Appendix

7 Details of Sparse Annotation Tool

(1) Annotation Pipeline. As mentioned in Section 3, we develop a user-friendly annotation pipeline based on the off-the-shelf software. Note that, this tool is important to justify the feasibility/suitability of the low-cost random sparse annotation scheme, as most existing methods have directly overlooked this or taken it for granted that such tool is available. Here, we provide more detailed information on the pipeline. Specifically, given large-scale raw point clouds, the sparse annotation pipeline could be generally divided into the following steps:

1. Load the raw point clouds;
2. Random downsample to a specified ratio (*e.g.,* 0.1%);
3. Increase the point size of the downsampled points;
4. Visualize the original point cloud and the down-sampled point cloud simultaneously;
5. Annotate downsampled points in polygonal edition mode;
6. Refine point labels.

We also provide a video illustrating the annotation pipeline, which can be viewed at: https://youtu.be/NOUAeY31msY.

(2) The Number of Annotated Points on 7 Datasets. Considering that the existing large-scale point cloud datasets usually have millions of points, and typically have relatively high density, we therefore follow [66,24] to perform grid downsampling of raw point clouds at the beginning, and then execute the random based annotation steps in practice. Note that, all experiments of our SQN on the seven public datasets follow this setting. As shown in Table 9, the grid downsampling at the beginning can significantly reduce the number of raw points. Taking the Semantic3D dataset which has high density as an example, the total number of points after grid downsampling becomes 1/50 of the original point clouds. Following the 0.1% sparse annotation pipeline in our SQN, the total number of annotated points is only 78100, which is an approximately 0.002% of the total raw points. To avoid confusion, we still report the 0.1% labeling ratio in the main paper to keep consistency (*i.e.,* the number of annotated points after grid downsampling / the total number of points after grid downsampling). Importantly, this is significantly different from 1T1C [37] and cannot be directly compared, which calculates its labeling ratio by using the number of labeled instances divided by the total number of points, so as to achieve an over-exaggerated labeling ratio.

(3) Annotation Cost. The sparse annotation scheme used in our SQN can greatly reduce the annotation cost in practice, especially for extremely large-scale 3D point clouds with billions of points. Taking 0.1% sparse point annotation as an example, with the developed CloudComapre-based labelling tool, a professional annotator can finish the annotation of the whole SensatUrban [23] dataset within **16** hours. By comparison, the original dense point-wise labeling

	Grid size	Raw pts	Grid sampled pts	Anno. pts (0.1%)
S3DIS [2]	0.04	273M	18.6M	18,600
Semantic3D [18]	0.06	4000M	78.1M	78,100
ScanNet [68]	0.04	242M	60.2M	60,200
SemanticKITTI [3]	0.06	5299M	3401M	3.4M
DALES [68]	0.32	505M	211M	211,000
SensatUrban [23]	0.2	2847M	221M	221,000
Toronto3D [58]	0.04	78.3M	24.3M	24,300

Table 9: A comparison of the total number of points (M: Million) before and after grid sampling for seven public datasets. The grid size and the number of actual annotated points under our 0.1% supervision setting are also reported.

costs **600** person-hours. Primarily, **this is because the random annotation based pipeline offers great error tolerance to avoid annotating boundary areas** (as only a very small number of points fall on the boundary), hence advanced functions such as polygonal edition can be freely and flexibly use, finally improve the productivity. In the traditional dense labeling pipeline, annotators are usually required to rotate and zoom back and forth to accurately separate the boundary areas, which consumes most of the labeling time. However, the random annotation based pipeline used in our developed tool can greatly reduce such time-consuming labelling of boundary areas, hence greatly reducing the overall annotation cost. Note that, the annotation cost (*i.e.,* the total annotation time) could be further reduced if more advanced annotation software such as QTModeler[8] is used, where the user interface is more friendly.

8 Implementation Tricks

(1) Data augmentation. We follow [87] 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 [18]) 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 (0.1%) are also different (18600 *vs.* 78100, as reported in Table 9). 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 0.1% points, and then 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 large-scale datasets including Semantic3D [18], SensatUrban [23], SemanticKITTI [3],

[8] https://appliedimagery.com/

DALES [68] in Section 5.2, our SQN can achieve satisfactory results trained with 0.1% annotated points, while the retraining trick does not noticeably improve the performance. Advanced techniques such as pseudo label refining [91] will be further explored in future work.

(3) Code Release. The code is available at: https://github.com/QingyongHu/SQN.

