Point Cloud Semantic Segmentation with Cross-Correction Features

Yuehua Zhao (cassiel0406@gmail.com)
Hebei University of Technology https://orcid.org/0000-0003-3475-2545
Ma Jie
Hebei University of Technology
Chong Nannan
Tianjin University Renai College
Wen Junjie
Hebei University of Technology

Research Article

Keywords: Point cloud, Semantic segmentation, Spatial geometric, Semantic context, Cross-correction

Posted Date: January 10th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1218117/v1

Point Cloud Semantic Segmentation with Cross-correction Features

Zhao Yuehua[1], Ma Jie[1*], Chong Nannan[1,2] and Wen Junjie[1]

[1*]School of Electronics and Information Engineering, Hebei University of Technology, Xiping Road, Tianjin, 300401, Tianjin, China.

[2]School of Information and Communication Engineering, Tianjin University Renai College, Beian Road, Tianjin, 300401, Tianjin, China.

*Corresponding author(s). E-mail(s): jma@hebut.edu;
Contributing authors: cassiel0406@gmail.com;
chongnannan@163.com; q1403501@163.com;

Abstract

Real time large scale point cloud segmentation is an important but challenging task for practical application like autonomous driving. Existing real time methods have achieved acceptance performance by aggregating local information. However, most of them only exploit local spatial information or local semantic information dependently, few considering the complementarity of both. In this paper, we propose a model named Spatial-Semantic Incorporation Network (SSI-Net) for real time large scale point cloud segmentation. A Spatial-Semantic Cross-correction (SSC) module is introduced in SSI-Net as a basic unit. High quality contextual features can be learned through SSC by correct and update semantic features using spatial cues, and vice verse. Adopting the plug-and-play SSC module, we design SSI-Net as an encoder-decoder architecture. To ensure efficiency, it also adopts a random sample based hierarchical network structure. Extensive experiments on several prevalent datasets demonstrate that our method can achieve state-of-the-art performance.

Keywords: Point cloud, Semantic segmentation, Spatial geometric, Semantic context, Cross-correction

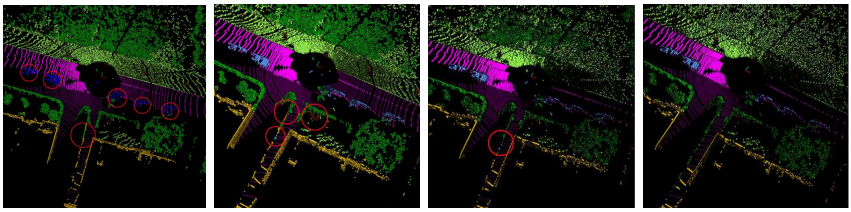

Fig. 1 Semantic predictions. From left to right are the results of SPG[35], RandLA[10], SSI-Net, and Ground truth.

1 Introduction

Recently, high-quality 3D scanners and depth sensors are available to many agents like self-driving cars and robots, making it possible to utilize the point cloud data collected from these sensors to assist many downstream tasks. Among these downstream tasks, point cloud segmentation that predicts a classification score for each point has attracted significant research interests, mainly because that the segmentation result plays a basic and critical role in providing self-driving cars or robots with scene-level understandings.

In recent years, researches [1, 3–8, 10, 35] have shown great success in terms of semantic segmentation using deep learning models. Early works [3–8] focus on researching how to segment small scale point clouds such as object surface sampled from CAD models. Though have achieved promising results, they are not suitable for large scale point clouds collected from in-the-wild scene by advanced sensors, due to both poor-effectiveness and poor-efficiency. SPG[35] is a pioneering work that tailored for large scale point cloud segmentation. However, it would cause huge computational cost, make it impossible to achieve real time performance. Therefore, RandLANet[10] is proposed. This method proposes to randomly down sample the point cloud at each layer to ensure the efficiency of the model, and the segmentation accuracy is kept by leaning powerful contextual local features using some delicately designed local feature aggregation modules. However, it neglects the complementarity between spatial information and semantic information, i.e, it only exploits local spatial information or local semantic information dependently. Therefore, the model would always prediction wrong segmentation results at some ambiguous regions or loss local details, as shown in Fig. 1.

To tackle the above mentioned problems, this paper proposes a novel model Spatial-Semantic Incorporation Network (SSI-Net) for real time large scale point cloud segmentation. SSI-Net aims at efficiently learning discriminative geometric and semantic feature representation. Specifically, to better incorporate spatial and semantic information, we inspect two questions: 1) how to encode geometric patterns with the guidance of semantic information, and 2) how to combine both information to aggregate most positive features. To this end, a Spatial-Semantic Cross-correction (SSC) module is introduced in SSI-Net as a basic unit. High quality contextual features can be learned through

SSC by correct and update semantic features using spatial cues, and vice verse. More specifically, to learn high quality semantic features, it first performs k-NN locally in spatial space to find K candidate neighboring points and generates neighborhood deformation based on feature distance. The K neighbors, center points and corresponding deformation form a cluster to encode geometric structures. Finally, an attention block is introduced to append spatial disparities to semantic information to restrain the attention weights and selects the most positive neighboring features. The spatial features can be updated by the same way. Adopting the plug-and-play SSC module, we design SSI-Net as an encoder-decoder architecture. We also adopt the Random sampling strategy to ensure run time efficiency, so that our model can achieve real time performance. We conduct extensive experiments on several public datasets, and experiments show our model can achieve state-of-the-art performance in terms of both segmentation accuracy and running time efficiency.

