remote sensing

MDPI

Article

Large-Scale Semantic Scene Understanding with Cross-Correction Representation

Yuehua Zhao [1], Jiguang Zhang [2], Jie Ma [1,*] and Shibiao Xu [3,*]

[1] School of Electronics and Information Engineering, Hebei University of Technology, Tianjin 300401, China
[2] Institute of Automation of Chinese Academy of Sciences, Beijing 100090, China
[3] School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China
* Correspondence: jma@hebut.edu.cn (J.M.); shibiaoxu@bupt.edu.cn (S.X.)

Abstract: Real-time large-scale point cloud segmentation is an important but challenging task for practical applications such as remote sensing and robotics. Existing real-time methods have achieved acceptable performance by aggregating local information. However, most of them only exploit local spatial geometric or 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 correcting and updating high-level semantic information using spatial geometric cues and vice versa. 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 indoor and outdoor datasets for point cloud semantic segmentation demonstrate that the proposed approach can achieve state-of-the-art performance.

Keywords: point cloud; large-scale semantic segmentation; spatial geometric; semantic context; cross-correction

Citation: Zhao, Y.; Zhang, J.; Ma, J.; Xu, S. Large-Scale Semantic Scene Understanding with Cross-Correction Representation. *Remote Sens.* **2022**, *14*, 6022. https://doi.org/10.3390/rs14236022

Academic Editor: Benoit Vozel

Received: 8 October 2022
Accepted: 23 November 2022
Published: 28 November 2022

1. Introduction

Recently, high-quality 3D scanners and depth sensors are available to many agents such as 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 the segmentation result plays a basic and critical role in providing self-driving cars or robots with scene-level understandings such as urban remote sensing information from a LiDAR point cloud.

In recent years, researchers [1–8] have shown great success in terms of semantic segmentation using deep learning models. Early works [3–7] focus on researching how to segment small-scale point clouds such as object surfaces sampled from CAD models. Although having 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. Loic Landrieu et al. [9] presenting a efficient structure called Superpoint graph (SPG) is a pioneering work that tailored large-scale point cloud segmentation. However, it would cause huge computational cost, making it impossible to achieve real-time performance. Therefore, RandLANet [10] is proposed. This method suggests randomly downsampling 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 predict wrong segmentation results at some ambiguous regions or lose local details, as shown in Figure 1.

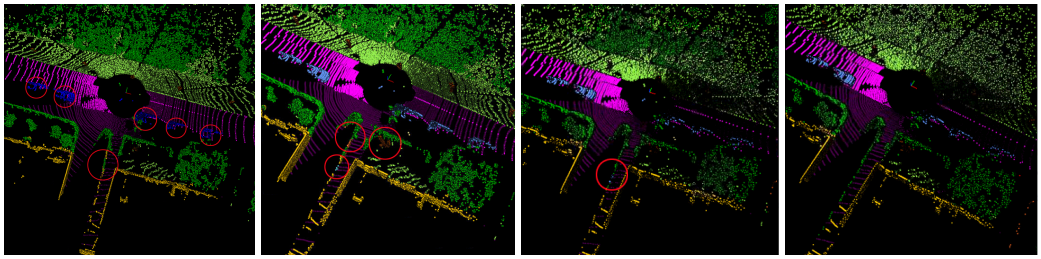

Figure 1. Semantic predictions of LiDAR scans. From left to right are the results of SPG [9], RandLA [10], SSI-Net, and ground truth.

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 a robust non-local features for point cloud semantic segmentation via subtly relating spatial geometric and high-level semantic information. To better incorporate these two kinds of information, we inspect two questions: (1) how to augment geometric patterns with the guidance of semantic information, and (2) how can spatial geometric be used to aggregate most positive semantic information in turn. 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 correcting and updating semantic features using spatial cues and vice verse. More specifically, to learn discriminative non-local features for semantic segmentation, both branches first perform the nearest neighboring algorithm locally to find K candidate neighbors and form fundamental clusters where each cluster is constructed with the K neighbors, center points and the differences between the neighbors and the corresponding center point. Then, a constraint condition is generated in one branch to append on the other to refine its representation so that the spatial geometric and high-level semantics can be well acquired. Finally, the outputs of the two branches are updated by a feature aggregation operation to obtain the cross-promoted features. 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 that can be appropriate for both indoor and outdoor point cloud semantic segmentation to implement real-time scene perception guidance. SSI-Net highly aggregates high-level semantic information and spatial geometric patterns to enhance the descriptor's representation ability. Thus, robust non-local features of indoor and outdoor scenes can be obtained to improve the precision of semantic segmentation on point clouds.
- We propose the Spatial–Semantic Cross-correction (SSC) module, which can delicately interconnect high-level semantic and spatial geometric features in the latent space through two intersecting point cloud nearest neighbor clustering branches. Specifically, the constraints of high-level semantic information reduce the error rate of geometric expression, and conversely, spatial features can expand the scope of high-level semantic information. As a consequence, the mutual promotion and fusion provide more sufficient context information for point cloud semantic segmentation.
- Being computationally efficient, SSI-Net meets the needs of large-scale scenes. Results on several indoor and outdoor public datasets for point cloud segmentation demonstrate the state-of-the-art power of our proposed method in terms of Intersection-over-Union and overall accuracy for large-scale processing.