9 Video Illustration

We also provide a video illustrating the proposed SQN, which can be viewed at https://youtu.be/Q6wICSRRw3s.

10 Additional Ablation Results

(1) Varying backbones of our SQN framework. To further study the performance of our SQN framework with different backbones, we further implement our SQN based on the representative voxel-based baseline MinkowskiNet [11]. Specifically, we follow the implementation provided in [59], and the point local feature extractor, in this case, includes 4 encoding layers, each containing a 3D convolution block (kernel size and stride are set as 2) and 2 residual blocks (kernel size and stride are set as 3 and 1, respectively). Additionally, the feature query network gathers feature vectors from multi-level feature volumes through trilinear interpolation, and then simply concatenated and sent to MLPs (256-128-96) for semantic prediction.

The experimental results achieved by our SQN and baseline networks on the SemanticKITTI dataset under different weak supervision settings are shown in Table 10. We can see that our SQN achieves comparable performance with the baseline under 0.1% settings, primarily because the supervision signal is still sufficient at this time, considering the large scale of the dataset. However, we can clearly observe that our SQN outperforms the baseline by a large margin (6.8% improvement in mIoU scores) when there are only 0.01% points are annotated. This further demonstrates the effectiveness of our semantic query framework.

(2) Detailed Results on Varying Annotated Points. In Section 5.3, we evaluate the sensitivity of the proposed SQN to different randomly annotated points. Here, we provide detailed experimental results on Table 11. It can be seen that the major performance variations are in minor categories such as *door*, *sofa*, and *board*, indicating that the underrepresented categories are more sensitive to our weakly-supervised settings, *i.e.,* 0.1% random annotated point labels. This is not surprising because such imbalanced distribution issue also widely exists in fully-supervised methods.

11 Additional Discussion

(1) Performance on Boundary Areas. We further evaluate the segmentation performance of our SQN on the boundary points, since the assumption about

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*
MinkUNet 0.1%	55.5	21.9	92.5	79.0	43.0	0.9	88.7	95.0	64.5	0.9	47.4	46.4	87.3	63.4	73.7	45.3	70.3	0.3	53.4	59.9	43.2
SQN (MinkUNet) 0.1%	55.8	8.8	91.5	78.0	41.1	0.9	88.5	94.9	66.8	5.7	43.2	43.6	88.2	64.0	75.5	49.5	66.3	0.0	55.9	61.1	45.2
MinkUNet 0.01%	43.2	21.9	89.3	74.8	32.1	0.0	87.6	92.4	25.8	0.0	24.8	20.1	87.1	56.4	73.2	9.6	15.9	0.0	55.1	48.2	28.3
SQN (MinkUNet) 0.01%	50.0	8.8	89.7	75.6	31.9	0.2	87.6	93.5	47.2	0.2	35.6	31.6	88.2	58.0	76.0	33.8	59.1	0.0	52.9	52.1	36.4

Table 10: Quantitative results achieved by our SQN (MinkUNet) on the validation set of the SemanticKITTI dataset under different weak supervision settings.

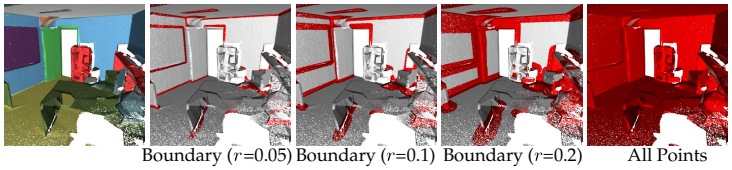
	OA(%)	**mIoU(%)**	*ceil.*	*floor*	*wall*	*beam*	*col.*	*wind.*	*door*	*table*	*chair*	*sofa*	*book.*	*board*	*clut.*
Iter1	**86.53**	60.97	**92.33**	96.70	**78.99**	0.00	25.01	**56.76**	58.99	**74.22**	79.06	58.41	67.73	53.29	51.08
Iter2	85.63	59.24	91.72	**97.01**	77.35	0.00	20.10	53.55	**65.28**	71.63	**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.

the consistency of neighborhood semantics may not hold at the boundary areas with different semantics. Specifically, we first define the boundary points as follows: if the queried spherical neighboring points within a radius r have different semantics, the query point is regarded as on the boundary (red points in the Figure 6). Not surprisingly, given a smaller r, the performance drops significantly for both RandLA-Net (full supervision) and SQN (0.1%), showing that it is still a common issue for existing methods. We will leave this issue for future exploration.