Our main contributions are:

- We propose a novel model named SSI-Net for real time large scale point cloud segmentation , which achieves state-of-the-art performance on several challenging public benchmark datasets.
- We propose the Spatial-Semantic Cross-Correction (SSC) module, which can delicately learns high quality contextual features by incorporating and aggregating semantic features and spatial features in the latent space.
- We conduct extensive experiments on both indoor and outdoor datasets to demonstrate the efficiency and effectiveness of our proposed network.

2 Related Work

This part makes a simple list of the point cloud analysis development mainly based on deep learning methods. Besides, we emphasize some works that are related to large scale point cloud.

2.1 Deep Learning on Point Cloud

Deep learning has immensely promoted the progress of 2D and 3D computer vision. This segment presents a brief introduction of the four following main deep learning approaches to point cloud.

Multiview-based methods: Deep learning methods first designed for image processing cannot be directly applied to point clouds. As the early way, multiview-based methods reduce the data dimension and represent 3D data by a set of their rendered views on 2D images, which allows the application of 2D CNNs and resolves the unstructured problem of point clouds. The most well-behaved multiview-based method is Multi-view CNN [11] for 3D shape recognition. SnapNet [12] is one example that uses multiview-based method to deal with 3D semantic segmentation. Approaches in this category have proven effective in dealing with unstructured problems related to point clouds; however, the transformation process leads to geometrical information loss (i.e.,

2D images cannot fully express the 3D structures). Moreover, to cover an entire scene with a number of virtual viewpoints is not easy. As a result, multiview-based deep learning architectures are seldom used for semantic segmentation.

Voxel-based methods: Voxel-based methods intend to address unordered and unstructured problems simultaneously via transforming point cloud into voxel grid and applying 3D CNNs directly. VoxNet [13] is a representative voxel-based method for detection applications. One well-known voxel-based deep learning architecture for semantic segmentation is SegCloud [14] which utilizes a preprocessing step to voxelize point clouds to adapt the 3D fully convolutional neural network. Other works such as [15–17] also propose some typical networks for semantic segmentation. Unfortunately, this kind of approaches introduces artifacts and information loss. And the storage scheme lacks efficiency in terms of computation and memory usage. With the appearance of methods directly on point clouds, both multiview-based and voxel-based architectures eclipse more or less.

Pointwise MLP methods: As the first job to process point clouds without any formal transformation, PointNet [2] takes advantage of some symmetric operations, i.e. max-pooling and MLPs, to learn point features individually which guarantees the fundamental properties of point clouds. However, such a brilliant idea at that time does not capture the contextual features from local neighborhood. To further improve their research, Qi et al. propose PointNet++ [18] to perform mini-pointnet in groups. At the same period, methods [19–21] spring up to specify features of each point based on local neighbouring connection for better representation. Although a large amount of approaches has been proposed, the idea of PointNet[2] remains the standard.

Graph-based methods: The advances and difficulties of current research inspire the combination of graph concept and point cloud analysis. Approaches of graph convolutional networks (GCNs) first build graphs $G(V, E)$ and then conduct convolution on graphs that have been proven to be suitable for non-Euclidean data. For example, Wang et al.[6] constructs a local neighborhood graph and applies edge convolution (EdgeConv) in which the graph is not fixed but dynamically updated after each layer. Next, Li et al.[7] utilizes residual/dense connections and dilated convolutions to realize deep GCN which breaks the bottleneck that GCNs are limited to very shallow models due to the vanishing gradient problem.

2.2 Large-scale point cloud semantic segmentation

Many tasks need to deal with large scale scenes such as autonomous driving and remote sensing. Approaches [10, 22, 23, 35] have explored large-scale point cloud analysis. SPG[35] uses superpoint graph structure to tackle the challenge of semantic segmentation of millions of points. In addition to this structured format of point clouds, Voxel-based representation has been applied to some networks[22, 23] for large-scale semantic segmentation. However, these representations require a huge amount of computation. The recent RandLA-Net[10]

built by point representation learning with MLPs reaches considerable performances. However, it encodes local features simply with Euclidean distance based K nearest neighbors neglecting the interaction between geometric and semantic context which may limit the capability in capturing more positive representation.

3 Proposed method

As an important data source, one of the overwhelming traits of a point cloud is its adequate position information. Thus, the spatial geometric relationship has been targeted as the major element to encode local features. Besides, some work about 2D semantic segmentation emphasizes the role of semantic context. Based on this observation, this article proposes the Semantic-Spatial Cross-correction (SSC) module to mine the local patterns not only with the spatial positions but also the semantic expression.

3.1 Spatial-Semantic Cross-correction module

Fig. 2 shows the structure of our SSC module which can be decomposed into two parts: a semantic-aware spatial block to encode geometric relationship and an attention block to extract information in feature space.