2. Related Work

This part makes a simple list of the point cloud analysis development mainly based on deep learning methods. We emphasize some work that is related to large-scale point cloud clouds.

2.1. Deep Learning on Point Cloud Segmentation

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 clouds.

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 rendered views on 2D images to allow the direct application of 2D CNNs. SnapNet [11] is one example that uses a multiview-based method to deal with 3D semantic segmentation. Approaches in this category can deal 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). When it comes to large-scale tasks, covering 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 the point cloud into voxel grid and applying 3D CNNs directly. One well-known voxel-based deep learning architecture for semantic segmentation is SegCloud [12], which utilizes a preprocessing step to voxelize point clouds to adapt the 3D fully convolutional neural network. Other works such as [13–15] also propose some typical networks for semantic segmentation. Unfortunately, this kind of approach will introduce information loss, and the storage scheme lacks efficiency in terms of computation and memory usage. PCSCNet [16] tries to fix this problem and suggests a fast voxel-based semantic segmentation model using Point Convolution and 3D Sparse Convolution, which outperforms at both high and low resolution and accelerates the feature propagation.

Pointwise MLP methods: As the first job to process point clouds without any formal transformation, PointNet [17] takes advantage of some symmetric operations, i.e., max-pooling and Multi-Layer Perceptrons (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 the 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 a local neighboring connection for better representation. Inspired by the non-local operation, PointASNL [22] proposes a local–non-local module to further capture the neighbor and long-range dependencies of the sampled point. Ref. [23] selectively performs the neighborhood feature aggregation with dynamic pooling and an attention mechanism. Although a large amount of approaches have been proposed, the idea of PointNet [17] remains the standard. Moreover, ref. [24] employs a two-layer MLP to realize a binary segmentation module and reach comparable detection with the help of semantics.

Graph-based methods: The advances and difficulties of current research inspire the combination of a graph concept and point cloud analysis. Approaches of this category 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, to realize deep graph convolutional networks (GCNs), Li et al. [7] utilize residual/dense connections and dilated convolutions, which breaks the bottleneck that GCNs are limited to very shallow models due to the vanishing gradient problem. Ref. [25] improves point representations and local neighborhood graph construction within the general framework of graph neural networks by a 9D local geometric representation and a locality adaptive graph construction algorithm.

2.2. Large-Scale Point Cloud Semantic Segmentation

A large proportion of the above approaches aims at partial segmentation or small scene segmentation. In recent years, more and more scene perception tasks, for example autonomous driving and remote sensing, require large-scale processing techniques. To meet the demand, some approaches [9,10,26,27] have explored large-scale point cloud analysis. SPG [9] uses a 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 [26,27] 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. MVP-Net [28] proposes an end-to-end network to realize a novel neighbor searching for pointwise point cloud semantic segmentation. With current development in sensors and tasks, there are increasingly high-precision requirements for semantic prediction. Nevertheless, current research is far from enough. To explore high-quality contextual features for semantic segmentation and use these results to represent scenes more effectively, this work puts effort into these problems and proposes a cross-promoted method.

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. 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 improve the feature representation not only with the spatial geometric but also the high-level semantic information.

3.1. Spatial-Semantic Cross-Correction Module

Figure 2 shows the structure of our SSC module which can be decomposed into two parts: a semantic-aware spatial block to encode spatial geometric features and an attention block to extract high-level semantic information. Each block first aggregates nearest neighbors to acquire preliminary spatial geometric and semantics. The neighboring information clustered based on single criterion can introduce outliers or redundancies to some extent. Here, we associate the two branches together. The spatial geometric is rectified with position offsets inferred from semantics and vice versa. With such an in-depth feature learning style, the representation ability of the descriptor will be improved, and robust non-local features can finally be extracted to realize high-precision semantic segmentation.

3.1.1. Semantic-Aware Spatial Block

Given a point cloud with n points, their coordinates and acquired features can be denoted as $P = \{p_1, \ldots, p_i, \ldots, p_n\} \subset R^3$ and $F = \{f_1, \ldots, f_i, \ldots, f_n\} \subset R^N$, respectively. As shown in Figure 2, we firstly perform k-nearest neighboring (kNN) to search K candidate neighborhoods denoted as $\{p_i^1, \ldots, p_i^k, \ldots, p_i^K\}$ and $\{f_i^1, \ldots, f_i^k, \ldots, f_i^K\}$. Neighbors only based on Euclidean distance may bring in noisy points, so we affiliate with semantic information to revise the neighboring points. Then, the K neighbors 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 neighbors p_i^k, neighbors 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 $G = \{g_1, g_2, \ldots, g_K\}$ to represent the geometric information.

