A Category-Contrastive Guided-Graph Convolutional Network Approach for the Semantic Segmentation of Point Clouds

Xuzhe Wang, Juntao Yang, Zhizhong Kang, Junjian Du, Zhaotong Tao, and Dan Qiao

***Abstract*—The semantic segmentation of light detection and ranging (LiDAR) point clouds plays an important role in 3-D scene intelligent perception and semantic modeling. The unstructured, sparse and uneven characteristics of point clouds pose great challenges to the representation of the local geometric shapes, which degrades semantic segmentation performance. To address the challenges of describing local geometric shapes due to unstructured and sparse 3-D point clouds, this article proposes a category-contrastive-guided graph convolutional network (CGGC-Net) for the semantic segmentation of LiDAR point clouds. First, a detailed geometric structure of the raw point clouds is encoded to represent the inherent geometric pattern within the local neighborhood. At the same time, the geometric structures information is transmitted across multiple layers, so that the geometric structure encoding information containing different receptive fields and richer neighborhood spatial structure can be aggregated. Following this, the graph convolution neural network uses the edge convolution layer to adaptively describe the semantic correlation between the query point and its neighboring points, and combines the attention mechanism to gather the surrounding feature information to the query point. As a result, the graph convolution neural network and attention mechanism are iteratively stacked for the aggregation and fusion of spatial context semantic information, to generate highly discriminative semantic feature representation. Finally, the superparameters of the model are learned through a multitask optimization strategy guided by category-aware contrastive loss and cross-entropy loss. Experiments are conducted on the public SemanticKITTI dataset and the Stanford large-scale 3-D Indoor Spaces dataset to demonstrate the effectiveness and reliability of the proposed CGGC-Net from both quantitative and qualitative perspectives. The results indicate its capability of automatically classifying LiDAR point clouds, with a mean intersection-over-union of 58.4%. Moreover, multiple comparative experiments also demonstrate the superior performance of the proposed method, exceeding state-of-the-art methods.**

***Index Terms*—Attention mechanism, contrastive learning, graph convolutional network, light detection and ranging (LiDAR), semantic segmentation.**

I. Introduction

LIGHT detection and ranging (LiDAR) point clouds have increasingly attracted interests in numerous applications, especially autonomous driving [1], [2], virtual reality and robotics [3], [4], due to their superior ability to preserve the spatial detail information of objects or sceneries [5], [6], [7]. In these applications, fine-grained classification, which assigns semantic labels to each point that belongs to the objects of interest, is a fundamental and important task. This detailed semantic information plays an important role in the downstream tasks, such as place recognition [8], instance segmentation [9], and scene reconstruction [10]. Therefore, the automatic fine-grained classification of LiDAR point clouds has been an active topic.

To date, many methods have been developed for the semantic segmentation of 3-D point clouds. Traditionally, machine learning-based approaches (e.g., support vector machines [11] and random forests [12]), where hand-crafted features are designed for representing the geometric structure information of point clouds, have been adopted for the semantic segmentation of point clouds. Although these approaches have shown the capability of automatically classifying point clouds, their performance is limited by the descriptive ability of the designed hand-crafted features and the reliability of the selected classifier.

More recently, deep learning techniques have demonstrated excellent abilities in various computer vision and natural language processing fields and are increasingly popular in scene understanding tasks, such as classification, object detection and instance segmentation, based on point clouds. Due to the discrete and disordered data characteristics of point clouds, it is challenging to directly implement classic convolutional neural networks on raw point clouds. Some solutions that transform raw point clouds into regular representations, such as projected images and structured volumetric grids [13], [14], have been presented, which then serve as the input of classic convolutional neural

Manuscript received 30 December 2022; revised 7 March 2023 and 28 March 2023; accepted 29 March 2023. Date of publication 5 April 2023; date of current version 17 April 2023. This work was jointly supported by the National Natural Science Foundation of China under Grant 42201486 and Grant 42271436. *(Corresponding author: Juntao Yang.)*

Xuzhe Wang, Junjian Du, Zhaotong Tao, and Dan Qiao are with the College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao 266590, China (e-mail: 2419128127@qq.com; 3262604200@qq.com; tzt427@163.com; 1960642531@qq.com).

Juntao Yang is with the College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao 266590, China, and also with the Subcenter of International Cooperation and Research on Lunar and Planetary Exploration, Center of Space Exploration, Ministry of Education of The People's Republic of China, Beijing 100083, China (e-mail: jtyang@sdust.edu.cn).

Zhizhong Kang is with the School of Land Science and Technology, China University of Geosciences, Beijing 100083, China, with the Lunar and Planetary Remote Sensing Exploration Research Center, China University of Geosciences, Beijing 100083, China, and also with the Subcenter of International Cooperation and Research on Lunar and Planetary Exploration, Center of Space Exploration, Ministry of Education of The People's Republic of China, Beijing 100083, China (e-mail: zzkang@cugb.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2023.3264240

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

networks for semantic segmentation. Although these solutions enhance the descriptiveness and discriminativeness of feature representation to some extent, the local geometric structures and fine-grained semantic contexts are difficult to preserve due to the regular transformation.

To directly carry out an end-to-end deep learning model on raw point clouds for extracting pointwise semantic features, pioneering networks, such as PointNet [15] and PointNet++ [16] have paved the way using shared multilayer perceptrons (MLPs). However, they ignore semantic correlations between neighbors. Subsequently, many researchers have devoted efforts to encoding semantic correlations between neighbors [17], [18], [19], [20] and fusing multiscale semantic features [21], [22], due to the different semantic contexts at different scales [23]. For example, graph convolution [24], [25] and attention pooling mechanisms [20], [26], [27], [28], [29], [30] were used to generate promising classification results by efficiently aggregating contextual semantic information.

Although there have been many deep learning-based methods presented for the semantic segmentation of point clouds in recent years, this task is remarkably challenging due to the following aspects. First, the efficient aggregation of rich semantic information at various scales remains difficult due to unstructured data characteristics. Although graph convolution-based methods have been explored for the semantic segmentation of point clouds in recent years [17], [18], [31], capturing local geometric patterns and aggregating spatial context is necessary, especially for dynamic and large scenarios [27], [32], [33]. Second, although semantic supervised labels effectively improve the descriptiveness of feature representation via an end-to-end deep learning architecture, most approaches update the model superparameters to converge by only comparing their predictions with the associated semantic labels. Few studies have focused on explicitly using semantic supervised information to guide the process of generating high-level semantic feature representations of point clouds.

To address the aforementioned issues, this article develops a category-contrastive guided graph convolutional network (CGGC-Net) method for the semantic segmentation of LiDAR point clouds. Moreover, both quantitative and qualitative analyses are conducted on the public SemanticKITTI dataset benchmark [34] and S3DIS dataset [35] to evaluate its robustness and reliability. Our contributions in this article are as follows.

1) The spatial context information from both the detailed geometric structure and semantic feature are locally aggregated for each point in parallel using a graph convolution module and attention mechanism as the receptive field progressively increases, to generate highly discriminative semantic feature representation.
2) A category-contrastive loss is designed to guide the learning process of semantic feature representation, which would make the semantic features of the same class remain close while put those of different classes far apart. Moreover, combined with the cross-entropy loss, a multitask optimization strategy can jointly utilize the discrepancies among different categories to highly-discriminative and descriptive semantic representation.

The rest of this article is organized as follows. The related works are briefly reviewed in Section II. Section III describes the developed LiDAR point cloud classification framework in detail. Section IV presents the experimental results and an analysis for both quantitatively and qualitatively evaluating the developed method. Finally, Section V concludes this article.

II. Related Work

Semantic segmentation, especially for large scale scenes, has been an active topic. However, this is also challenging for accurate semantic classification in large scenes due to complex elements, varieties of scene classes, occlusions, and noise. In recent years, deep learning techniques have become increasingly popular since they can produce promising interpreted results. In this section, we review the relevant literature, which can be generally divided into four groups: projected image methods, voxel-based methods, point-based methods, and graph-based methods.