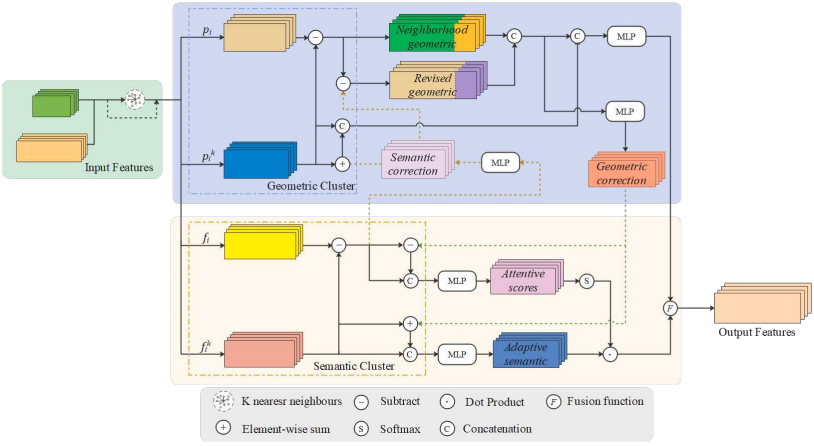

Fig. 2 Structure of the proposed Spatial-Semantic Cross-correction module.

3.1.1 Semantic-aware spatial block

Given a point cloud with n points, their coordinates and semantic information can be denoted as $P = \{p_1, ..., p_i, ..., p_n\} \subset R^3$ and $F = \{f_1, ..., f_i, ..., f_n\} \subset R^N$ respectively. As shown in Fig. 2, we firstly perform k-nearest neighboring (kNN) to search K candidate neighbourhoods denoted as $\{p_i^1, ..., p_i^k, ..., p_i^K\}$ and $\{f_i^1, ..., f_i^k, ..., f_i^K\}$. Neighbours only based on Euclidean distance may

bring in noisy points. So we affiliate with semantic information to revise the neighbouring points. Then the K neighbours and its center points construct a cluster to describe the local geometry. The elements of the cluster consist of four parts: point-wise distance-based neighbours p_i^k, neighbours rectified with semantic information h_i^k, the relative coordinates r_i^k and distances d_i^k. The cluster characteristics are fed into a shared-MLP to generate feature map $G = \{g_1, g_2, ..., g_K\}$ to represent the geometric information.

The graphical representation of our semantic-aware spatial block is shown in Fig. 2 blue part. 3D x-y-z coordinates are natural elements for point clouds. This block encodes local spatial structures with geometrical relationship from both distance and semantic approximable neighbours. We have demonstrated its effect in ablation study.

3.1.2 Attention block

Attention mechanism and transformer have been popular in point cloud analysis recently. Except for geometric features, semantic information is also essential for local representation. This attention block is designed to extract the most relevant neighboring semantic information for local feature learning. We investigate two questions in section 1 and this part is the solution to the second one.

Only taking feature difference as the criterion of attention weights will introduce redundancy. As a result, we consider both spatial distance and feature difference between center point and its neighbors to dynamically adjust the selection of neighborhoods. In paper [9], an attention mechanism has been introduced into the Graph-based method as follows:

$$\alpha(\Delta p_{ij}, \Delta h_{ij}) = M_s\{\Delta p_{ij}, \Delta h_{ij}\}, \tag{1}$$

where $\{.,.\}$ is the concatenation operation, and M_s is a mapping function to calculate the weights. Δp_{ij} and Δh_{ij} indicate the distance and feature difference between each vertex and its center point respectively.

Here, we transfer this idea to point-wise based methods. The difference is that this paper appends the distance factor to feature difference by calculating a distance-aware semantic deformation:

$$\Delta f_i^k = M_f\{(p_i - p_i^k), \parallel p_i - p_i^k \parallel, (p_i - h_i^k), \parallel p_i - h_i^k \parallel\}, \tag{2}$$

where Δf_i^k represents the semantic deformation, and M_f is a shared MLP.

Then the deformable neighboring feature can be denoted as $\nu_i^k = f_i^k + \Delta f_i^k$ and attention weight of each neighboring point is computed as follows:

$$\alpha_i^k = M_\alpha\{M_{g_1}(f_i, f_i^k), M_{g_2}(f_i, \nu_i^k)\}, \tag{3}$$

where $M_{g_i}(.,.)$ are mapping functions to assess the effects of neighbors, and M_α is the softmax function.

After the above operations, the feature of each neighboring point is recounted as follows:

$$\tilde{f}_i^k = \alpha_i^k \cdot (M_g\{f_i^k, \nu_i^k\}), \tag{4}$$

where M_g performs an MLP operation with a ReLU activation. The output of the semantic context encoding is the new set of high-level neighboring features, which softly selects the positive information by a set of adaptive attention weights controlled by modified feature difference.

The encoding procedures of spatial location and semantic context are not isolated in this Spatial-Semantic Cross-correction module. In this way, geometric and semantic information can be sufficiently exploited to acquire improved local feature representation.

3.2 Feature aggregation

Given the feature maps of geometric representation $G = \{g_1, ..., g_k, ..., g_K\}$ and semantic context $\tilde{F}_i = \{\tilde{f}_i^1, \tilde{f}_i^2, ..., \tilde{f}_i^k, ..., \tilde{f}_i^K\}$, we use a feature fusion strategy to concatenate them:

$$\Phi_i = \psi(\tilde{f}_i^k, r_i^k), \tag{5}$$

where ψ represents concatenate operation.