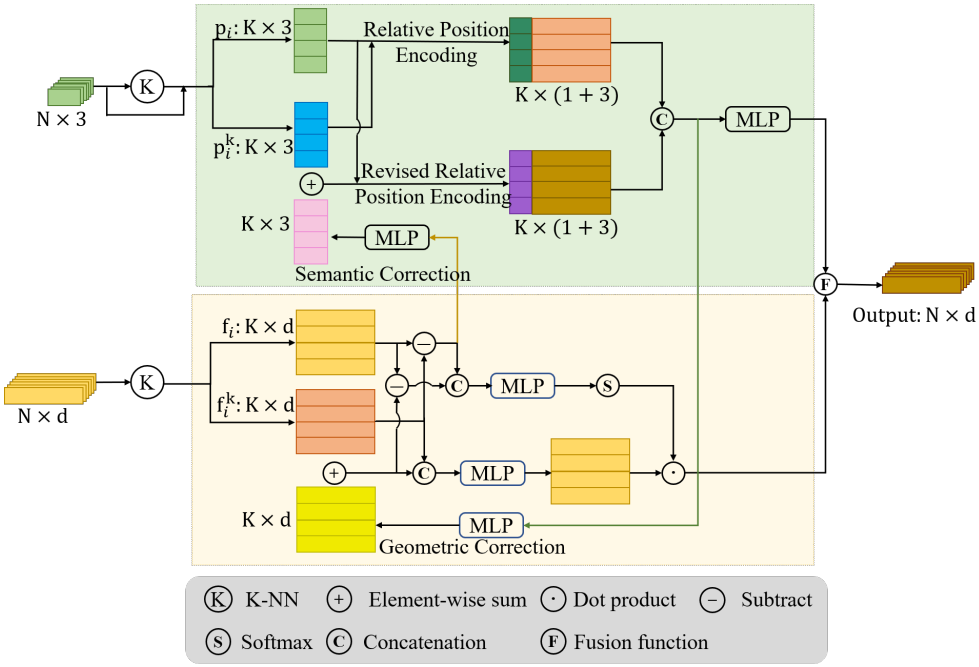

Figure 2. Structure of the proposed Spatial–Semantic Cross-correction module.

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

3.1.2. Attention Block

We have investigated two questions in Section 1, and this part is the solution to the second one. Our modified attention block is designed to better refine semantic information for local feature learning. The attention mechanism can help update point representation by assigning weights to different neighbors. To capture local representation, feature difference is taken as a constraint to select key information. However, only considering the feature difference as the criterion will introduce redundancy, and the spatial geometric relationship should play a role in correcting neighboring features.

Here, we combine this idea with point-wise based methods. Different from the previous feature difference, this paper appends the distance factor by calculating a distance-aware semantic deformation to refine the weight constraint:

$$\Delta f_i^k = M_f\{(p_i - p_i^k), \parallel p_i - p_i^k \parallel, (p_i - h_i^k), \parallel p_i - h_i^k \parallel\}, \tag{1}$$

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

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

$$\alpha_i^k = M_\alpha\{M_{g_1}(f_i, f_i^k), M_{g_2}(f_i, v_i^k)\}, \tag{2}$$

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:

$$\tilde{f}_i^k = \alpha_i^k \cdot (M_g\{f_i^k, v_i^k\}), \tag{3}$$

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 the 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

The SSC module explores correlation to represent scenes more effectively with augmented spatial geometric patterns and high-level semantic information. Given the feature maps of geometric representation $G = \{g_1, \ldots, g_k, \ldots, g_K\}$ and semantic context $\tilde{F}_i = \{\tilde{f}_i^1, \tilde{f}_i^2, \ldots, \tilde{f}_i^k, \ldots, \tilde{f}_i^K\}$ generated by the SSC module in Section 3.1, we use a feature fusion strategy to aggregate them:

$$\Phi_i = \psi(\tilde{f}_i^k, r_i^k), \tag{4}$$

where ψ represents concatenate operation.

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

$$\Phi_{max} = MAX(\Phi_i). \tag{5}$$

Considering max-pooling operation tends to save features in a hard way, we insistently borrow an 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:

$$\Phi_{att} = \sum_{i=k}^{K} \Im_i(\varphi_i^k, W) \cdot \varphi_i^k, \tag{6}$$

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,

$$\Phi_{com} = MLPs\{\Phi_{max}, \Phi_{att}\}, \tag{7}$$

Finally inspired by the idea of ResNet [29], skip connection is added to implement the cross-promotion features:

$$\tilde{\Phi}_i = \eta(F_i) + \Phi_{com}, \tag{8}$$

where η is the function consisting of a shared MLP to improve F_i and guarantee the same dimension with Φ_{com}.

