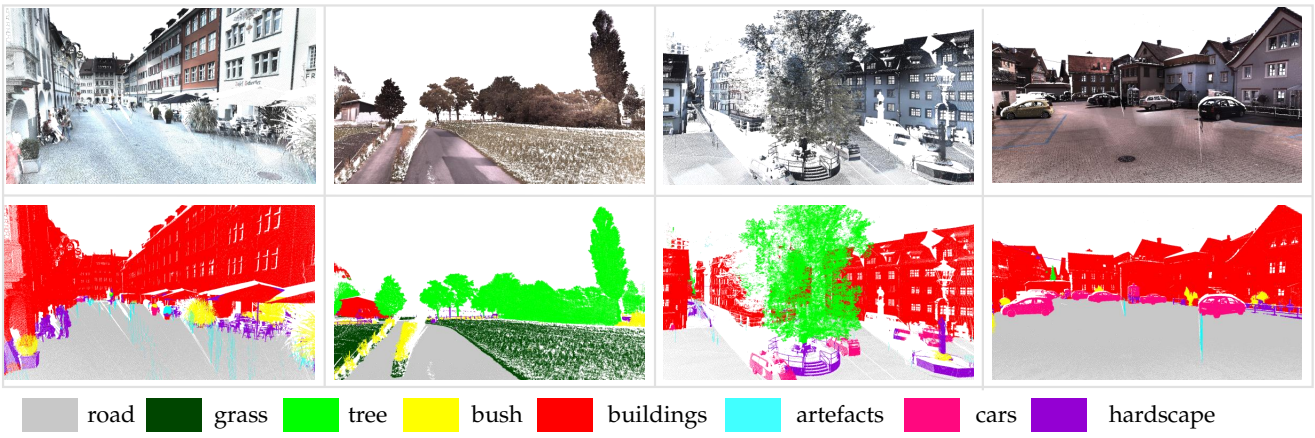

Figure 6: Qualitative results achieved by our SQN on the reduced-8 split of Semantic3D. Note that, the ground truth of the test set is not publicly available.

Settings	Methods	OA(%)	mAcc(%)	**mIoU(%)**	*ground*	*veg.*	*building*	*wall*	*bridge*	*parking*	*rail*	*traffic.*	*street.*	*car*	*footpath*	*bike*	*water*
Full supervision	PointNet [41]	80.78	30.32	23.71	67.96	89.52	80.05	0.00	0.00	3.95	0.00	31.55	0.00	35.14	0.00	0.00	0.00
	PointNet++ [42]	84.30	39.97	32.92	72.46	94.24	84.77	2.72	2.09	25.79	0.00	31.54	11.42	38.84	7.12	0.00	56.93
	TagentConv [56]	76.97	43.71	33.30	71.54	91.38	75.90	35.22	0.00	45.34	0.00	26.69	19.24	67.58	0.01	0.00	0.00
	SPGraph [28]	85.27	44.39	37.29	69.93	94.55	88.87	32.83	12.58	15.77	15.48	30.63	22.96	56.42	0.54	0.00	44.24
	SparseConv [13]	88.66	63.28	42.66	74.10	97.90	94.20	63.30	7.50	24.20	0.00	30.10	34.00	74.40	0.00	0.00	54.80
	KPConv [60]	93.20	63.76	57.58	87.10	98.91	95.33	74.40	28.69	41.38	0.00	55.99	54.43	85.67	40.39	0.00	86.30
	RandLA-Net [22]	89.78	69.64	52.69	80.11	98.07	91.58	48.88	40.75	51.62	0.00	56.67	33.23	80.14	32.63	0.00	71.31
Weak supervision	**Ours (1‰)**	**90.97**	**70.84**	**53.97**	**83.41**	**98.22**	**94.22**	**48.38**	**50.84**	**40.89**	**14.53**	**50.72**	**38.48**	**75.62**	**34.03**	0.00	**72.26**
	Ours (1‱)	85.57	49.40	37.17	74.89	96.67	88.77	32.43	7.49	12.84	0.00	29.32	22.15	67.25	0.02	0.00	51.38

Table 16: Benchmark results of the baselines on our SensatUrban. Overall Accuracy (OA, %), mean class Accuracy (mAcc, %), mean IoU (mIoU, %), and per-class IoU (%) scores are reported. Bold represents the best result in weakly-supervised methods, and underlined represents the best results in fully-supervised methods.