Once the aggregated features $\Phi_i = \{\varphi_i^1, ..., \varphi_i^k, ..., \varphi_i^K\}$ are obtained, we firstly perform a vector max operator to collect the most prominent neighbors,

$$\Phi_{max} = MAX(\Phi_i). \tag{6}$$

Considering max-pooling operation tends to save features in a hard way, we insistently borrow attention mechanism to obtain useful features abandoned by max-pooling. The attention weights can be calculated by a specific function $\Im_i(.,.)$ where the learned attention scores play the role of a soft mask to automatically focus on the important features, and these neighboring features are summed in the following way:

$$\Phi_{att} = \sum_{i=k}^{K} \Im_i(\varphi_i^k, W) \cdot \varphi_i^k, \tag{7}$$

where $\Im_i(,.,)$ consists of a shared MLP followed by softmax, and W is the learnable weights of the shared MLP.

Then the maximum and attentive features are combined,

$$\Phi_{com} = MLPs\{\Phi_{max}, \Phi_{att}\}, \tag{8}$$

Finally inspired by the idea of ResNet [25], skip connection is added to implement the cross-promotion features:

$$\tilde{\Phi}_i = \eta(F_i) + \Phi_{com}, \tag{9}$$

where η is the function of MLP.

3.3 Our Network Architecture

Fig. 3 shows the SSI-Net which follows the encoder-decoder structure to stack multiple-scale features. The input of this network is a large-scale point cloud with a dimension of $N \times d_{in}$ where N is the number of points, d_{in} is the feature dimension of each input point represented by its 3D coordinates and color information. The input is first fed into a fully connected layer to extract per-point features, and then several encoding layers and decoding layers are used to learn rich feature representation. Finally, three fully-connected layers and a dropout layer are appended to predict the semantic labels of the input.

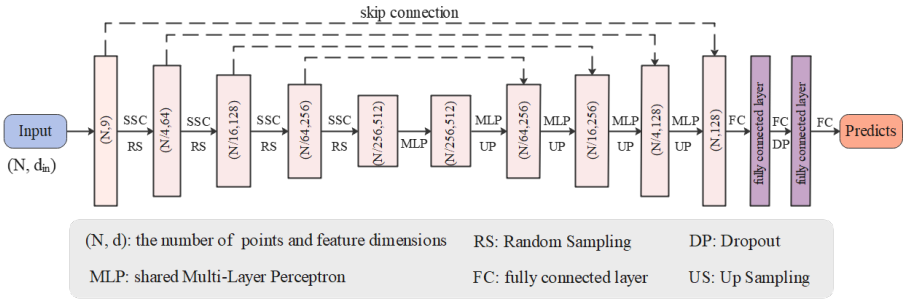

Fig. 3 The architecture of our proposed SSI-Net.

The structure settings refer to the work [10] as follows:

Encoding layer: Each encoding layer adopts a given sampling ratio ([4, 4, 4, 4, 2] for S3DIS, and [4, 4, 4, 4] for SemanticKITTI) to gradually reduce the point size, and the output dimensions of each layer are $(16, 64, 128, 256, 512)$ and $(16, 64, 128, 256)$ accordingly.

Decoding layer: The decoding layer used after the encoding layer is to restore the size of the input point cloud via a hierarchical propagation. Each decoding layer uses skip connection to help facilitate the feature extraction which concatenates the interpolated features with the features from the set abstraction layer to reduce information loss.

Semantic Prediction: Semantic segmentation generates one label for each point of the input point cloud. After restoring to the original size, three fully connected layers followed by one dropout layer with drop ratio 0.5 are joined to predict the final semantic labels.

4 Experiments

In this section, we demonstrate how our method can be trained to perform semantic segmentation on point cloud and divide experimental descriptions into three parts. First, some necessary settings about our experiments are provided for comparison with the state-of-the-art. Second, detailed quantitative

and qualitative results on different datasets are shown to illustrate the performance. Finally, ablation studies are performed to explain the selection of our network design.

4.1 Experimental Settings

Evaluation Metrics: The mean Intersection-over-Union (**mIoU**), the average value of **IoUs** for all semantic classes, the overall accuracy (**OA**), and the average class accuracy (**mAcc**) are common standard scores to evaluate the semantic segmentation performance.

Datasets: This work targets an accurate semantic segmentation on large-scale point cloud scenes. To validate the proposed SSI-Net, we conduct experiments on some indoor and outdoor datasets.

1. Evaluation on S3DIS: Stanford Large-Scale 3D Indoor Spaces (S3DIS) dataset is derived from real 3D scans and extensively used by lots of jobs. S3DIS dataset includes 271 rooms from 6 areas containing 13 classes of objects typically encountered in an indoor scene: ceiling, floor, wall, beam, column, window, door, table, chair, sofa, bookcase, board, and clutter. Points in this dataset provide both 3D coordinates and color information. In the experiment, the number of input points on this dataset is set as $4096*10$. To evaluate the semantic segmentation results on S3DIS, we respectively provide evaluation on Area 5 and 6-fold cross-validation results to compare the performances with certain state-of-the-are networks. The **mACC**, **OA**, and **mIoU** of the overall classes are compared in this paper.