3.3. Our Network Architecture

This paper concentrates on improving the representation ability of the descriptor. We adopt the described module to construct the SSI-Net, shown in Figure 3, which follows the encoder–decoder structure to acquire 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, and 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.

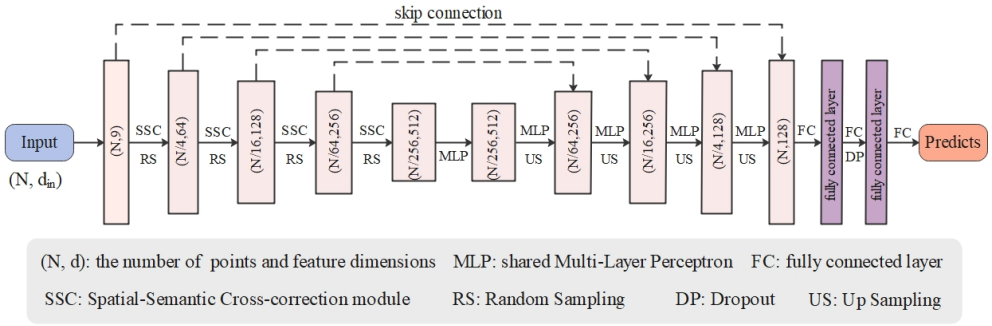

Figure 3. Structure of the proposed Spatial–Semantic Cross-correction module.

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 a drop ratio of 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 clouds 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 Intersection over Union (**IoU**) for each semantic class, 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: The Stanford Large-Scale 3D Indoor Spaces (S3DIS) dataset is derived from real 3D scans and extensively used by lots of jobs. The S3DIS dataset includes 271 rooms from six 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 provide evaluation on Area 5 and 6-fold cross-validation results to compare the performances with certain state-of-the-art 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 is based on the KITTI odometry dataset showing inner city traffic and residential areas but also highway scenes and countryside roads. There are 22 sequences ($00 \sim 10$ as the training set, and $11 \sim 21$ as the 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 the experiment. For this dataset, the **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 for 100 epochs. During training, the batch size is set as 4, and the Adam optimizer is used. 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

This part shows 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 [30], and the **mACC** has increased by 6.2% over PointWeb [20]. The **OA** value of SSI-Net is 0.1% lower than ELGS [31] which builds a more complex structure with graph attention block, the spatial-wise and channel-wise attention and is 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. For better comparison, Table 2 shows the IoU value of each class of S3DIS on Area 5 from which one can see that our SSI-Net achieves the best performance on some complex structures such as a table, sofa, and bookcase.

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

Methods	OA	mACC	mIoU
PointNet [17]	-	49.0	41.1
SegCloud [12]	-	57.4	48.9
PointCNN [32]	85.9	63.9	57.3
SPG [9]	86.4	66.5	58.0
PCCN [33]	-	67.0	58.3
PointWeb [20]	86.9	66.6	60.3
ELGS [31]	88.4	-	60.1
MinkowskiNet20 [34]	-	62.6	69.6
MinkowskiNet32 [34]	-	65.4	71.7
BoundaryAwareGEM [30]	-	-	61.4
DPFA [23]	87.4	-	53.0
DPFA+BF Reg [23]	88.0	-	55.2
3D GrabCut [35]	-	-	57.7
Box2Seg (AST) [35]	-	-	60.4
SSI-Net	88.3	73.2	65.1

Table 3 gives the results on the 6-fold cross-validation compared with PointNet [17], RSNet [36], 3P-RNN [37], PointCNN [32], ShellNet [38], PointWeb [20], KPConv$_{rigid}$ [39], KPConv$_{deform}$[39], PointASNL [22], and RandLA [10]. Approaches [10,17,20,22,38] are classified into point-based methods. Other researchers such as [32,39] utilize convolution-like operation to improve feature representation. The **mACC** of SSI-Net rises to 82.3% and surpasses the listed results in this test mode. However, the scores of **OA** and **mIoU** are slightly inferior to PointASNL [22] and KPConv$_{deform}$ [39], 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.

Table 2. IoU (%) of each class of S3DIS evaluated on Area 5.

Methods	Ceil.	Floor	Wall	Beam	Col.	Wind.	Door	Table	Chair	Sofa	Book.	Board	Clut.
PointNet [17]	88.8	97.3	69.8	0.05	3.9	46.3	10.8	52.6	58.9	40.3	5.9	26.4	33.2
SegCloud [12]	90.1	96.1	69.9	0.0	18.4	38.4	23.1	75.9	70.4	58.4	40.9	13.0	41.6
PointCNN [32]	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
SPG [9]	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
PCCN [32]	92.3	96.2	75.9	0.3	5.98	69.5	63.5	66.9	65.6	47.3	68.9	59.1	46.2
PointWeb [20]	92.0	98.5	79.4	0.0	21.1	59.9	34.8	76.3	88.3	46.9	69.3	64.9	52.5
3D GrabCut [35]	80.3	88.7	69.6	0.0	28.8	61.0	35.2	66.5	71.7	69.2	69.7	61.7	48.2
Box2Seg (AST) [35]	82.0	92.0	70.8	0.0	28.8	61.9	38.1	71.4	85.3	74.3	68.4	63.6	48.0
DPFA [23]	93.7	98.7	75.5	0.0	14.5	50.1	31.8	73.7	73.4	13.7	55.5	57.1	51.2
DPFA+BF Reg [23]	93.0	98.6	80.2	0.0	14.7	55.8	42.8	72.3	73.5	27.3	55.9	53.0	50.5
SSI-Net	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