A. *Projected Image Methods*

Due to the great success of 2-D images semantic segmentation [36], [37], unordered and unstructured 3-D point clouds have been initially projected onto regular and structured 2-D images. Subsequently, numerous mature deep neural networks for 2-D image semantic segmentation could have been used for pixelwise labeling. 2-D multiview synthetic images have been generated from point clouds [38], [39], [40], [41], and multistream convolutions on different views have been applied for labeling the semantic information, which is then reprojected to each point. In addition, spherical projections, such as SPLAT-Net [42] and SequeezeSeg [43], have been used to alleviate geometric information loss during preprocessing. To consider the uneven distribution of point clouds in grid cells, polar bird's-eye-view representations, such as PolarNet [44], have been defined through the polar coordinate system. Although mature 2-D semantic segmentation methods can be implemented on regular and structured projected images, geometric information is inevitably missing during projection transformation, which can inhibit the classification quality.

B. *Voxel-Based Methods*

Voxel-based methods, such as VoxNet [13] and OctNet [45], convert discrete 3-D point clouds into volumetric occupancy grids whose feature maps can be generated through a 3-D convolutional neural network (3-D CNN). It is obvious that as the resolution of voxelization increases, richer geometric information can be retained, which results in excessive memory consumption and a heavy computational cost. To ease this issue, sparse convolution has been designed and implemented on flexible unbalanced octrees that adaptively partition 3-D point clouds based on sparsity [45], [46], [47]. In fact, voxel-based methods extend the success of 2-D convolution into 3-D space, which can effectively deal with unstructured point clouds. Similar to projected image-based methods, voxel-based methods inevitably lead to information loss although they retain 3-D information to some extent.

C. Point-Based Methods

Point-based methods carry out the convolution operation directly on unordered and irregular point clouds. PointNet [15] was a pioneering model which that utilized on unordered and irregular point clouds. Although it enhanced the feature representation capability by directly using raw point clouds, PointNet ignored the spatial context without taking neighborhood information into consideration. Subsequently, PointNet++ [16] applied a ball-query module to extract and aggregate local features using a hierarchical structure. Nevertheless, PointNet++ still lost the relationship between points within a ball-query set. To this end, other works [24], [25] concentrated on how to carry out an effective and efficient convolutional operator directly on raw point clouds via graph convolutional networks and attention mechanisms [22], [27], which augments and fuses feature maps of multiple resolutions for large-scale semantic segmentation.

D. Graph-based Methods

Point clouds inherently lack topological information, so designing a model to recover topology can enrich the representation power of point clouds. To better exploit semantic relevance between neighbors, numerous studies have focused on relationship modeling via graph structures or attention mechanisms, where semantic context could then be extracted and aggregated into the corresponding center points. Graph neural networks (GNN) were first proposed by Hu et al. [27], and have been widely used in different fields, including semantic understanding [49], medical neuroimaging [50] and social networks [51], to describe the local and global contexts within unstructured data in recent years. For instance, superpoint graph (SPG) [52] was constructed to realize semantic segmentation in a large scene. However, this required extra preprocessing for segmenting the superpoints, and the labeling quality in large scenarios was unsatisfactory. Wang et al. [18] proposed the graph attention convolutional network, where adaptive weights were assigned to different neighbors through a self-attention mechanism, and then the local spatial context could be aggregated using adaptive pooling to automatically classify point cloud data. Wang et al. [31] developed a dynamic graph convolutional neural network that adopted edge convolution to extract and dynamically update local semantic features through the characteristic relationship between center points and neighbors. Liu et al. [32] explored the graph convolutional network to preserve rich geometric details and capture long spatial dependencies for enhancing the network feature representation.

To summarize, inspired by [22], [27], [32], our work uses a graph convolutional neural network as the baseline and is dedicated to semantic segmentation directly on raw LiDAR point clouds. Unlike previous GNNs that focused on updating semantic features and neglected the detailed geometric structure information [18], [22], [31], [32], our work encodes detailed geometric structure information into semantic features and aggregates the long-range spatial context as the receptive fields are expanded and stacked. In addition, our work further extends the applications of contrastive learning on a small number of objects such as 2-D images [53], [54] to the semantic segmentation of massive 3-D point clouds and verifies the effectiveness of the category-aware optimization strategy in the point cloud domain.

III. Methodology

To capture local geometric patterns and aggregate spatial context effectively and efficiently, this article follows the encoder-decoder architecture [16], [20], [27], [37] and develops a CGGC-Net method for the semantic segmentation of LiDAR point clouds, which has an encoder network and a corresponding decoder network, followed by a final point-wise classification layer. Fig. 1 illustrates the pipeline of the proposed CGGC-Net for the semantic segmentation of point clouds.

The encoder network consists of four encoder layers; the detailed geometric structure and semantic feature are locally aggregated for each point in parallel using graph convolution module and attention mechanism as the receptive field progressively increases through iterative stacking. Moreover, the detailed geometric structure information is transmitted across multiple encoder layers to effectively preserve complex local geometric patterns. Consequently, multiple encoder layers are progressively stacked for the aggregation and fusion of spatial context information to generate highly discriminative semantic feature representation. Each encoder layer has a corresponding decoder layer, and thus the decoder network also has four layers. The encoder layer and its corresponding decoder layer are connected through skip connections, which combine deep, semantic, coarse-grained feature maps from the decoder layer with shallow, low-level, fine-grained feature maps from the encoder layer. Finally, the decoder output is fed into a classification layer, consisting of three fully connected layers and a multiclass softmax classifier, to produce class probabilities for each point independently. The superparameters of the model are learned through a multitask optimization strategy guided by category-aware contrastive loss and cross entropy loss. As a result, the raw point clouds are interpreted to obtain the final pointwise labeling results. More details of the proposed CGGC-Net are given below.

A. Detailed Geometric Structure Encoding

In this section, a detailed geometric structure encoding module is designed to describe inherent spatial relations within the local neighborhood and preserve complex local geometric patterns as much as possible, which enhances the expression and refinement of the subsequent semantic features.

1) Local Geometric Structure Descriptor: X-Y-Z coordinate information is incapable of directly describing complex local geometric patterns since relative spatial relations between the query point and its neighbors are unexplored. Thus, inspired by a previous work [32], we use a local geometric structure descriptor to explicitly represent their potential geometric structure within the local neighborhood directly on 3D coordinates.

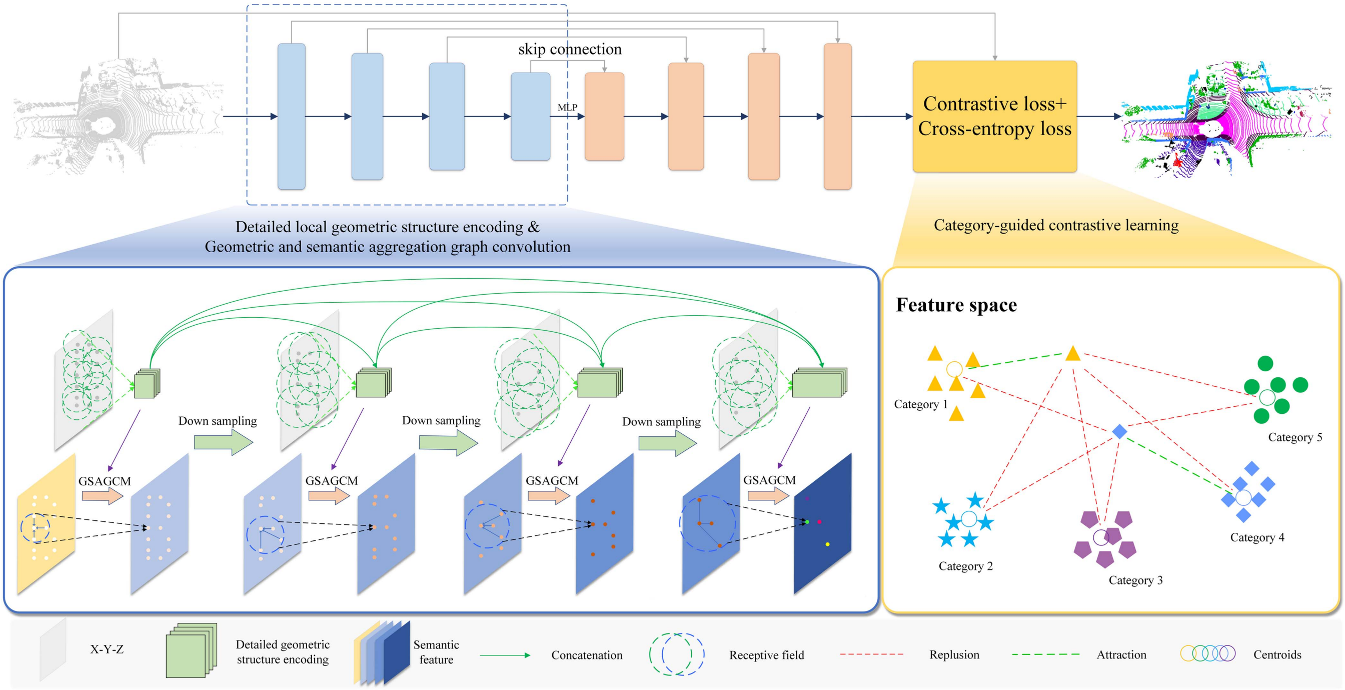


Fig. 1. Pipeline of the proposed CGGC-Net for semantic segmentation of point clouds.