Settings	Methods	OA(%)	**mIoU(%)**	*Road*	*Rd mrk.*	*Natural*	*Building*	*Util. line*	*Pole*	*Car*	*Fence*
Full supervision	PointNet++ [42]	84.88	41.81	89.27	0.00	69.06	54.16	43.78	23.30	52.00	2.95
	PointNet++ (MSG) [42]	92.56	59.47	92.90	0.00	86.13	82.15	60.96	62.81	76.41	14.43
	DGCNN [68]	94.24	61.79	93.88	0.00	91.25	80.39	62.40	62.32	88.26	15.81
	KPFCNN [60]	95.39	69.11	94.62	0.06	96.07	91.51	87.68	81.56	85.66	15.72
	MS-PCNN [36]	90.03	65.89	93.84	3.83	93.46	82.59	67.80	71.95	91.12	22.50
	TGNet [32]	94.08	61.34	93.54	0.00	90.83	81.57	65.26	62.98	88.73	7.85
	MS-TGNet [52]	95.71	70.50	94.41	17.19	95.72	88.83	76.01	73.97	94.24	23.64
	RandLA-Net (w/ RGB)† [22]	97.15	81.88	96.69	64.10	96.85	94.14	88.03	77.48	93.21	44.53
	RandLA-Net (w/o RGB) [22]	95.63	77.72	94.53	42.44	96.62	93.10	86.56	76.83	92.55	39.14
Weak supervision	**Ours (w/ RGB, 1‰)†**	**96.67**	**77.75**	**96.69**	**65.67**	**94.58**	**91.34**	**83.36**	**70.59**	**88.87**	**30.91**
	Ours (w/o RGB, 1‰)	92.84	69.35	93.74	16.83	92.55	89.04	82.50	63.98	88.17	28.01
	Ours (w/ RGB, 1‱)†	94.19	68.17	95.26	54.44	88.20	84.07	75.87	57.52	84.33	5.69
	Ours (w/o RGB, 1‱)	90.47	57.57	90.97	4.99	84.10	80.29	62.78	56.51	69.49	11.44

Table 17: Quantitative results of different approaches on the Toronto3D [52] dataset. The scores of the baselines are obtained from [52]. Bold represents the best result in weakly-supervised methods, and underlined represents the best results in fully-supervised methods. Accessed on 17 March 2021.

Settings	Method	OA(%)	**mIoU(%)**	*ground*	*buildings*	*cars*	*trucks*	*poles*	*power lines*	*fences*	*vegetation*
Full supervision	ShellNet [82]	96.4	57.4	96.0	95.4	32.2	39.6	20.0	27.4	60.0	88.4
	PointCNN [31]	97.2	58.4	97.5	95.7	40.6	4.80	57.6	26.7	52.6	91.7
	SuperPoint [28]	95.5	60.6	94.7	93.4	62.9	18.7	28.5	65.2	33.6	87.9
	ConvPoint [4]	97.2	67.4	96.9	96.3	75.5	21.7	40.3	86.7	29.6	91.9
	PointNet++ [42]	95.7	68.3	94.1	89.1	75.4	30.3	40.0	79.9	46.2	91.2
	KPConv [60]	97.8	81.1	97.1	96.6	85.3	41.9	75.0	95.5	63.5	94.1
	RandLA-Net [22]	97.1	80.0	97.0	93.2	83.7	43.8	59.4	94.8	71.5	96.6
	Pyramid Point [63]	98.3	83.6	97.8	97.3	88.4	47.9	77.6	96.7	67.5	95.4
Weak supervision	**Ours (1‰)**	**97.1**	**72.0**	**96.7**	**92.0**	**75.2**	**27.3**	**87.4**	**48.1**	**53.7**	**95.8**
	Ours (1‱)	95.9	60.4	95.9	90.1	57.7	12.8	75.2	32.9	24.9	93.4

Table 18: Quantitative results of different approaches on the DALES dataset. Overall Accuracy (OA, %), mean class Accuracy (mAcc, %), mean IoU (mIoU, %), and per-class IoU (%) are reported. Bold represents the best result in weakly-supervised methods, and underlined represents the best results in fully-supervised methods.