Table 3. 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 [17]	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 [36]	-	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 [37]	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 [32]	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 [38]	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$_{rigid}$ [39]	-	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$_{deform}$ [39]	-	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 [22]	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

Detailed semantic segmentation results for each area on the S3DIS dataset with 6-fold cross-validation are shown in Table 4 to better represent the performance of our approach.

Table 4. Detailed results (%) for each area on S3DIS dataset with 6-fold cross-validation.

Testing Area	OA	mACC	mIoU	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

4.2.2. Results of SemanticKITTI

SemanticKITTI is a challenging dataset. Table 5 reports the quantitative results of SSI-Net on this dataset compared with some representative methods. The **mIoU** of our method achieves the best value of 55.4%, which surpasses most point-based methods [9,10,18,22,40,41] by a large margin. From Table 4, one can see that the projection-based methods [42–45] are superior to the point-based methods on the whole. However, our SSI-Net also outperforms these mentioned projection-based methods. For example, the **mIoU** value increases by 1.1% compared to the best projection-based method [45]. It obtains the maximum mIoU in 11 out of the 19 categories: particularly the values of the 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 5. 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 [17]	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 [9]	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 [40]	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
TangentConv [41]	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
SqueezeSeg [42]	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
SqueezeSegV2 [43]	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
DarkNet21Seg [44]	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 [44]	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
RangeNet53++ [46]	52.2	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 [47]	54.5	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
LatticeNet [48]	52.2	88.8	73.8	64.6	25.6	86.9	88.6	43.4	12.0	20.8	24.8	76.4	57.9	54.7	34.2	39.9	60.9	55.2	41.5	42.7
PointASNL [22]	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 [45]	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
MVP-Net [28]	53.9	91.4	75.9	61.4	25.6	85.8	92.7	20.2	37.2	17.7	13.8	83.2	64.5	69.3	50.0	55.8	12.9	55.2	51.8	59.2
SSI-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

Figures 4–6 present concerned illustrations of the SemanticKITTI dataset. Figure 4 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. Figure 5 shows the inference time on sequence 08 of SemanticKITTI and the overall mIoU to demonstrate the efficiency of the proposed method.

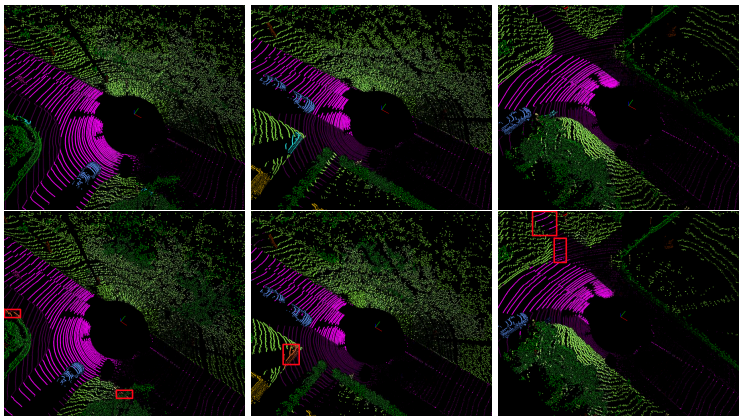

Figure 4. Qualitative results of our SSI-Net on the validation set of SemanticKITTI. Red rectangles represent the failure cases.

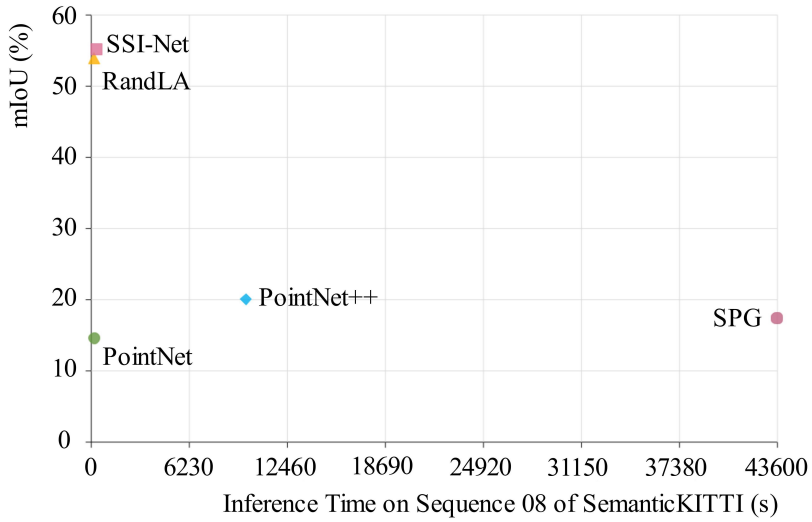

Figure 5. Inference time comparison on sequence 08 of SemanticKITTI and mIoU of SemanticKITTI of PointNet [17], PointNet++ [18], SPG [9], RandLA [10] and SSI-Net (ours).