2. Evaluation on SemanticKITTI: SemanticKITTI is a large-scale outdoor scene dataset, which based on the KITTI odometry dataset showing inner city traffic, residential areas, but also highway scenes and countryside roads. There are 22 sequences ($00 \sim 10$ as training set, and $11 \sim 21$ as test set) which are annotated in 19 semantic classes: road, sidewalk, parking, other-ground, building, car, truck, bicycle, motorcycle, other-vehicle, vegetation, trunk, terrain, person, bicyclist, motorcyclist, fence, pole, and traffic-sign. The raw point cloud contains 3D coordinates information. The number of input points is set as $4096 * 11$ in experiment. For this dataset, **mIoU** and **IoU** of each class are taken as the evaluation metrics.

Training Settings: Our experiments have been performed with Python 3.6, Tensorflow 1.12 GPU version and trained 100 epochs on a single GeForce RTX 2080Ti GPU. During training, the batch size is set as 4, and the Adam optimizer is used. Besides, the initial learning rate is 0.01 and decays with a rate of 0.5 after every 10 epochs.

4.2 Performance Comparison

4.2.1 Results of S3DIS

Table 1 and Table 2 show the results on the S3DIS compared with different methods under the two evaluation modes mentioned in section 4.1. Table 1

presents the results tested on Area 5. Our SSI-Net achieves the best performance in terms of **mACC** (73.2%) and **mIoU** (65.1%) compared to these methods that supply evaluation on Area 5. The **mIoU** has improved by 3.7% relative to the latest BoundaryAwareGEM [31], and the mACC has increased by 6.2% over PointWeb [20]. The **OA** value of SSI-Net is 0.1% lower than ELGS[24] which builds a more complex structure with graph attention block, the spatial-wise and channel-wise attention designed for small-scale point cloud processing. Moreover, most of the compared methods in the table prefer the farthest point sampling (FPS) as it leads to less information loss in comparison to random sampling.

Table 1 Results (%) on S3DIS evaluated on Area 5

Methods	OA	mACC	mIoU
PointNet[2]	-	49.0	41.1
SegCloud[14]	-	57.4	48.9
PointCNN[26]	85.9	63.9	57.3
SPGraph[27]	86.4	66.5	58.
PCCN [28]	-	67.0	58.3
PointWeb[20]	86.9	66.6	60.3
ELGS[20]	**88.4**	-	60.1
BoundaryAwareGEM[31]	-	-	61.4
SSI-Net	88.3	**73.2**	**65.1**

Table 2 gives the results on the 6-fold cross-validation compared with PointNet[2], RSNet[32], 3P-RNN[33], PointCNN[26], ShellNet[20], PointWeb[20], KPConv$_{rigid}$[29], KPConv$_{deform}$[29], PointASNL[30], and RandLA[10]. Approaches [2, 10, 20, 30, 32] are classified into point-based methods. Other researches such as [26, 29] utilize convolution-like operation to improve feature representation. The **mACC** of SSI-Net rises to 82.3% and surpasses listed results in this test mode. However, the scores of **OA** and **mIoU** are slightly inferior to PointASNL[30] and KPConv$_{deform}$[29] respectively. Most of the methods return similarly high IoU values for simple classes. Significant differences distribute in classes with complex structures such as table, chair and sofa. Besides, detailed semantic segmentation results for each area on S3DIS dataset with 6-fold cross-validation are shown in Table 3 to better represent the performance of our approach.

4.2.2 Results of SemanticKITTI

SemanticKITTI is a challenging dataset. Table 4 reports the quantitative results of SSI-Net on this dataset compared with some representative methods. **mIoU** of our method achieves best value 55.4% which surpasses most point-based methods [2, 18, 30, 35, 36, 39] by a large margin. From Table 4, one can see that the projection-based methods[37, 38, 40, 41] are superior

Table 2 Results (%) on S3DIS dataset with 6-fold cross-validation

Methods	OA	mACC	mIoU	ceil.	floor	wall	beam	col.	wind.	door	table	chair	sofa	book.	board	clut.
PointNet[2]	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[32]	—	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[33]	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
PointCNN[26]	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
ShellNet[34]	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
PointWeb[20]	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
KPConv$_{rigid}$[29]	—	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$_{deform}$[29]	—	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
PointASNL[30]	**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**
RandLA[10]	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
SSI-Net	88.0	**82.3**	70.5	93.7	96.8	80.1	61.9	44.0	65.0	69.7	**72.8**	74.6	67.6	63.2	**66.0**	60.6

Table 3 Detailed results (%) for each area on S3DIS dataset with 6-fold cross-validation

Testing Area	OA	mIoU	mACC	ceil.	floor	wall	beam	col.	wind.	door	table	chair	sofa	book.	board	clut.
Area1	89.2	75.7	87.6	96.3	95.1	77.1	54.3	51.9	80.1	83.4	73.3	81.4	76.5	62.8	70.4	67.7
Area2	84.2	55.4	70.8	89.0	95.5	76.8	21.4	26.6	52.2	64.6	49.8	60.3	56.0	50.0	28.3	49.8
Area3	91.1	79.2	89.4	95.7	98.2	81.4	70.2	33.3	82.1	88.5	74.7	84.8	85.0	74.5	88.6	73.0
Area4	85.1	62.1	76.6	94.1	97.0	77.8	39.9	48.8	31.8	60.5	68.6	77.7	65.5	46.1	39.7	60.0
Area5	88.3	65.1	73.2	93.1	97.7	81.7	0.0	24.5	61.9	54.2	79.4	87.7	67.0	70.4	72.0	56.0
Area6	91.9	80.0	92.2	96.7	97.5	84.4	82.0	72.0	80.9	86.4	75.9	84.2	62.1	72.4	75.1	70.8

to the point-based methods on the whole. However, our SSI-Net also outperforms these mentioned projection-based methods. For example, **mIoU** value increases by 1.1% compared to the best projection-based method[41]. Besides, it obtains the maximum mIoU in 11 out of the 19 categories particularly the values of truck and motorcycle which are much higher than the second ones. We attribute this to the active cross-correction of the spatial and semantic information developed by our ssc module.