A tensor $P = [p_1, p_2, \cdots, p_n]^T$ is defined to represent a set of point clouds, where p_i denotes the i^{th} point. For the query point p_i, its neighboring points are gathered using a simple K-nearest neighbors (KNN) algorithm based on Euclidean distances, as $[p_i^1, p_i^2, \cdots, p_i^K]$ where K denotes the number of neighbors. Following this, we adopt (1) to describe the geometric structure of neighborhood

$$\boldsymbol{r}_i = \text{Concat}\left[p_i, p_i^k, \|p_i - p_i^k\|, \left(p_i - p_i^k\right)\right], \boldsymbol{r}_i \in \mathbb{R}^{K\times 10} \tag{1}$$

where p_i denotes each center point, p_i^k denotes the K neighboring points of the query point. $\| \cdot \|$ represents the Euclidean distance between the query point and its neighboring points, $(p_{\text{i}} - p_i^k)$ reflects the 3-D coordinates difference between p_i and p_i^k, and Concat$[\cdot]$ denotes the concatenation operation. As a consequence, $\boldsymbol{r} \in \mathbb{R}^{N\times K\times 10}$ is encoded as the representation of spatial relationships between the query point and its neighboring points from redundant 3-D coordinates.

To efficiently aggregate the neighboring relations, we use attentive pooling [27], which adaptively allocates a unique attention score to different neighbors, to automatically learn and select the salient geometric structures, as defined

$$\boldsymbol{g} = \text{AttentivePool}\left(\boldsymbol{r}\right), \boldsymbol{g} \in \mathbb{R}^{N\times 10} \tag{2}$$

where AttentivePool$(\cdot)$ denotes the attentive pooling function consisting of a shared MLP followed by softmax. To summarize, given the input point cloud P, an informative feature vector $\boldsymbol{g} \in \mathbb{R}^{N\times 10}$ in the first layer is generated to effectively describe complex local geometric patterns.

2) Local Geometric Structure Transmission: To capture complex local structure patterns, a series of downsampling operations is performed to alleviate the limited the size of receptive field. With the encoder layer deepening through downsampling operations, the receptive field of each point increases. In this way, richer local structures are progressively aggregated due to wider context information for each point. It is inevitable that fine-grained spatial relationships might be lost. Therefore, the geometric structure feature $\boldsymbol{g}$ is transmitted across multiple encoder layers to effectively preserve complex local geometric patterns, so that it is efficiently augmented and enriched with a combination of different receptive fields, which provides a fundamental spatial basis for mining the semantic correlation between neighboring points of discrete 3-D point clouds. Finally, the detailed local geometric structure encoding in the t^{th} layer can be represented as

$$\begin{aligned}\tilde{\boldsymbol{g}}^1 &= \boldsymbol{g}\\ \tilde{\boldsymbol{g}}^t &= \text{Concat}\left[\text{DS}(\tilde{\boldsymbol{g}}^{t-1}), \boldsymbol{g}^t\right]\end{aligned} \tag{3}$$

where DS denotes down-sampling operation, Concat$[\cdot]$ denotes the concatenation operation, $1 \leq t \leq 4$ in this article. Fig. 2 illustrates differences of local structure patterns under different receptive fields.

B. Geometric and Semantic Aggregation Graph Convolution Module (GSAGCM)

In this section, we design a graph convolutional neural network module to produce new semantic features by aggregating the neighboring semantic information, which takes the detailed local geometric structure encoding $\tilde{\boldsymbol{g}}$ and semantic features $\boldsymbol{F}$ as the inputs. Initially, the semantic features are embedded from 3-D coordinates using a simple MLP operation. In the GSAGCM, the propagated edge convolution (PEConv) is used to extract the semantic feature relationship between the query

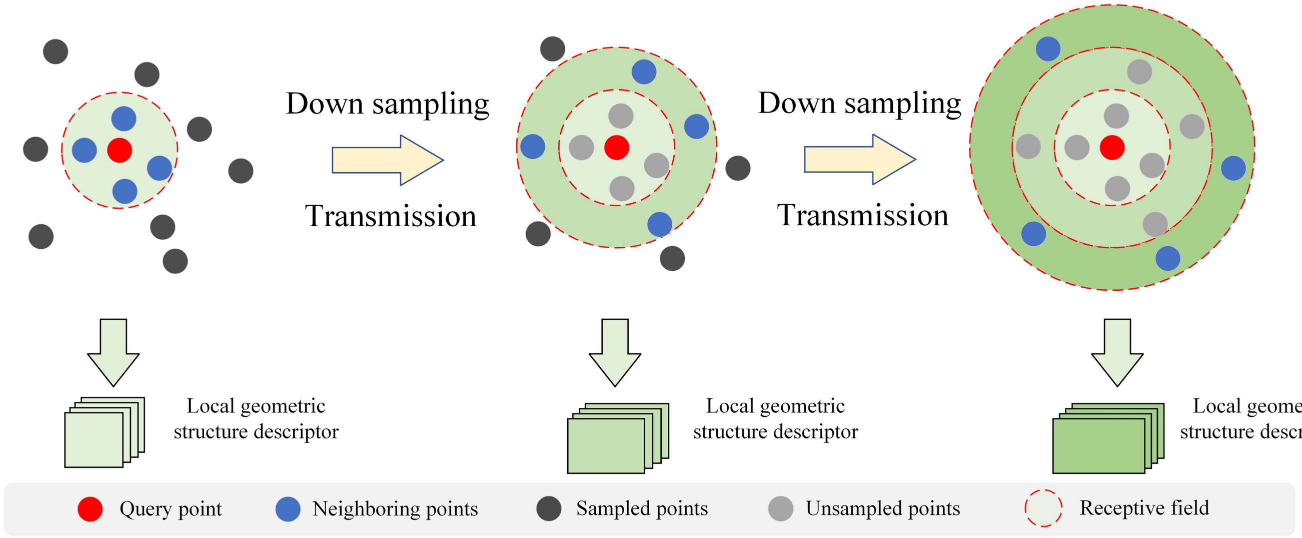


Fig. 2. Differences of local structure patterns under different receptive fields, where three layers are illustrated.

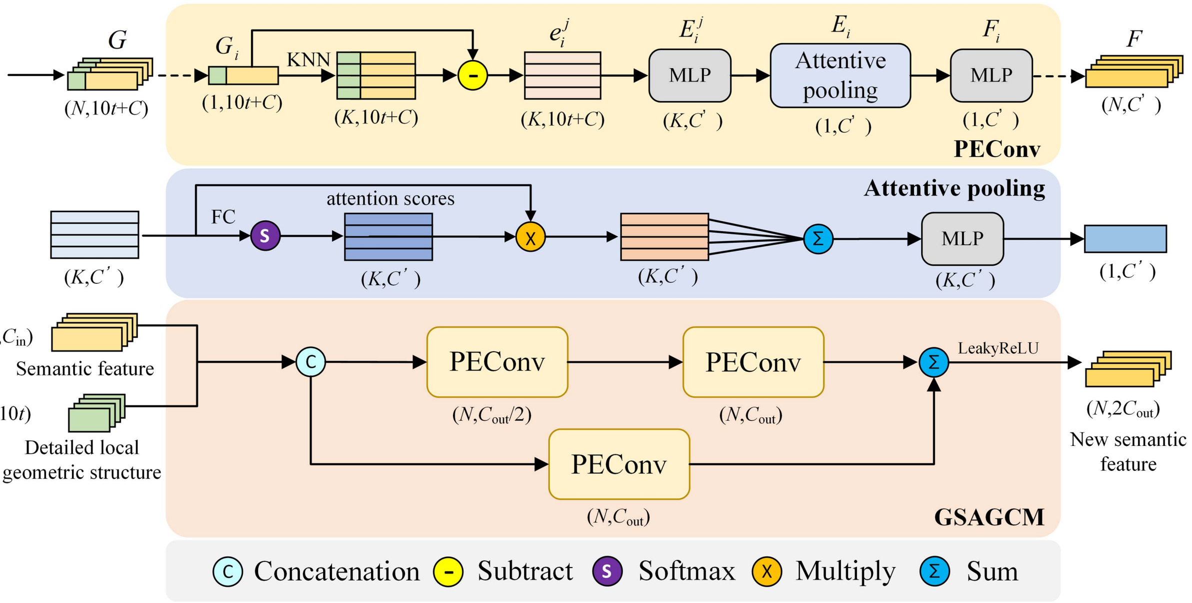


Fig. 3. Pipeline of GSAGCM.