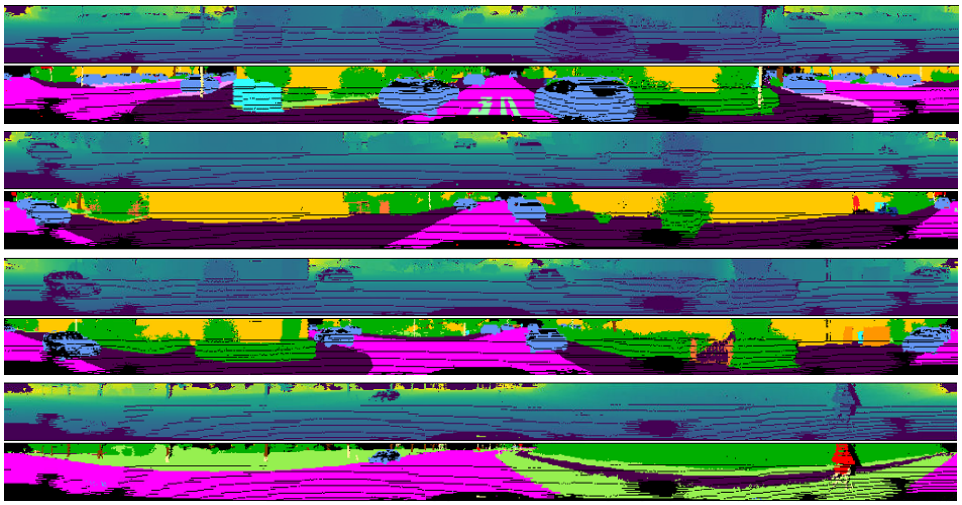

Figure 6. Qualitative results of online test on sequence 11 ∼ 21.

Sequences 11∼21 are used as a test set showing a large variety of challenging traffic situations and environment types. Figure 6 exhibits some qualitative results of online test (sequences 11∼21) in 2D panorama views.

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 of the S3DIS dataset and SemanticKITTI datset:

(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.

Results of this part are shown in Table 6: (1) The encoder with our full units reaches the best mIoU; (2) The greatest impact on mIoU is caused by the removing of an attentional semantic block probably because attention mechanism and neighboring deformation can help aggregate key semantic information discarded by random sampling.

Table 6. Ablation studies on Area 5 of S3DIS and SemanticKITTI (%)

	S3DIS			SemanticKITTI
Methods	**OA**	**mACC**	**mIoU**	**mIoU**
Removing semantic-aware information	86.1	72.2	63.6	53.8
Removing attentional semantic block	85.2	71.8	61.9	51.7
Max operation	86.7	73.0	63.1	52.9
With full units	88.3	73.2	65.1	55.4

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

$$g_i^k = M_s\{p_i^k, h_i^k, (p_i - p_i^k), \| p_i - p_i^k \|, (p_i - h_i^k), \| p_i - h_i^k \|\}. \quad (9)$$

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 neighboring 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 7 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 7. Selection for spatial location encoding test on Area 5 of S3DIS and SemanticKITTI (%).

	S3DIS	SemanticKITTI
Methods	**mIoU**	**mIoU**
(1) $\{p_i^k, h_i^k\}$	63.5	52.0
(2) $\{(p_i - p_i^k), \| p_i - p_i^k \|, (p_i - h_i^k), \| p_i - h_i^k \|\}$	65.0	53.1
(3) $\{p_i, (p_i - p_i^k), \| p_i - p_i^k \|, (p_i - h_i^k), \| p_i - h_i^k \|\}$	63.4	54.0
(4) $\{p_i^k, h_i^k, (p_i - p_i^k), \| p_i - p_i^k \|, (p_i - h_i^k), \| p_i - h_i^k \|\}$	65.1	55.4

FPS vs. RS: We compare the **mIoU** and **IoU** value of each class in this part, Table 8. 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 8. 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. Conclusions

This paper pays attention to large-scale semantic segmentation on point clouds to provide reliable information for indoor and outdoor semantic understanding such as autonomous vehicles that need detailed objects mapping in urban roads. Specifically, our proposed architecture SSI-Net is built on an SSC module that focuses on a more effective feature description via a spatial and semantic cross-correction manner. By mutually revising the neighboring information, robust representation can be obtained to support our work. Experimental results on S3DIS and SemanticKITTI datasets achieve the state-of-the-art

performance compared to relevant methods. However, this work also has its limitations. We only consider simple characteristics of point clouds (i.e., distance and relative position), which may face ambiguities of feature representation in more complex scenes. For instance, this kind of geometric representation cannot well handle the situation where different objects are cluttered closely in a table. In future work, we will focus on these problems by improving boundary information or acquiring more semantic expression with cross-modality segmentation et al., and we will further promote its combination with more 3D tasks such as 3D reconstruction, real-time localization and point cloud pose estimation.