Table 4 Semantic segmentation results (%) on the SemanticKITTI dataset

Methods	mIoU	road	sidewalk	parking	other-ground	building	car	truck	bicycle	motorcycle	other-vehicle	vegetation	trunk	terrain	person	bicyclist	motorcyclist	fence	pole	traffic-sign
PointNet[2]	14.6	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[35]	17.4	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[36]	18.4	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++[18]	20.1	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
SquSeg[37]	29.5	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	29.9	17.5	24.5
SquSegV2[38]	39.7	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
TangentConv[39]	40.9	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
DarkNet21Seg[40]	47.4	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[40]	49.9	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
PointASNL[30]	46.8	87.4	74.3	24.3	1.8	83.1	87.9	39.0	0.0	25.1	29.2	**84.1**	52.2	70.6	34.2	57.6	0.0	43.9	57.8	36.9
RandLA-Net[10]	53.9	90.7	73.7	60.3	20.4	86.9	**94.2**	40.1	26.0	25.8	38.9	81.4	61.3	66.8	49.2	48.2	7.2	56.3	49.2	47.7
PolarNet[41]	54.3	90.8	74.4	61.7	21.7	**90.0**	93.8	22.9	**40.3**	30.1	28.5	84.0	**65.5**	67.8	43.2	40.2	5.6	67.8	51.8	**57.5**
SSC-Net	**55.4**	**92.2**	12.9	36.9	**72.0**	27.8	55.0	**66.4**	0.0	**92.9**	**42.2**	79.6	0.8	**89.5**	**54.2**	**86.8**	**66.0**	**76.1**	**58.6**	41.9

Fig. 3 and Fig. 4 present concerned illustrations of SemanticKITTI dataset. Fig. 3 shows some qualitative results on the validation set. SemanticKITTI provides an unprecedented number of scans covering the full 360 degree field-of-view of the employed automotive LiDAR and we choose four scans from

sequence 08 to reveal a contrast of segmentation results. Pictures in the first row are the ground truths, and these in the second row are the outputs of SSI-Net.

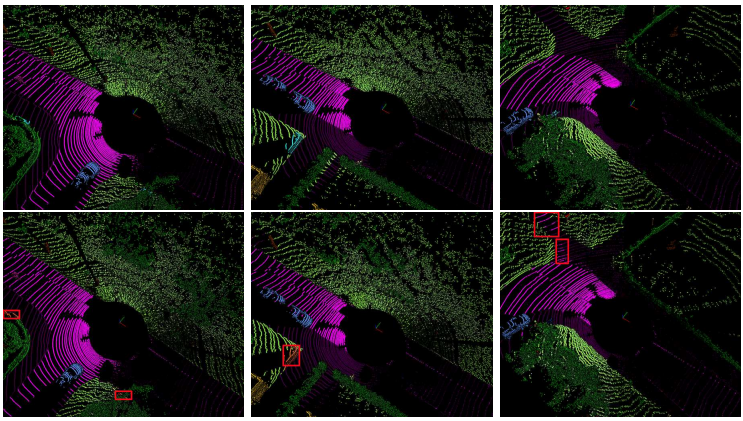

Fig. 4 Qualitative results of our SSI-Net on the validation set of SemanticKITTI. Red rectangles represent the failure cases.

Sequences $11 \sim 21$ are used as a test set showing a large variety of challenging traffic situations and environment types. Fig. 4 exhibits some qualitative results of online test (sequences $11 \sim 21$) in 2D panorama views.

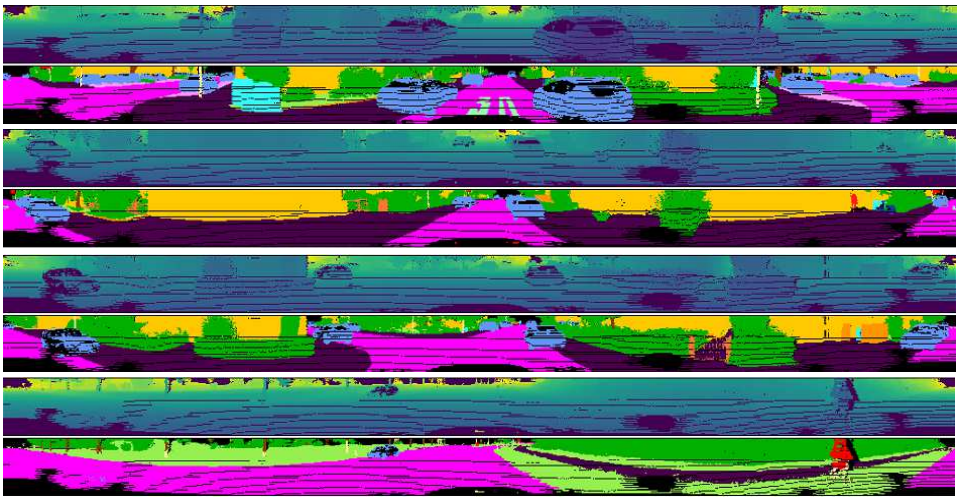

Fig. 5 Qualitative results of online test on sequence $11 \sim 21$.