(2) Flexibility of SQN. Thanks to the flexibility of the SQN framework, the proposed method should be able to take any point in the space as input and infer its semantic label through query and interpolation (even if that point itself does not exist in the point clouds). To further validate this, we tried to train our SQN on the partial point clouds (*i.e.*, raw point clouds), but test on the aggregated point clouds based on the SemanticKITTI dataset. The qualitative results are shown in Figure 7. It can be seen that the proposed SQN can still achieve satisfactory performance on the complete point clouds, even though our model is only trained with partial and incomplete point clouds. Considering that point clouds are irregularly sampled points from the continuous surface, it would be interesting to further explore the continuous semantic surface learning based on our framework.

(3) Potential Negative Societal Impact. Our work aims to achieve label-efficient learning of large-scale 3D point clouds, which could potentially be used in autonomous driving or robotics systems. It is targeted for semantic segmentation of 3D point clouds with weak supervision, hence there is no known society



	Boundary (r=0.05)	Boundary (r=0.1)	Boundary (r=0.2)	All Points
RandLA (Full sup.)	38.3	46.6	53.7	63.0
SQN (0.1%)	34.3	42.7	50.5	61.4

Fig. 6: Quantitative comparison of RandLA-Net and the proposed SQN on the boundary areas.

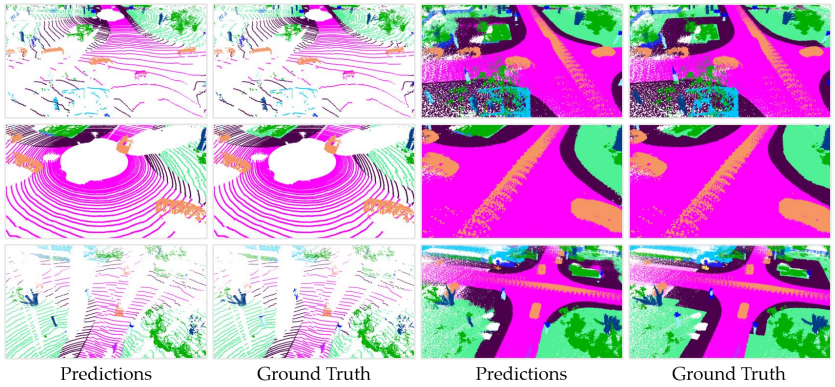


Fig. 7: Qualitative performance achieved by SQN on the SemanticKITTI dataset when trained on partial point clouds.

negative impact. However, the robustness, security, safety issues should be further checked before being applied in real-world data.

(4) Limitation and Future Work. Our SQN is intuitive simple yet effective. Extensive experiments have demonstrated the superiority on large-scale datasets. However, it still relies on human annotations, albeit extremely sparse. Ideally, the 3D semantics can be automatically discovered from raw point clouds. We leave this unsupervised learning of 3D semantic segmentation for our future exploration.

12 Additional Experimental Results

(1) Detailed Results of Fully Supervised Baselines under Sparse Annotations. As mentioned in Section 3, we evaluate several baseline methods under different forms of weak supervision. Here, we further provide the detailed benchmarking results on Table 12, with per-class IoU scores reported. Note that, this table is corresponding to Figure 2 in the main paper.

Settings	Methods	**mIoU(%)**	*ceil.*	*floor*	*wall*	*beam*	*col.*	*win.*	*door*	*chair*	*table*	*book.*	*sofa*	*board*	*clutter*
100% (Full supervision)	PointNet [46]	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++ [47]	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 [24]	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 [46]	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++ [47]	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 [24]	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 [46]	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++ [47]	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 [24]	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
0.1% (Random)	PointNet [46]	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++ [47]	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 [24]	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
0.01% (Random)	PointNet [46]	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++ [47]	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 [24]	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 12: 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.

	Methods	OA(%)	mAcc(%)	**mIoU(%)**	*ceil.*	*floor*	*wall*	*beam*	*col.*	*wind.*	*door*	*table*	*chair*	*sofa*	*book.*	*board*	*clut.*
Full supervision	PointNet [46]	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 [25]	-	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 [90]	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 [31]	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 [34]	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 [99]	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 [98]	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 [89]	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*) [66]	-	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*) [66]	-	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 [24]	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 (0.1%)**	**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 13: 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.

(2) 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 13.

(3) Additional Results on ScanNet. The ScanNet [14] 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. We provided the detailed per class IoU results on Table 14.