Author Contributions: Y.Z. made substantial contributions to the conception, design of the work, and the analysis of the performance; J.Z. played an important role in revising the manuscript critically; J.M. and S.X. directed the main work. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Hebei Natural Science Foundation (Grant number [F2020202045]), in part by the National Natural Science Foundation of China (Nos. U21A20515, 62271074, 61972459, 61971418, U2003109, 62171321, 62071157, 62162044 and 32271983) and in part by the Open Research Fund of Key Laboratory of Space Utilization, Chinese Academy of Sciences (No. LSU-KFJJ-2021-05), and this work was supported by the Open Projects Program of National Laboratory of Pattern Recognition.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest. This work has not been published elsewhere in any form or language. The results presented in this article are clear and honest.

References

1. Sun, W.; Zhang, Z.; Huang, J. RobNet: real-time road-object 3D point cloud segmentation based on SqueezeNet and cyclic CRF. *Soft Comput.* **2020**, *24*, 5805–5818. [CrossRef]
2. Hao, F.; Li, J.; Song, R.; Li, Y.; Cao, K. Mixed Feature Prediction on Boundary Learning for Point Cloud Semantic Segmentation. *Remote Sens.* **2022**, *14*, 4757. [CrossRef]
3. Lan, S.; Yu, R.; Yu, G.; Davis, L.S. Modeling local geometric structure of 3d point clouds using geo-cnn. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019.
4. Wu, W.; Qi, Z.; Fuxin, L. Pointconv: Deep convolutional networks on 3d point clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019.
5. He, T.; Shen, C.; Hengel, A. Dynamic Convolution for 3D Point Cloud Instance Segmentation. *arXiv* **2021**, arXiv:2107.08392.
6. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph. (TOG)* **2019**, *38*, 1–12. [CrossRef]
7. Li, G.; Muller, M.; Thabet, A.; Ghanem, B. DeepGCNs: Can gcns go as deep as cnns? In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019.
8. Wang, L.; Huang, Y.; Hou, Y.; Zhang, S.; Shan, J. Graph attention convolution for point cloud semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019.
9. Landrieu, L.; Simonovsky, M. Large-scale point cloud semantic segmentation with superpoint graphs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA , 18–23 June 2018.
10. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. Learning Semantic Segmentation of Large-Scale Point Clouds with Random Sampling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 8338–8354. [CrossRef] [PubMed]
11. Boulch, A.; Guerry, J.; Le Saux, B.; Audebert, N. Snapnet: 3d point cloud semantic labeling with 2d deep segmentation networks. *Comput. Graph.* **2018**, *71*, 189–198. [CrossRef]
12. Tchapmi, L.; Choy, C.; Armeni, I.; Gwak, J.; Savarese, S. Segcloud: Semantic segmentation of 3d point clouds. In Proceedings of the 2017 International Conference on 3D Vision, Qingdao, China, 10 October 2017; pp. 537–547.
13. Wang, P.S.; Liu, Y.; Guo, Y.X.; Sun, C.Y.; Tong, X. O-cnn: Octree-based convolutional neural networks for 3d shape analysis. *ACM Trans. Graph. (TOG)* **2017**, *36*, 1–11. [CrossRef]
14. 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, Seoul, Korea, 27 October–2 November 2019.