4.3 Ablation Studies

Effect of each unit: To demonstrate the effect of each component: semantic-aware spatial location encoding, attentional semantic encoding, and feature aggregation block, we perform the following ablation studies on Area 5, S3DIS dataset:

(1) Removing the semantic-aware information of spatial location encoding: this part aims to encode more detailed local geometry with position and high-level information;

(2) Replacing the attentional semantic encoding by general MLP layers;

(3) Aggregating local features only by max operation.

Table 5 Ablation studies on Area 5, S3DIS (%)

Methods	mIoU
Removing semantic-aware information	63.6
Removing attentional semantic block	61.9
Max operation	63.1
With full units	65.1

Results of this part are shown in Table 5: 1) The encoder with our full units reaches the best mIoU; 2) The greatest impact on mIoU is caused by the removing of attentional semantic block probably because attention mechanism and neighboring deformation can help aggregate key semantic information discarded by random sampling.

Selection of spatial encoding block: As described in section 3.1.1, the semantic-aware spatial block can be expressed as follows:

$$g_i^k = M_s\{p_i^k, h_i^k, (p_i - p_i^k), \| p_i - p_i^k \|, (p_i - h_i^k), \| p_i - h_i^k \|\}. \tag{10}$$

We perform other experiments for the selection of this module:

(1) Encoding the neighboring points p_i^k and deformable neighboring points h_i^k;

(2) Encoding the relative position: $p_i - p_i^k$ and $p_i - h_i^k$, and corresponding Euclidean distance: $\| p_i - p_i^k \|$ and $\| p_i - h_i^k \|$;

(3) Encoding the point p_i, the relative position: $p_i - p_i^k$ and $p_i - h_i^k$, and Euclidean distance: $\| p_i - p_i^k \|$ and $\| p_i - h_i^k \|$;

(4) Encoding the neighbouring points: p_i^k and h_i^k, the relative position: $p_i - p_i^k$ and $p_i - h_i^k$, and the Euclidean distance: $\| p_i - p_i^k \|$ and $\| p_i - h_i^k \|$.

Table 6 compares the mIoU values with different selections: 1) Encoding the neighboring points, the relative position and the Euclidean distance outputs the highest mIoU; 2) The neighboring points play an important role in our spatial location encoding block.

Table 6 Selection for spatial location encoding test on Area 5, S3DIS (%)

Methods	mIoU
(1) $\{p_i^k, h_i^k\}$	63.5
(2) $\{(p_i - p_i^k), \parallel p_i - p_i^k \parallel, (p_i - h_i^k), \parallel p_i - h_i^k \parallel\}$	65.0
(3) $\{p_i, (p_i - p_i^k), \parallel p_i - p_i^k \parallel, (p_i - h_i^k), \parallel p_i - h_i^k \parallel\}$	63.4
(4) $\{p_i^k, h_i^k, (p_i - p_i^k), \parallel p_i - p_i^k \parallel, (p_i - h_i^k), \parallel p_i - h_i^k \parallel\}$	65.1

FPS vs. RS: We compare the **mIoU** and **Iou** value of each class in this part. The **mIou** value of RS is only lower than FPS by 0.1%; however, as we know the FPS is much less efficient than RS. Thus, we choose random sampling in our design.

Table 7 Effect of different sampling strategies (%)

Samplings	mIoU	ceil.	floor	wall	beam	col.	wind.	door	table	chair	sofa	book.	board	clut.
Farthest Point sampling (FPS)	65.2	93.9	97.5	82.0	0.0	30.2	62.2	48.9	80.4	87.4	62.5	72.4	72.5	57.4
Random Sampling (RS)	65.1	93.1	97.7	81.7	0.0	24.5	61.9	54.2	79.4	87.7	67.0	70.4	72.0	56.0

5 Conclusion

This paper pays attention to semantic segmentation on point clouds for large-scale scenes. Specifically, our proposed architecture SSI-Net focuses on a more effective feature description via a spatial and semantic cross correction manner with an SSC module. By mutually revising the neighboring information, robust representation can be obtained to support our work. Extensive experiments on relevant benchmarks have proven the state-of-the-art performance of this work. Applying the proposed method to practical applications of point cloud processing will be an interesting job in the future.

Acknowledgments

All authors thank reviews and editors for their valuable comments and suggestions. Besides, the authors acknowledge the support of Hebei Natural Science Foundation, Hebei Postgraduate Innovation Funding Project, and the Scientific Research Project of Tianjin Municipal Commission of Education.

Declarations

Conflict of interest: All authors declare that there is no professional or other personal interest of any nature or kind in any product, service and/or company.
Ethical approval: This work has not been submitted and published elsewhere in any form or language. The results presented in this article are clear, honest, and without any fabrication.

Funding details: This work is supported by Hebei Natural Science Foundation (Grant number [F2020202045]), Hebei Postgraduate Innovation Funding Project (Grant number [CXZZBS2020026]), and the Scientific Research Project of Tianjin Municipal Commission of Education (Grant number [2018KJ268]).