point and its neighbors, and aggregates the neighboring feature information to the query point through attention pooling. Finally, detailed local geometric structures and semantic features are fused by stacking several layers with residual connection to update the new semantic feature per point. Different from original RandLA-Net [27], we adopt attentive pooling to aggregate the encoded local spatial information into the associated query point. In addition, after multiple transmissions, we use a further augmented local geometric structure to induce the expression and refinement of semantic features with the help of graph convolution rather than a single layer of MLP.

1) Propagated Edge Convolution for Feature Aggregation: Within the local neighborhood, PEConv and attention pooling are used to achieve the extraction and transmission of neighborhood information. This consists of three parts: graph model construction; edge feature representation; and edge feature aggregation. As a result, a new semantic feature per point is produced, which aggregates newer semantic features using PEConv and attention pooling (see Fig. 3) or serves as the input of the subsequent encoder layer with the detailed local geometric structure encoding (see Fig. 1).

1) *Graph Model Construction*: Unlike 2-D raster images, 3-D point clouds are discrete and disordered, and there is no explicit topological relationship between points. However, points that are adjacent to each other in Euclidean space usually have interaction relationships. In addition, for a specific point, the geometric structure formed by its several neighboring points is the foundation of semantic mining. As mentioned above, we obtain the index of the

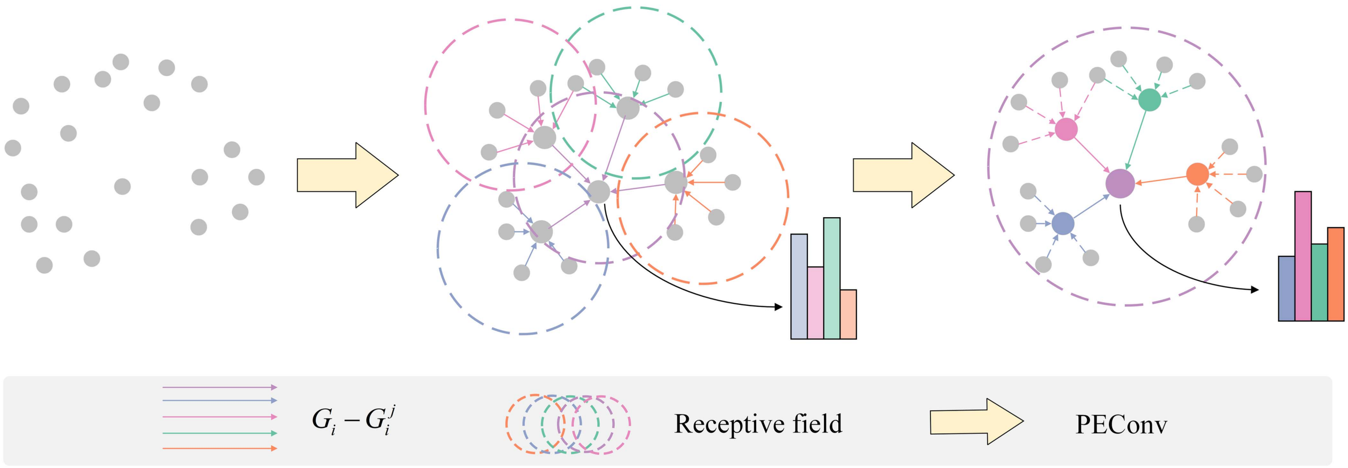


Fig. 4. Illustration of the increasing size of receptive field when stacking.

K nearest points of each point by KNN, and establish the directed edge between the query points and the neighbors.

2) *Edge Feature Representation*: Many graph-based networks stack both global and local information as their edge representations. What distinguishes us from them is that the local geometric structure is also included in addition to the semantic feature used in building undirected edges. Considering that the global information has been embodied in $\boldsymbol{g}$, we eventually use the difference between the query points and neighbors, which can be calculated as

$$\boldsymbol{G} = [\tilde{\boldsymbol{g}}, \boldsymbol{F}] \tag{4}$$

$$e_i^j = \boldsymbol{G}_i - \boldsymbol{G}_i^j, e_i^j \in \mathbb{R}^{10\times t+C} \tag{5}$$

where $\boldsymbol{G}_i^j$ represents the geometric and semantic stacked feature of the j^{th} point in the corresponding neighborhood of the i^{th} point.

Ultimately, we extract the edge attribute features from e by means of a three-layer successive stacked MLP, which can be expressed as follows:

$$\boldsymbol{E}_i^j = h_\Theta \left(e_i^j\right) \tag{6}$$

where h_{Θ_1} denotes feature learning of $\mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^{d'}$, d is the feature dimension, and Θ_1 denotes the learnable weights of the multiple groups.

3) *Edge Feature Aggregation:* To aggregate edge attribute features into the query point while avoiding the loss of important edge information, we introduce a self-attention mechanism to adaptively learn the unique score of each edge attribute and maximize the characterization of the edges they contain. The aggregated feature of the query point can be calculated, as defined in

$$\boldsymbol{F}_i = h_{\Theta_2} \left(\text{AttPool}\left(\boldsymbol{E}_i\right)\right) \tag{7}$$

where Θ_2 is also a group of learnable weights.

2) *Residual Connected and Dilate Stacked Module:* To expand the receptive fields, many existing works [55], [56] optimized the K-nearest searching strategy, which was required to search more neighboring points in different receptive regions and select a fixed number of neighboring points regularly. Undoubtedly, these approaches would create additional memory costs on searching more nearest points. In this section, we stack multiple propagated edge convolutional layers to increase the receptive field by means of feature propagation. Moreover, to address the problem of gradient vanishing and model degradation in deep neural networks, a PEConv is used as our residual connection rather than MLP [57], [58].

Fig. 4 illustrates the increasing size of the receptive field when stacking. When the PEConv is first performed on the input $\boldsymbol{G}$, the receptive field of each point is the corresponding number of neighborhoods K. In regard to the second layer, although the number of neighborhoods remains constant, the actual receptive field becomes K^2 since the semantic feature of neighbors has contained information of its own neighborhood in the previous layer. As a result, the size of the receptive field is repeatedly expanded through feature aggregation within the local neighborhood. In this article, we ultimately stack 2 layers.

C. Category-contrastive and Cross-Entropy Guided Optimization Strategy

It is well known that the superparameters within the whole network are learned for mapping a set of inputs to a set of outputs from massive high-quality labeled training data. Generally, the problem of learning is cast as an optimization problem, which navigates the space of possible sets of superparameters within the whole network to produce the satisfactory predictions. Here, we present a category-contrastive and cross-entropy guided optimization strategy to search for a candidate solution with the optimal values.

For the multitask classification of point clouds, the most typical and effective method is to describe the cross-entropy loss between predictions and the ground truth, which can be calculated as follows:

$$L_{\text{cro}} = -\sum_{i=1}^{V} y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i) \tag{8}$$

where y_i denotes the predictions, $\hat{y}_i$ denotes the ground truths, and V represents the number of categories. Consequently, the predicted probability distribution gradually approaches the true probability distribution by minimizing L_{cro} through gradient back-propagation.