15. 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, Honolulu, HI, USA, 21–26 July 2017.
16. Park, J.; Kim, C.; Jo, K. PCSCNet: Fast 3D Semantic Segmentation of LiDAR Point Cloud for Autonomous Car using Point Convolution and Sparse Convolution Network. *arXiv* **2022**, arXiv:2202.10047.
17. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp 652–660.
18. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In Proceedings of the 2017 Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
19. Zhang, D.; He, F.; Tu, Z.; Zou, L.; hen, Y. Pointwise geometric and semantic learning network on 3D point clouds. *Integr.-Comput.-Aided Eng.* **2020**, *27*, 57–75. [CrossRef]
20. Zhao H, Jiang L, Fu C W, et al. Pointweb: Enhancing local neighborhood features for point cloud processing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019.
21. Han W.; Wen, C.; Wang, C.; Li, X.; Li, Q. Point2Node: Correlation learning of dynamic-node for point cloud feature modeling. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 10925–10932. [CrossRef]
22. Yan, X.; Zheng, C.; Li, Z.; Wang, S.; Cui, S. PointASNL: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seoul, Korea, 14–19 June 2020.
23. Chen, J.; Kakillioglu, B.; Velipasalar, S. Background-Aware 3D Point Cloud Segmentation with Dynamic Point Feature Aggregation. *arXiv* **2021**, arXiv:2111.07248.
24. Chen, C.; Chen, Z.; Zhang, J.; Tao, D. SASA: Semantics-Augmented Set Abstraction for Point-based 3D Object Detection. *arXiv* **2022**, arXiv:2201.01976.
25. Srivastava, S.; Sharma, G. Exploiting local geometry for feature and graph construction for better 3d point cloud processing with graph neural networks. In IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 5 June 2021.
26. Rethage, D.; Wald, J.; Sturm, J.; Navab, N.; Tombari, F. Fully-convolutional point networks for large-scale point clouds. In Proceedings of the 15th European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
27. Chen, S.; Niu, S.; Lan, T.; Liu, B. PCT: Large-scale 3D point cloud representations via graph inception networks with applications to autonomous driving. In Proceedings of the 26th IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019.
28. Luo, C.; Li, X.; Cheng, N.; Li, H.; Lei, S.; Li, P. MVP-Net: Multiple View Pointwise Semantic Segmentation of Large-Scale Point Clouds. *arXiv* **2022**, arXiv:2201.12769.
29. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA , 27–30 June 2018.
30. Gong, J.; Xu, J.; Tan, X.; Zhou, J.; Qu, Y.; Xie, Y.; Ma, L. Boundary-aware geometric encoding for semantic segmentation of point clouds. *arXiv* **2021**, arXiv:2101.02381.
31. Wang, X.; He, J.; Ma, L. Exploiting local and global structure for point cloud semantic segmentation with contextual point representations. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 4571–4581.
32. Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; Chen, B. Pointcnn: Convolution on X-transformed points. In Proceedings of the 2018 Conference on Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018.
33. Wang, S.; Suo, S.; Ma, We.; Pokrovsky, A.; Urtasun, R. Deep parametric continuous convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA , 18–23 June 2018.
34. Choy; Christopher; Gwak, J.; Savarese, S. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019.
35. Liu, Y.; Hu, Q.; Lei, Y.; Xu, K.; Li, J.; Guo, Y. Box2Seg: Learning Semantics of 3D Point Clouds with Box-Level Supervision. *arXiv* **2022**, arXiv:2201.02963.
36. Huang, Q.; Wang, W.; Neumann, U. Recurrent slice networks for 3d segmentation of point clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA , 18–23 June 2018.
37. Ye, X.; Li, J.; Huang, H.; Du, L.; Zhang, X. 3D recurrent neural networks with context fusion for point cloud semantic segmentation. In Proceedings of the 15th European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
38. Zhang, Z.; Hua, B.S.; Yeung, S.K. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019.
39. Thomas, H.; Qi, C.R.; Deschaud, J.; Marcotegui, B.; Goulette, F.; Guibas, L.J. Kpconv: Flexible and deformable convolution for point clouds. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019.
40. Su, H.; Jampani, V.; Sun, D.; Maji, S.; Kalogerakis, E.; Yang, Mi.; Kautz, J. SPLATNet: sparse lattice networks for point cloud processing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA , 18–23 June 2018.
41. Tatarchenko, M.; Park, J.; Koltun, V.; Zhou, Q. Tangent convolutions for dense prediction in 3D. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA , 18–23 June 2018.

42. Wu, B.; Wan, A.; Yue, X.; Keutzer, K. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3D lidar point cloud. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018.
43. Wu, B.; Zhou, X.; Zhao, S.; Yue, X.; Keutzer, K. Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In Proceedings of the International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019.
44. Behley, J.; Garbade, M.; Milioto, A.; Quenzel, J.; Behnke, S.; Stachniss, C.; Gall, J. SemanticKITTI: A dataset for semantic scene understanding of lidar sequences. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019.
45. Zhang, Y.; Zhou, Z.; David, P.; Yue, X.; Xi, Z.; Gong, B.; Foroosh, H. 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, Seattle, WA, USA, 13–19 June 2020.
46. Milioto, A.; Vizzo, I.; Behley, J.; Stachniss, C. RangeNet++: Fast and accurate lidar semantic segmentation. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macao, China, 3–8 November 2019.
47. Cortinhal, T.; Tzelepis, G.; Aksoy, E.E. SalsaNext: Fast semantic segmentation of lidar point clouds for autonomous driving. *arXiv* **2020**, arXiv:2003.03653.
48. Rosu, R.A.; Schütt, P.; Quenzel, J.; Behnke, S. LatticeNet: Fast point cloud segmentation using permutohedral lattices. *arXiv* **2019**, arXiv:1912.05905.