(4) Additional results 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 0.01% randomly annotated points, considering the extremely large amount of 3D points scanned. The detailed experimental results achieved on the *Semantic8* and *Reduced8* subset of the Semantic3D dataset are reported in Table 15 and Table 16. In addition, we also show the qualitative results achieved by our SQN on the *Reduced-8* subset with 0.1% labels in Fig 8.

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 [14]	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++ [47]	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 [56]	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 [62]	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 [34]	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 [83]	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 [33]	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 [66]	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 [16]	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 [32]	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 [24]	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
Weak sup.	**Ours (0.1%)**	**56.9**	**67.6**	**69.6**	**65.7**	**49.7**	**77.9**	**42.4**	**54.8**	**51.5**	**37.6**	**90.2**	**42.2**	**35.7**	**37.9**	**45.6**	**59.6**	**65.9**	**54.4**	**68.5**	**66.5**	**55.6**

Table 14: Quantitative results of different approaches on ScanNet (*online test set*). Mean IoU (mIoU, %), and per-class IoU (%) scores are reported.

Settings	Methods	**mIoU(%)**	OA(%)	*man-made.*	*natural.*	*high veg.*	*low veg.*	*buildings*	*hard scape*	*scanning art.*	*cars*
Full supervision	TML-PC [44]	39.1	74.5	80.4	66.1	42.3	41.2	64.7	12.4	0.0	5.8
	TMLC-MS [19]	49.4	85.0	91.1	69.5	32.8	21.6	87.6	25.9	11.3	55.3
	PointNet++ [47]	63.1	85.7	81.9	78.1	64.3	51.7	75.9	36.4	43.7	72.6
	EdgeConv [12]	64.4	89.6	91.1	69.5	65.0	56.0	89.7	30.0	43.8	69.7
	SnapNet [4]	67.4	91.0	89.6	79.5	74.8	56.1	90.9	36.5	34.3	77.2
	PointGCR [41]	69.5	92.1	93.8	80.0	64.4	66.4	93.2	39.2	34.3	85.3
	RGNet [67]	72.0	90.6	86.4	70.3	69.5	68.0	96.9	43.4	52.3	89.5
	LCP [6]	74.6	94.1	94.7	85.2	77.4	70.4	94.0	52.9	29.4	92.6
	SPGraph [31]	76.2	92.9	91.5	75.6	78.3	71.7	94.4	56.8	52.9	88.4
	ConvPoint [5]	76.5	93.4	92.1	80.6	76.0	71.9	95.6	47.3	61.1	87.7
	RandLA-Net [24]	75.8	95.0	97.4	93.0	70.2	65.2	94.4	49.0	44.7	92.7
	WreathProdNet [76]	77.1	94.6	95.2	87.1	75.3	67.1	96.1	51.3	51.0	93.4
Weak supervision	**Ours (0.1%)**	**72.3**	**94.8**	**97.9**	**93.2**	**65.5**	**63.4**	**94.9**	**44.9**	**47.4**	**70.9**
	Ours (0.01%)	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*) [18]. 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.

(5) Additional Results 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 imbalanced class distributions. The detailed experimental results achieved on the SensatUrban dataset are reported in Table 17. In addition, we also show the qualitative results achieved by our SQN trained with only 0.1% labels on this dataset in Fig. 9.

(6) Additional Results 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. Here, we provide the quantitative comparison of our SQN and several fully-supervised methods in Table 18. Following [24], 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 KPConv, with merely 0.1% of point annotations for training. We also notice that the usage of color information closes the gap between our method and the top-performing RandLA-Net [24]. This implies that it could be helpful to introduce auxiliary information under the setting of weak supervision.

(7) Additional Results on DALES. This dataset consists of large-scale earth scans acquired by an aerial LiDAR. It covers over 10 km^2 spatial ranges