Settings	Methods	**mIoU(%)**	Params(M)	*road*	*sidewalk*	*parking*	*other-ground*	*building*	*car*	*truck*	*bicycle*	*motorcycle*	*other-vehicle*	*vegetation*	*trunk*	*terrain*	*person*	*bicyclist*	*motorcyclist*	*fence*	*pole*	*traffic-sign*
Full supervision	PointNet [41]	14.6	3	61.6	35.7	15.8	1.4	41.4	46.3	0.1	1.3	0.3	0.8	31.0	4.6	17.6	0.2	0.2	0.0	12.9	2.4	3.7
	SPG [28]	17.4	0.25	45.0	28.5	0.6	0.6	64.3	49.3	0.1	0.2	0.2	0.8	48.9	27.2	24.6	0.3	2.7	0.1	20.8	15.9	0.8
	SPLATNet [50]	18.4	0.8	64.6	39.1	0.4	0.0	58.3	58.2	0.0	0.0	0.0	0.0	71.1	9.9	19.3	0.0	0.0	0.0	23.1	5.6	0.0
	PointNet++ [42]	20.1	6	72.0	41.8	18.7	5.6	62.3	53.7	0.9	1.9	0.2	0.2	46.5	13.8	30.0	0.9	1.0	0.0	16.9	6.0	8.9
	TangentConv [56]	40.9	0.4	83.9	63.9	33.4	15.4	83.4	90.8	15.2	2.7	16.5	12.1	79.5	49.3	58.1	23.0	28.4	8.1	49.0	35.8	28.5
	LatticeNet [44]	52.2	-	88.8	73.8	64.6	25.6	86.9	88.6	43.3	12.0	20.8	24.8	76.4	57.9	54.7	34.2	39.9	60.9	55.2	41.5	42.7
	PolarNet [80]	54.3	14	90.8	74.4	61.7	21.7	90.0	93.8	22.9	40.2	30.1	28.5	84.0	65.5	67.8	43.2	40.2	5.6	61.3	51.8	57.5
	RandLA-Net [22]	55.9	1.24	90.5	74.0	61.8	24.5	89.7	94.2	43.9	47.4	32.2	39.1	83.8	63.6	68.6	48.4	47.4	9.4	60.4	51.0	50.7
	SqueezeSeg [70]	29.5	1	85.4	54.3	26.9	4.5	57.4	68.8	3.3	16.0	4.1	3.6	60.0	24.3	53.7	12.9	13.1	0.9	29.0	17.5	24.5
	SqueezeSegV2 [71]	39.7	1	88.6	67.6	45.8	17.7	73.7	81.8	13.4	18.5	17.9	14.0	71.8	35.8	60.2	20.1	25.1	3.9	41.1	20.2	36.3
	DarkNet21Seg [3]	47.4	25	91.4	74.0	57.0	26.4	81.9	85.4	18.6	26.2	26.5	15.6	77.6	48.4	63.6	31.8	33.6	4.0	52.3	36.0	50.0
	DarkNet53Seg [3]	49.9	50	91.8	74.6	64.8	27.9	84.1	86.4	25.5	24.5	32.7	22.6	78.3	50.1	64.0	36.2	33.6	4.7	55.0	38.9	52.2
	RangeNet53++ [39]	52.2	50	91.8	75.2	65.0	27.8	87.4	91.4	25.7	25.7	34.4	23.0	80.5	55.1	64.6	38.3	38.8	4.8	58.6	47.9	55.9
	SalsaNext [11]	54.5	6.73	90.9	74.0	58.1	27.8	87.9	90.9	21.7	36.4	29.5	19.9	81.8	61.7	66.3	52.0	52.7	16.0	58.2	51.7	58.0
	SqueezeSegV3 [74]	55.9	26	91.7	74.8	63.4	26.4	89.0	92.5	29.6	38.7	36.5	33.0	82.0	58.7	65.4	45.6	46.2	20.1	59.4	49.6	58.9
Weak supervision	**Ours (1‰)**	**50.8**	1.05	**90.5**	**72.9**	**56.8**	**19.1**	**84.8**	**92.1**	**36.7**	**39.3**	**30.1**	**26.0**	**80.8**	**59.1**	**67.0**	**36.4**	**25.3**	**7.2**	**53.3**	**44.5**	**44.0**
	Ours (1‱)	39.1	1.05	86.6	66.4	43.0	16.9	80.0	85.5	12.9	4.0	1.4	18.4	72.7	49.6	58.8	16.9	22.3	4.3	42.3	31.7	16.6

Table 19: Quantitative results of different approaches on SemanticKITTI [3]. The scores are obtained from the recent publications. Bold represents the best result in weakly-supervised methods, and underlined represents the best results in fully-supervised methods. Accessed on 17 March 2021.