Informed Consent: All authors agree with the content and approve the final submitted work.

Author contributions: The network design and code were performed by Zhao Yuehua. The first draft of the manuscript was written by Zhao Yuehua, and commented by all authors especially Ma Jie. All authors read and approved the final version to be published.

References

[1] Sun W, Zhang Z, Huang J. RobNet: real-time road-object 3D point cloud segmentation based on SqueezeNet and cyclic CRF[J]. Soft Computing, 2020, 24(8): 5805-5818.

[2] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 652–660, 2017.

[3] Lan S, Yu R, Yu G, et al. Modeling local geometric structure of 3d point clouds using geo-cnn. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[4] Wu W, Qi Z, Fuxin L. Pointconv: Deep convolutional networks on 3d point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[5] He T, Shen C, Hengel A. Dynamic Convolution for 3D Point Cloud Instance Segmentation[J]. arXiv preprint arXiv:2107.08392, 2021.

[6] 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 Transactions on Graphics (TOG), 2019.

[7] Li G, Muller M, Thabet A, et al. DeepGCNs: Can gcns go as deep as cnns? In Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019.

[8] Zhang Y, Rabbat M. A graph-cnn for 3d point cloud classification. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018.

[9] Wang L, Huang Y, Hou Y, et al. Graph attention convolution for point cloud semantic segmentation. In CVPR, 2019.

[10] Hu Q, Yang B, Xie L, et al. Learning Semantic Segmentation of Large-Scale Point Clouds with Random Sampling[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021.

[11] Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E. Multi-view convolutional neural networks for 3d shape recognition. In Proceedings of the IEEE international conference on computer vision, 2015.

[12] Boulch, A., Guerry, J., Le Saux, B., Audebert, N., 2018. Snapnet: 3d point cloud semantic labeling with 2d deep segmentation networks. Computers Graphics 71, 189–198.

[13] Maturana, D., Scherer, S., 2015. Voxnet: A 3d convolutional neural network for real-time object recognition. In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE. pp. 922–928.

[14] Tchapmi, L., Choy, C., Armeni, I., Gwak, J., Savarese, S. Segcloud: Semantic segmentation of 3d point clouds. In 2017 international conference on 3D vision (3DV), IEEE. pp. 537–547.

[15] Wang P S, Liu Y, Guo Y X, et al. O-cnn: Octree-based convolutional neural networks for 3d shape analysis[J]. ACM Transactions On Graphics (TOG), 2017, 36(4): 1-11.

[16] Meng H Y, Gao L, Lai Y K, et al. Vv-net: Voxel vae net with group convolutions for point cloud segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019.

[17] Riegler G, Osman Ulusoy A, Geiger A. Octnet: Learning deep 3d representations at high resolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, 2017.

[18] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In NeurIPS, 2017.

[19] Zhang D, He F, Tu Z, et al. Pointwise geometric and semantic learning network on 3D point clouds[J]. Integrated Computer-Aided Engineering, 2020, 27(1): 57-75.

[20] Zhao H, Jiang L, Fu C W, et al. Pointweb: Enhancing local neighborhood features for point cloud processing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[21] Han W, Wen C, Wang C, et al. Point2Node: Correlation learning of dynamic-node for point cloud feature modeling[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(07): 10925-10932.

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

[23] Siheng Chen, Sufeng Niu, Tian Lan, and Baoan Liu. PCT: Large-scale 3D point cloud representations via graph inception networks with applications to autonomous driving. In ICIP, 2019.

[24] Wang, X., He, J., Ma, L., 2019b. Exploiting local and global structure for point cloud semantic segmentation with contextual point representations, in: Advances in Neural Information Processing Systems, pp. 4571–4581.

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

[26] Shenlong Wang, Simon Suo, Wei-Chiu Ma, Andrei Pokrovsky, and Raquel Urtasun. Deep parametric continuous convolutional neural networks. In CVPR, 2018.

[27] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In CVPR, 2018.

[28] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on X-transformed points. In NIPS, 2018.

[29] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In Proceedings of the IEEE International Conference on Computer Vision, 2019.

[30] Yan X, Zheng C, Li Z, et al. PointASNL: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In CVPR, 2020.

[31] Gong J, Xu J, Tan X, et al. Boundary-aware geometric encoding for semantic segmentation of point clouds[J]. arXiv preprint arXiv:2101.02381, 2021.

[32] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3d segmentation of point clouds. In CVPR, 2018.

[33] 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.

[34] Zhang Z, Hua B S, Yeung S K. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. In CVPR, 2019.

[35] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In CVPR, 2018.

[36] 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, 2018.

[37] 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, 2018.

[38] 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 International Conference on Robotics and Automation (ICRA), 2019.

[39] Maxim Tatarchenko, Jaesik Park, Vladlen Koltun, and QianYi Zhou. Tangent convolutions for dense prediction in 3D. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[40] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Juergen Gall. SemanticKITTI: A dataset for semantic scene understanding of lidar sequences. In ICCV, 2019.

[41] Yang Zhang, Zixiang Zhou, Philip David, Xiangyu Yue, Zerong Xi, Boqing Gong, and Hassan Foroosh. Polarnet: An 10 improved grid representation for online lidar point clouds semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.