Settings	Methods	**mIoU(%)**	OA(%)	*man-made.*	*natural.*	*high veg.*	*low veg.*	*buildings*	*hard scape*	*scanning art.*	*cars*
Full supervision	SnapNet_ [4]	59.1	88.6	82.0	77.3	79.7	22.9	91.1	18.4	37.3	64.4
	SEGCloud [63]	61.3	88.1	83.9	66.0	86.0	40.5	91.1	30.9	27.5	64.3
	RF_MSSF [65]	62.7	90.3	87.6	80.3	81.8	36.4	92.2	24.1	42.6	56.6
	MSDeepVoxNet [51]	65.3	88.4	83.0	67.2	83.8	36.7	92.4	31.3	50.0	78.2
	ShellNet [98]	69.3	93.2	96.3	90.4	83.9	41.0	94.2	34.7	43.9	70.2
	GACNet [74]	70.8	91.9	86.4	77.7	88.5	60.6	94.2	37.3	43.5	77.8
	SPG [31]	73.2	94.0	97.4	92.6	87.9	44.0	83.2	31.0	63.5	76.2
	KPConv [66]	74.6	92.9	90.9	82.2	84.2	47.9	94.9	40.0	77.3	79.9
	RGNet [67]	74.7	94.5	97.5	93.0	88.1	48.1	94.6	36.2	72.0	68.0
	RandLA-Net [24]	77.4	94.8	95.6	91.4	86.6	51.5	95.7	51.5	69.8	76.8
Weak supervision	**Ours (0.1%)**	**74.7**	**93.7**	**97.1**	**90.8**	**84.7**	**48.5**	**93.9**	**37.4**	**71.0**	**74.5**
	Ours (0.01%)	65.6	90.3	96.6	87.5	80.6	37.1	88.5	16.9	56.6	60.9

Table 16: Quantitative results of different approaches on Semantic3D (*reduced-8*) [18]. 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.

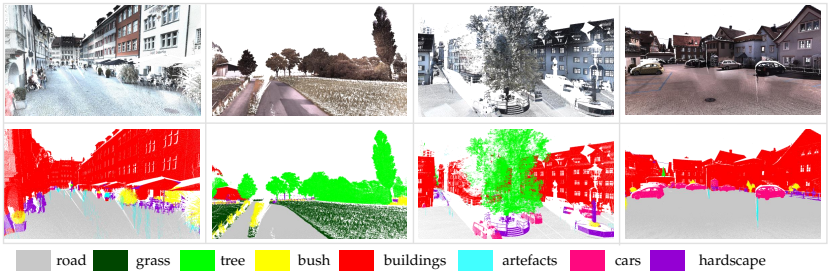


Fig. 8: Qualitative results achieved by our SQN on the reduced-8 split of Semantic3D. Note that, the ground truth of the test set is not publicly available.

with 5 million points belonging to 8 semantic categories. We compare our SQN with strong fully-supervised approaches. As shown in Table 19, our method achieves higher mIoU scores than PointNet++ [47], ConvPoint [6], SPGraph [31], PointCNN [34] and ShellNet [98], with only 0.1% labels for training. However, there is still a performance gap compared with the leading fully-supervised counterparts such as RandLA-Net [24], primarily due to our weak performance on minor categories such as *trucks* and *cars*. The potential reason is that the simple random annotation strategy may happen to ignore the underrepresented classes.

(8) Additional Results 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 with fully-supervised techniques on the online test set in Table 20. It can be seen that our approach achieves a satisfactory mIoU score, outperforming several strong baselines with only 0.1% 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(%)	mAcc(%)	**mIoU(%)**	*ground*	*veg.*	*building*	*wall*	*bridge*	*parking*	*rail*	*traffic.*	*street.*	*car*	*footpath*	*bike*	*water*
Full supervision	PointNet [46]	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++ [47]	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 [62]	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 [31]	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 [16]	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 [66]	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 [24]	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 (0.1%)**	**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 (0.01%)	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 17: 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++ [47]	84.88	41.81	89.27	0.00	69.06	54.16	43.78	23.30	52.00	2.95
	PointNet++ (MSG) [47]	92.56	59.47	92.90	0.00	86.13	82.15	60.96	62.81	76.41	14.43
	DGCNN [77]	94.24	61.79	93.88	0.00	91.25	80.39	62.40	62.32	88.26	15.81
	KPFCNN [66]	95.39	69.11	94.62	0.06	96.07	91.51	87.68	81.56	85.66	15.72
	MS-PCNN [40]	90.03	65.89	93.84	3.83	93.46	82.59	67.80	71.95	91.12	22.50
	TGNet [35]	94.08	61.34	93.54	0.00	90.83	81.57	65.26	62.98	88.73	7.85
	MS-TGNet [58]	95.71	70.50	94.41	17.19	95.72	88.83	76.01	73.97	94.24	23.64
	RandLA-Net (w/ RGB)† [24]	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) [24]	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, 0.1%)†**	**96.67**	**77.75**	**96.69**	**65.67**	**94.58**	**91.34**	**83.36**	**70.59**	**88.87**	**30.91**
	Ours (w/o RGB, 0.1%)	92.84	69.35	93.74	16.83	92.55	89.04	82.50	63.98	88.17	28.01
	Ours (w/ RGB, 0.01%)†	94.19	68.17	95.26	54.44	88.20	84.07	75.87	57.52	84.33	5.69
	Ours (w/o RGB, 0.01%)	90.47	57.57	90.97	4.99	84.10	80.29	62.78	56.51	69.49	11.44

Table 18: Quantitative results of different approaches on the Toronto3D [58] dataset. The scores of the baselines are obtained from [58]. Bold represents the best result in weakly-supervised methods, and underlined represents the best results in fully-supervised methods.

world applications. Finally, we also visualize the segmentation results achieved by our SQN on the validation set of the SemanticKITTI dataset in Fig. 10.