However, cross-entropy loss ignores the relations between categories themselves. In fact, class separation in the latent feature space would also be an ideal characteristic to discriminate among different categories. Theoretically, feature vectors of the same category should remain close in latent feature space, while those of different categories should be far apart. Therefore, based on output feature μ_v generated by the encoder-decoder architecture belonging to class v, we design a category-guided contrastive loss that is devoted to depicting the category-specific distance from the centroid representation of each class $\boldsymbol{\delta}_i$. It can be shown as

$$L_{\text{cont}} = \sum_{i=0}^{V} l(\boldsymbol{\delta}_i, \mu_v) \tag{9}$$

$$l\ (\boldsymbol{\delta_i}, \mu_v) = \begin{cases} D(\boldsymbol{\delta_i}, \mu_v), & i = v \\ \max\{0, \Delta - D(\boldsymbol{\delta}_i, \mu_v)\}, & \text{otherwise} \end{cases} \tag{10}$$

where $D(\cdot)$ denotes Euclidean, cosine or any other distance function, and Δ represents the maximum distance of the same class and the minimum distance of different classes.

In the specific implementation, we define a fixed size of tensor $\boldsymbol{\beta}_i \in \mathbb{R}^{S\times D}$ per category i for storing the corresponding features, where D is the dimension of μ_v and S represents the maximum number of stored features. In addition, we randomly select N points for updating the centroid representation of each category, which strikes a balance between effectiveness and efficiency. The centroid representations $\boldsymbol{\delta}_i^{\text{new}}$ is calculated based on $\boldsymbol{\beta}_i$ every I_{p} iterations. To avoid rapid fluctuation, we set a momentum m between $\boldsymbol{\delta}_i$ and $\boldsymbol{\delta}_i^{\text{new}}$ so that the centroid in the feature space can evolve steadily in an end-to-end manner, which is formulated as:

$$\boldsymbol{\delta}_i^{\text{new}\prime} = m \times \boldsymbol{\delta}_i + (1 - m) \times \boldsymbol{\delta}_i^{\text{new}}. \tag{11}$$

Finally, the total loss function can be represented as

$$L_{\text{total}} = \lambda \times L_{\text{cont}} + L_{\text{cro}} \tag{12}$$

where λ is a weight between category-guided contrastive loss L_{cont} and cross-entropy loss L_{cro}. As a result, the differences between the prediction and the ground truth are measured and the superparameters of the model are updated using the stochastic gradient descent algorithm so that the next evaluation reduces the differences, which enables the superparameters of the model to move toward convergence.

IV. Experimentation and Analysis

A. Experimental Dataset and Evaluation Metrics

To verify the effectiveness and reliability of the proposed approach, we select two well-known public dataset benchmarks: SemanticKITTI dataset [34] and the Stanford large-scale 3-D Indoor Spaces dataset [35]. Raw point clouds are manually classified into 19 categories as ground truths and the 3-D point cloud data only presents *X-Y-Z*, intensity information without RGB information. S3DIS dataset is divided into six large-scale indoor areas, containing more than 215 million labeled points. And raw points are manually annotated into 13 categories. Each point has 3-D coordinates and RGB information. According to previous work [27], [32], we adopt six-fold cross-validation for evaluation. To measure the classification quality, we conduct the quantitative evaluation using intersection over union (IoU) per class, mean IoU (mIoU), overall accuracy, mean accuracy over classes (mAcc) and Kappa as defined in (13)–(17). IoU is a measure which imposes the penalty of false positive on the class accuracy per class, and the mean IoU is the IoU over union in all classes. Overall accuracy denotes the sum of the true positives plus true negatives divided by the total number of queried individuals. And mAcc denotes the sum of the true positives plus true negatives divide by the total number of queried individuals, which reflects the proportion of the correct samples identified by the classifier to all samples

$$\text{IoU} = \frac{\text{TP}_i}{\text{GT}_i + \text{FP}_i} \tag{13}$$

$$\text{mIoU} = \frac{\sum_{i=1}^{C} \text{IoU}_i}{C} \tag{14}$$

$$\text{Overall accuracy (OA)} = \frac{\text{TP} + \text{FN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{15}$$

$$\text{mAcc} = \frac{1}{C} \cdot \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}} \tag{16}$$

$$\text{Kappa} = \frac{p_o - p_e}{1 - p_e} \tag{17}$$

where TP denotes the number of positives that are correctly classified as positives, TN denotes the number of positives that are correctly classified as negatives, FN denotes the number of negatives that are incorrectly classified as negatives, and FP denotes the number of negatives that are incorrectly classified as positives, TP_i, GT_i, and FP_i denote the number of positives that are correctly classified as positives, ground truth and the number of negatives that are incorrectly classified as positives in the class *i*, respectively. p_o is the overall accuracy, and p_e can be denoted as

$$p_e = \frac{\sum_{i=1}^{C} a_{i+} * a_{+i}}{N^2} \tag{18}$$

where a represents the confusion matrix, and N is the number of samples.

TABLE I
RESULTS OF DIFFERENT APPROACHES ON 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 [15]	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
Pointnet++ [16]	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
SPG [52]	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 [42]	18.4	64.6	39.1	0.4	0.0	58.3	58.2	0.0	0.0	0.0	0.0	71.7	9.9	19.3	0.0	0.0	0.0	23.1	5.6	0.0
SqueezeSeg [43]	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	29.0	17.5	24.5
SqueezeSeg2 [59]	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 [11]	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
RandLA-Net [27]	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
BAAF-Net [22]	59.9	90.9	74.4	62.2	23.6	89.8	95.4	48.7	31.8	35.5	46.7	82.7	63.4	67.9	49.5	55.7	53.0	60.8	53.7	52.0
CGGC-Net (ours)	58.4	89.6	73.7	59.0	15.7	91.3	94.5	50.8	35.2	40.8	41.7	83.9	64.9	68.2	58.8	57.6	15.6	62.8	52.5	53.3

B. Implementation Details

The experiments are implemented on deep learning framework PyTorch [63] with Ubuntu18.04. We train for 100 epochs on Geforce RTX 3080 GPU (memory size is 12GB) with a bath size of 6. Besides, we use Adam optimizer and weight decay is set as 0.00001. The initial learning rate is set as 0.004 and we adopt exponential scheduler with $\mathrm{gamma} = 0.95$ to maintain a better learning rate. Moreover, to prevent the overfitting, dropout with $p = 0.5$ is added after the fully connected layer.

C. Analysis of Semantic Segmentation Results

1) Semantic Segmentation on the SemanticKITTI Dataset: Table I gives the quantitative comparisons with different existing models on the SemanticKITTI dataset. It clearly illustrates that our proposed CGGC-Net has surpassed the other approaches by a large margin with an mIoU of 58.4%. In detail, the CGGC-Net demonstrates a remarkable advantage in classifying small instances such as person, bicycle, motorcycle, and bicyclist, achieving 58.8%, 35.2%, 40.8%, and 57.6%, respectively.

In addition, some qualitative results are visualized, as shown in Fig. 5, where the first and third rows represent the ground truth and the second and fourth rows represent our prediction. We could observe that our CGGC-Net is able to classify most objects and still perform well in incomplete places due to occlusions or defections. This could be attributed to the geometric structure encoding, which captures inherent geometric spatial relations within neighborhoods to provide more geometric information for the GSAGCM. Therefore, we could conclude that our CGGC-Net is capable of capturing and exploiting both the local geometric and semantic information of small local regions as well as incomplete places.

Moreover, the visualization of the confusion matrix is also provided in Fig. 6. Kappa reaches 0.847, demonstrating that our proposed CGGC-Net is an excellent classifier for semantic segmentation of large-scale outdoor scene point clouds.

2) Semantic Segmentation on the S3DIS Dataset: To further evaluate the effectiveness of the proposed network in a large-scale indoor scenario, experiments are reported on the S3DIS dataset. In our implementation, the six-fold cross-validation strategy is applied, where every five areas are used as the training set to evaluate the remaining area.

Table II gives the comparable quantitative results with different existing models on the S3DIS dataset. It shows that the OA and mIoU achieve 88.5% and 70.2%, respectively. In particularly, our method achieves the highest accuracy in the floor, beam, window, and sofa. It is worth noting that our proposed CGGC-Net is superior to a CAN [32], even though they can capture long-range dependencies to enhance the representation of point clouds.

Moreover, the detailed semantic segmentation results of 6 areas are also reported in Table III and the associated visualization of the confusion matrix is shown in Fig. 8. Many metrics have illustrated that our CGGC-Net is an ideal classifier for large-scale indoor scene point clouds. Fig. 7 shows the selected examples on the S3DIS dataset. We can observe that our CGGC-Net performs well in all categories, especially in wall, beam, door and chair. Owing to the local geometric multiple transmission and the GSAGCM, the network can capture geometric and semantic relations from long distances. As a result, there are few mistakes at the boundaries of objects.