Figure 7: Qualitative results achieved by our SQN on the validation set (Sequence 08) of SensatUrban[21] dataset. Best viewed in color.

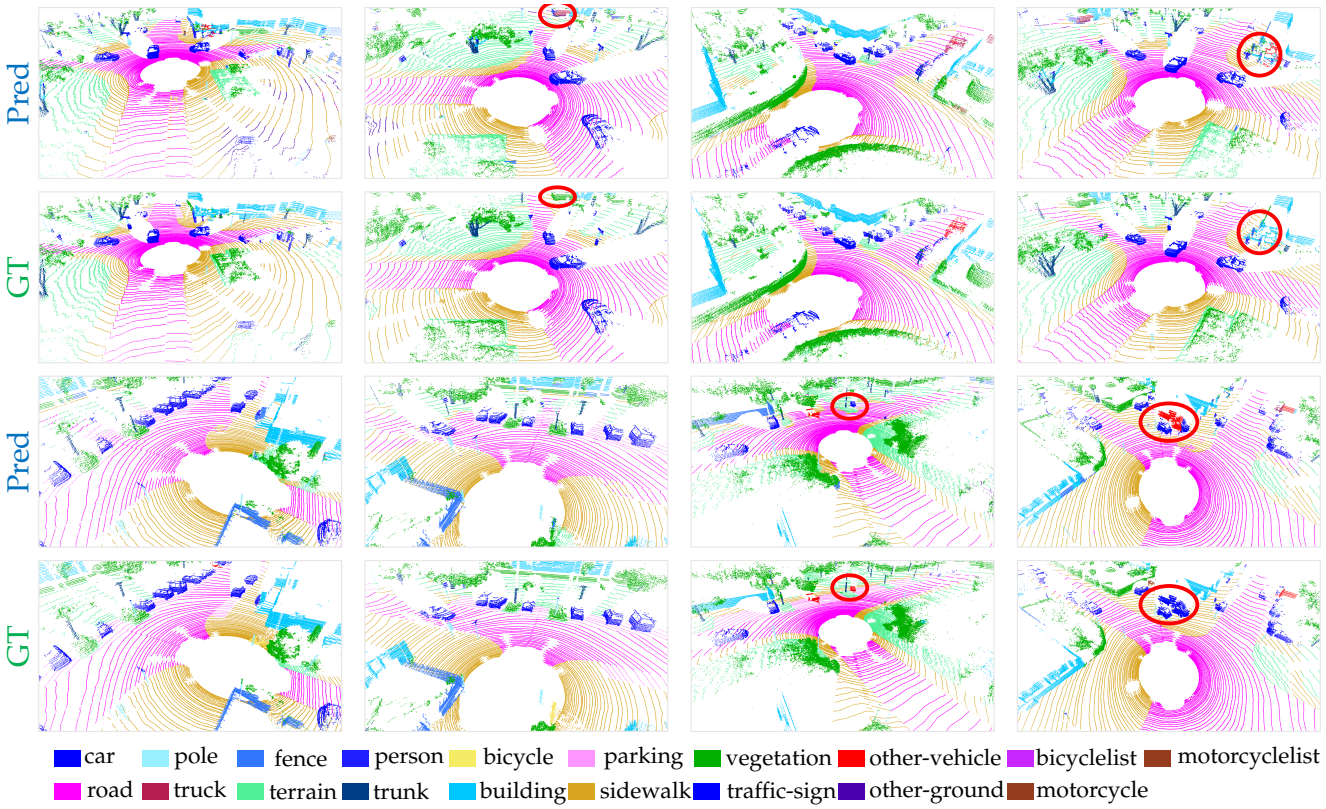

Figure 8: Qualitative results achieved by our SQN on the validation set (Sequence 08) of SemanticKITTI [3] dataset. The red circle highlights the failure case.