Settings	Method	OA(%)	**mIoU(%)**	*ground*	*buildings*	*cars*	*trucks*	*poles*	*power lines*	*fences*	*vegetation*
Full supervision	ShellNet [98]	96.4	57.4	96.0	95.4	32.2	39.6	20.0	27.4	60.0	88.4
	PointCNN [34]	97.2	58.4	97.5	95.7	40.6	4.80	57.6	26.7	52.6	91.7
	SuperPoint [31]	95.5	60.6	94.7	93.4	62.9	18.7	28.5	65.2	33.6	87.9
	ConvPoint [5]	97.2	67.4	96.9	96.3	75.5	21.7	40.3	86.7	29.6	91.9
	PointNet++ [47]	95.7	68.3	94.1	89.1	75.4	30.3	40.0	79.9	46.2	91.2
	KPConv [66]	97.8	81.1	97.1	96.6	85.3	41.9	75.0	95.5	63.5	94.1
	RandLA-Net [24]	97.1	80.0	97.0	93.2	83.7	43.8	59.4	94.8	71.5	96.6
	Pyramid Point [69]	98.3	83.6	97.8	97.3	88.4	47.9	77.6	96.7	67.5	95.4
Weak supervision	**Ours (0.1%)**	**97.1**	**72.0**	**96.7**	**92.0**	**75.2**	**27.3**	**87.4**	**48.1**	**53.7**	**95.8**
	Ours (0.01%)	95.9	60.4	95.9	90.1	57.7	12.8	75.2	32.9	24.9	93.4

Table 19: 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 [46]	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 [31]	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 [56]	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++ [47]	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 [62]	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 [50]	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 [95]	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 [24]	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 [80]	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 [81]	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++ [43]	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 [13]	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 [86]	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 (0.1%)**	**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 (0.01%)	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 20: 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.

Fig. 9: Qualitative results achieved by our SQN on the validation set (Sequence 08) of SensatUrban[23] dataset. Best viewed in color.

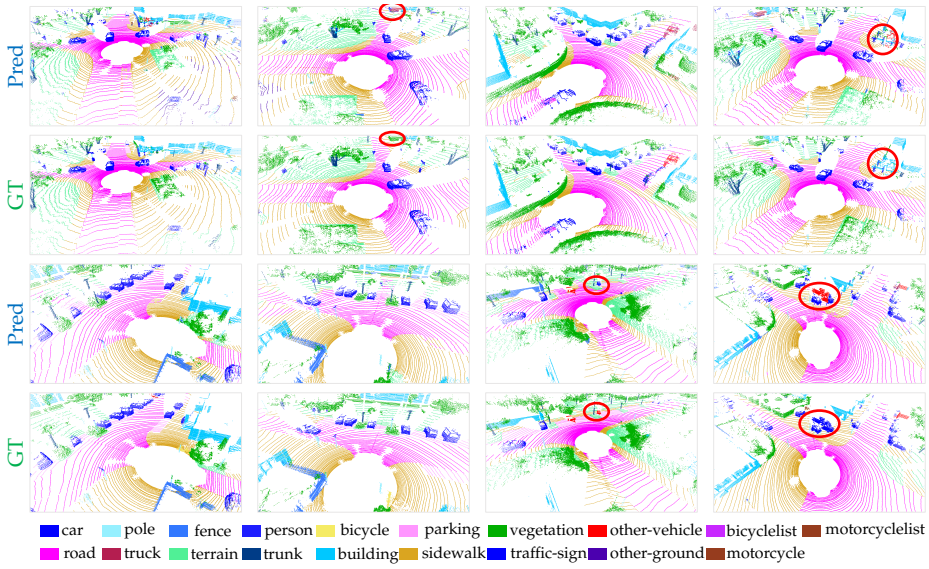


Fig. 10: Qualitative results achieved by our SQN on the validation set (Sequence 08) of SemanticKITTI [3] dataset. The red circle highlights the failure case.