D. Sensitivity Analysis of Numbers of Neighbors

The number of nearest neighbors directly determines the description of the local geometric structure as well as the extraction of semantic features in the GSAGCM. Thus, a series of comparative experiments are conducted to discuss the influence of the parameter K, which is set to 8, 12, 16, 20, and 24. Fig. 9 indicates the sensitivity analysis of the size of the neighborhood on the classification quality. When K is set to 8, CGGC-Net cannot effectively extract the geometric and semantic features due to the limited neighborhood information.

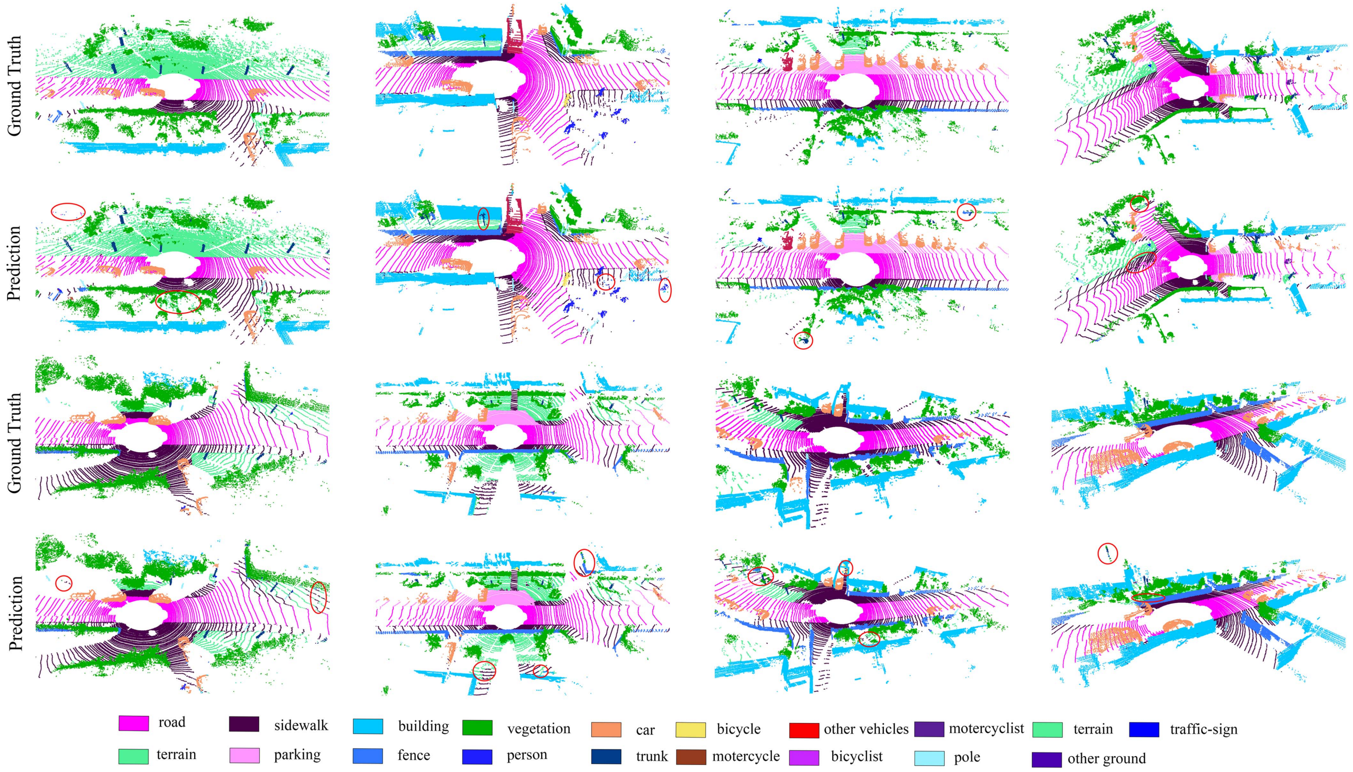


Fig. 5. Examples of semantic segmentation results on SemanticKITTI.

TABLE II
RESULTS OF DIFFERENT APPROACHES ON S3DIS DATASET

Methods	OA	mAcc(%)	mIoU(%)	ceil.	floor	wall	beam	col.	wind.	door	table	chair	sofa	book	board	clut.
PointNet [15]	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 [60]	-	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
SPG [52]	85.5	73.0	62.1	89.9	95.1	76.4	62.8	47.1	55.3	68.4	73.5	69.2	63.2	45.9	8.7	52.9
PointCNN [61]	88.1	75.6	65.4	94.8	97.3	75.8	63.3	51.7	58.4	57.2	71.6	69.1	39.1	61.2	52.2	58.6
PointWeb [62]	87.3	76.2	66.7	93.5	94.2	80.8	52.4	41.3	64.9	68.1	71.4	67.1	50.3	62.7	62.2	58.5
ShellNet [63]	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
RandLA-Net [21]	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
CAN [32]	88.5	79.0	68.3	93.5	96.0	81.5	42.6	46.2	61.0	74.1	67.4	82.7	63.5	59.5	56.9	62.9
CGGC-Net (ours)	88.0	80.3	70.2	93.2	96.6	80.7	64.6	45.9	65.6	67.0	70.6	78.5	65.0	62.7	63.1	59.4

As the size of the neighborhood rises, the classification quality progressively improves, with a fluctuation in mIoU of over 5%. However, when it reaches 24, a small degradation appears possibly due to potential noise and the adhesion of adjacent objects. Fig. 10 also shows a detailed comparison of different numbers of neighbors in each category. We can observe that it has a more prominent impact on some small-scale instances, such as bicycles, trucks and other-vehicles while some large-scale instances such as buildings, roads and vegetation are slightly influenced. Considering the classification performance and computational cost, we set K to 16 as an optimal value in our work.

E. Sensitivity Analysis of the Length of the Stored Tensor $\boldsymbol{\beta_i}$

The parameter S controls the length of each $\boldsymbol{\beta}_i$, determining the number of latest features stored in iteration. Unlike contrastive learning applied in 2-D images, the value of S is

TABLE III
DETAILED SEMANTIC SEGMENTATION RESULTS ON 6 AREAS OF S3DIS DATASET

Testing area	OA	mAcc(%)	mIoU(%)	Kappa	ceil.	Floor	wall	beam	col.	wind.	door	table	chair	sofa	bookcase	board	clut.
Area 1	90.3	84.9	76.3	0.885	96.5	95.7	82.2	63.6	60.1	82.9	85.7	73.5	83.0	66.6	62.4	69.6	70.7
Area 2	85.2	69.5	58.8	0.820	86.1	95.1	80.5	22.1	55.5	67.6	68.8	44.6	69.4	61.1	45.5	20.5	47.8
Area 3	90.7	86.5	76.5	0.889	95.4	98.3	82.3	70.0	39.6	70.4	87.0	73.8	81.2	70.0	68.5	85.7	71.9
Area 4	85.0	75.1	61.2	0.819	94.4	97.7	77.6	51.4	37.0	31.6	60.0	64.0	79.5	52.2	46.9	45.4	58.4
Area 5	87.2	70.7	62.4	0.843	93.5	96.8	79.6	0.0	24.2	62.2	32.6	74.5	86.5	71.1	69.9	67.6	53.3
Area 6	92.1	90.5	81.4	0.908	96.4	97.5	84.9	82.9	66.2	83.2	89.9	77.4	87.8	71.8	75.2	75.3	70.1

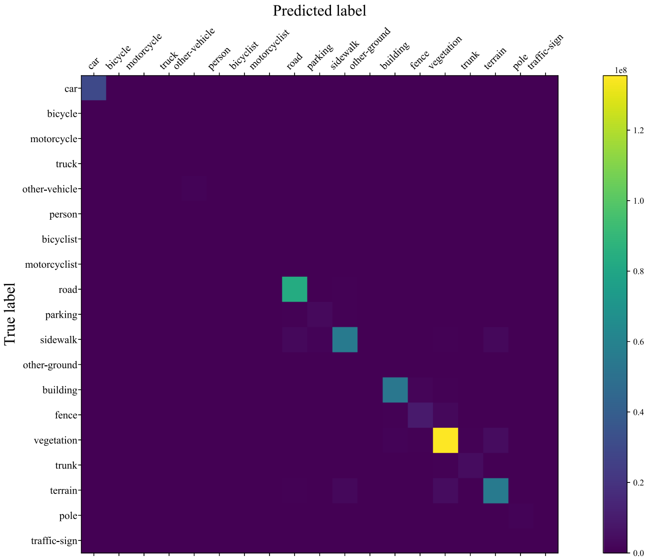


Fig. 6. Visualization of confusion matrix of SemanticKITTI 08 sequence.

TABLE IV
ANALYSIS OF THE LENGTH OF $\boldsymbol{\beta}_i$

S	mIoU(%)	OA(%)
100	58.2	91.1
150	58.6	90.7
200	59.2	91.4
250	58.8	91.1
300	57.7	91.1

recommended to be set higher, as $\boldsymbol{\beta}_i$ can be renewed rapidly with massive point clouds. However, as given in Table IV, when S exceeds 200, there is also a decreasing trend in mIoU. Ultimately, S is set as 200 in our implementation.

F. Sensitivity Analysis of the Margin in Contrastive Loss

The parameter margin Δ is a criterion of similarity measure using category-specific distances. It defines the maximum and minimum distance between input features and the centroid representation of the same class in the feature space. The results are given in Table V. It is worth noting that although the classification results may improve as the separation between categories increases theoretically, the mIoU reaches a peak when Δ is set as 1.5.

TABLE V
ANALYSIS OF THE NUMBER OF SELECTED POINTS EACH ITERATION

Δ	mIoU(%)	OA(%)
1.0	58.4	91.2
1.5	59.2	91.4
2.0	59.0	91.5
2.5	58.6	91.4

TABLE VI
ANALYSIS OF THE NUMBER OF SELECTED POINTS IN EACH ITERATION

N	mIoU(%)	OA(%)
2500	57.9	91.1
5000	59.2	91.4
7500	58.9	91.5
10000	56.9	90.8

G. Sensitivity Analysis of the Number of Selected Points in Each Iteration

As explained in Section III.C, it is definitely impossible to update $\boldsymbol{\beta}_i$ by using the entire point clouds of each iteration (almost 3×10^5) because of the large amount of data. Hence, we select a fixed number of points randomly. Here, we vary the parameter N, and the experimental results are given in Table VI. Consequently, N is set as 5000 in our CGGC-Net.

H. Ablation Studies

In this section, extensive ablation experiments are carried out to further demonstrate the effectiveness of our CGGC-Net.

1) Ablation Study of Detailed Geometric Structure Encoding: In the detailed geometric structure encoding module, a local

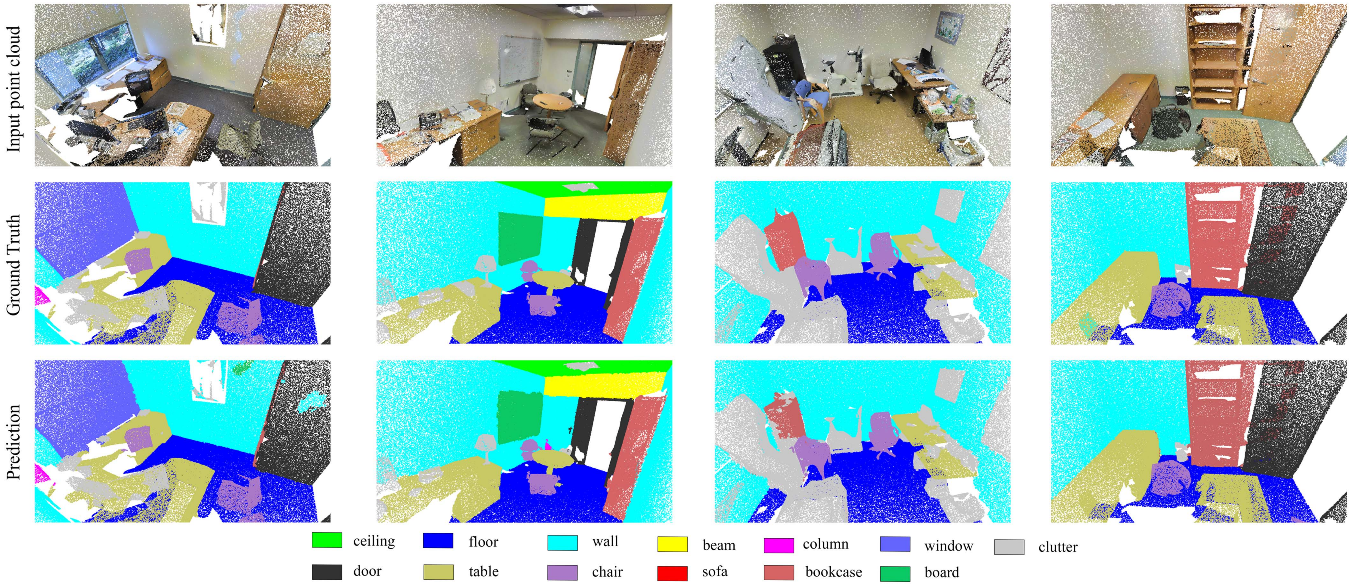
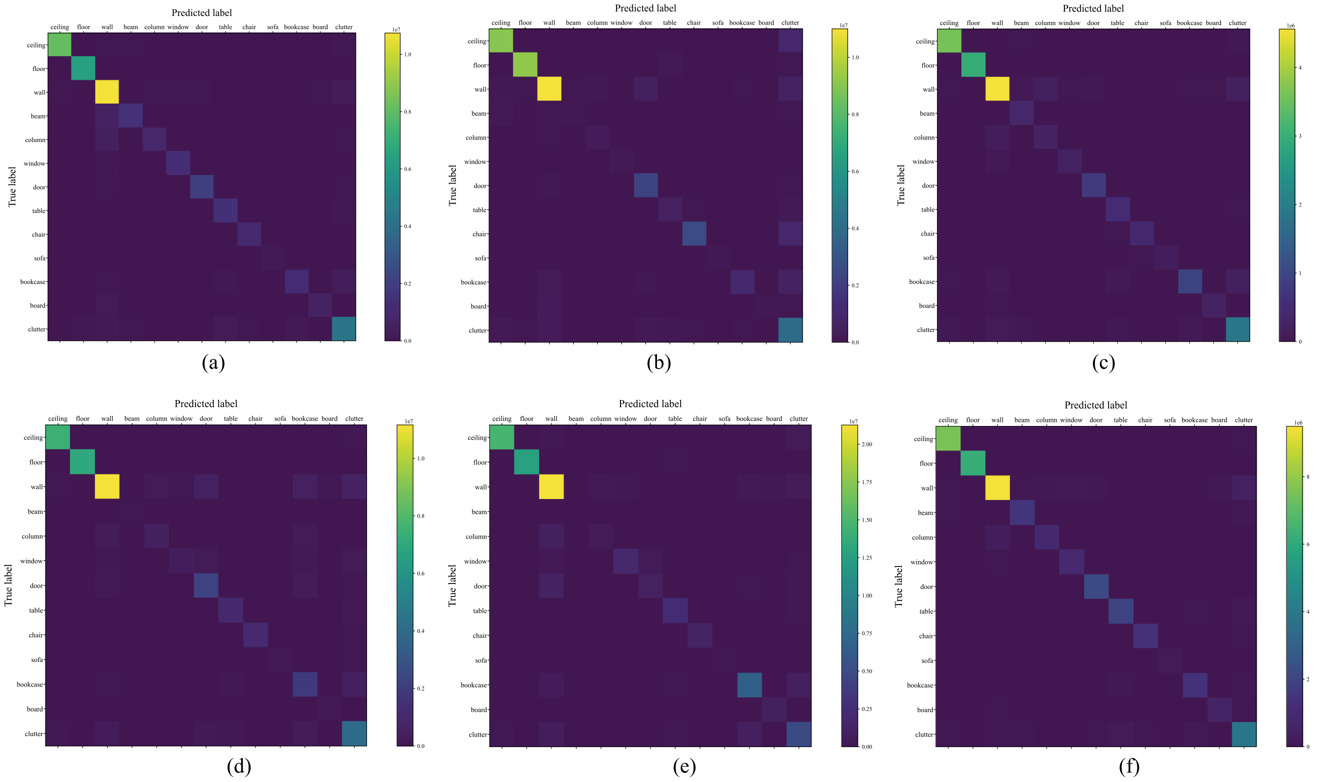


Fig. 7. Semantic segmentation results on S3DIS dataset.



(a) (b) (c) (d) (e) (f)

Fig. 8. Visualization of confusion matrix on 6 areas of S3DIS dataset. (a) Area 1. (b) Area 2. (c) Area 3. (d) Area 4. (e) Area 5. (f) Area 6.

geometric structure descriptor is employed to describe the inherent spatial relations within the neighborhood, and the local geometric structure transmission is designed to further augment geometric information with different sizes of the receptive field. Comparable results with different settings are given in Table VII. The simplification of the geometric structure leads to the lack of local geometric information in the neighborhood, resulting in a decrease of 1.4% in mIoU. In addition, we observe that different forms of transmission play a prominent role in enriching the local geometric structure. In detail, the geometric structure feature transmitted across single and multiple layers can cause an increase of 2.2% and 3.8% in mIoU, respectively.

2) Ablation Study of GSAGCM: Based on PEConv operation, the GSAGCM utilizes both geometric and semantic information

TABLE VII
Ablation Results of Augmented Local Geometric Structure Exploration

Local geometric structure	Transmission	OA(%)	mIoU(%)
×	×	89.8	54.0
√	×	90.1	55.4
√	single	91.1	57.6
√	multiple	91.4	59.2

TABLE VIII
Ablation Results of GSAGCM

GSAGCN	Propagated edge	Edge	Residual	Attentive pooling	OA(%)	mIoU(%)
×	×	×	×	×	89.0	47.2
√	1 layer	$h_\Theta(e_i^j)$	×	√	89.9	53.9
√	2 layers	$h_\Theta(e_i^j)$	√	√	91.4	59.2
√	3 layers	$h_\Theta(e_i^j)$	√	√	90.3	55.8
√	2 layers	$h_\Theta(e_i^j, \boldsymbol{G}_i)$	√	√	89.9	52.8
√	2 layers	$h_\Theta(e_i^j)$	MLP	√	88.1	48.5
√	2 layers	$h_\Theta(e_i^j)$	×	√	89.9	52.8
√	2 layers	$h_\Theta(e_i^j)$	×	max	91.3	57.4

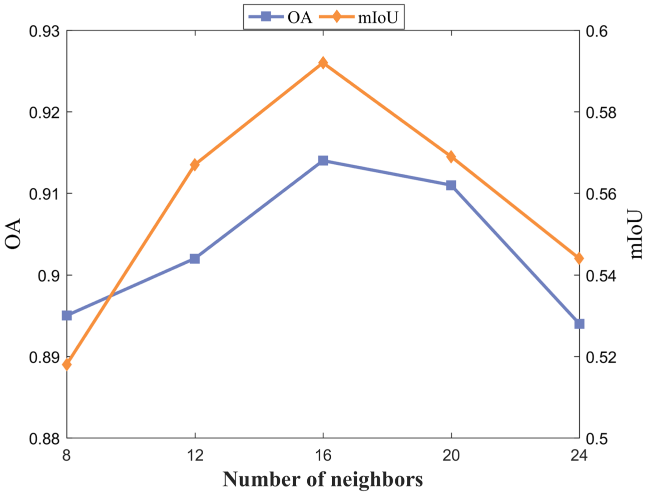


Fig. 9. Analysis of different number of neighbors.

TABLE IX
Ablation Results of Introducing Contrastive Loss

Data	OA (%)	mIoU(%)
Semantic KITTI 08	91.4 (+0.4)	59.2 (+1.2)
S3DIS Area 1	90.3 (+0.5)	76.3 (+0.9)
S3DIS Area 2	85.2 (+0.4)	58.8 (+1.6)
S3DIS Area 3	90.7 (+0.4)	76.5 (+0.8)
S3DIS Area 4	85.0 (+0.6)	61.2 (+2.0)
S3DIS Area 5	87.2 (+0.9)	62.4 (+1.3)
S3DIS Area 6	92.1 (+0.5)	81.4 (+0.7)

to extract the semantic relations between adjacent points, which achieves the aggregation of semantic contexts. To measure the performance of the GSAGCM with comparable settings, we set a series of experiments given in Table VIII. The most distinguished impact is brought by the GSAGCM, leading to 12.0% and 2.4% increases in mIoU and OA, respectively, which largely demonstrates that geometric and semantic information is critical for the semantic segmentation of point clouds. In addition, the number of stacked PEConvs is of great significance, which achieves the highest mIoU (approximately 59.2%). Using a single layer could prevent information propagation from a broader perspective, resulting in a decrease of 5.3% in mIoU. In addition, when the edge attribute feature adopts $h_\Theta(e_i^j, \boldsymbol{G}_i)$, there is a decline of 6.4% and 1.5% in mIoU and OA, respectively. Moreover, different forms of residual connections in the GSAGCM could affect the model to some extent, with the absence of residual connections and the use of MLP as a shortcut reducing the mIoU by 6.4% and 10.7%, respectively. Ultimately, although the max pooling operation retains the most distinguished features within the neighborhood, some useful information is inevitably lost compared with attentive pooling, contributing to a decrease of 1.8% in mIoU.

3) Ablation Study of the Category-Contrastive and Cross-Entropy Guided Optimization Strategy: The category-contrastive and cross-entropy guided optimization strategy adopts additional weighted contrastive loss L_{cont}, which could induce the high-dimensional semantic feature to be more discriminative. The results of introducing contrastive loss on different datasets are given in Table IX. The introduction of additional contrastive loss brings an improvement of 1.2% and 0.4% in mIoU and OA, respectively, in SemanticKITTI.

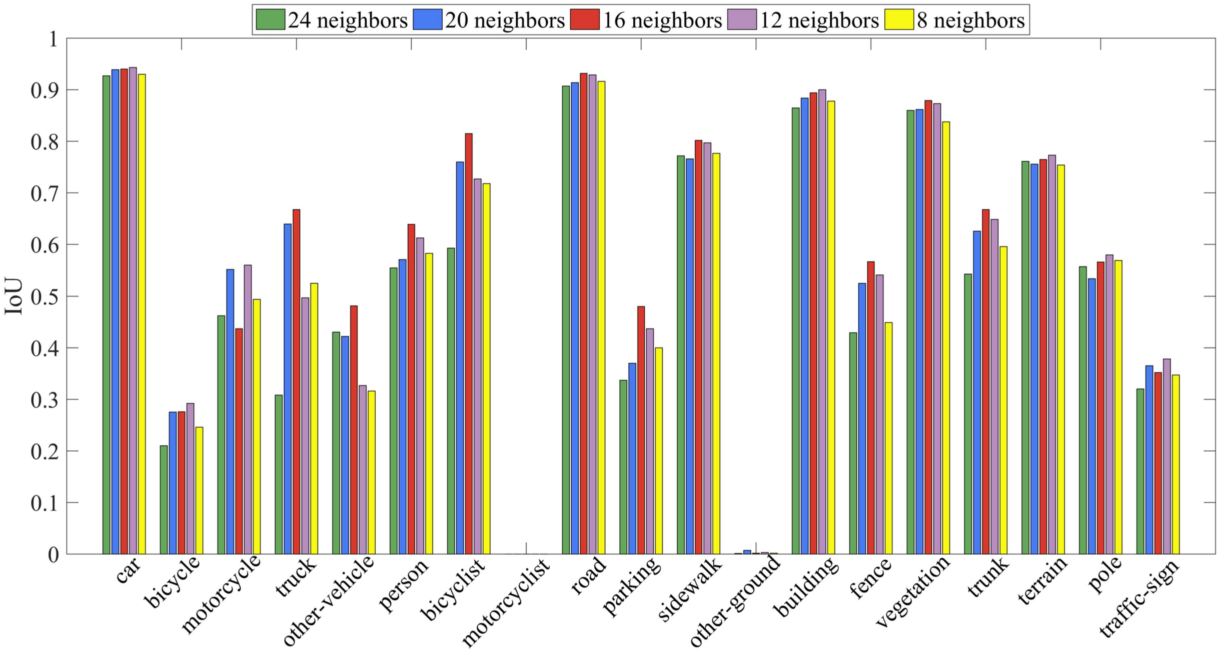


Fig. 10. IoU changes per class with the different number of neighbors.

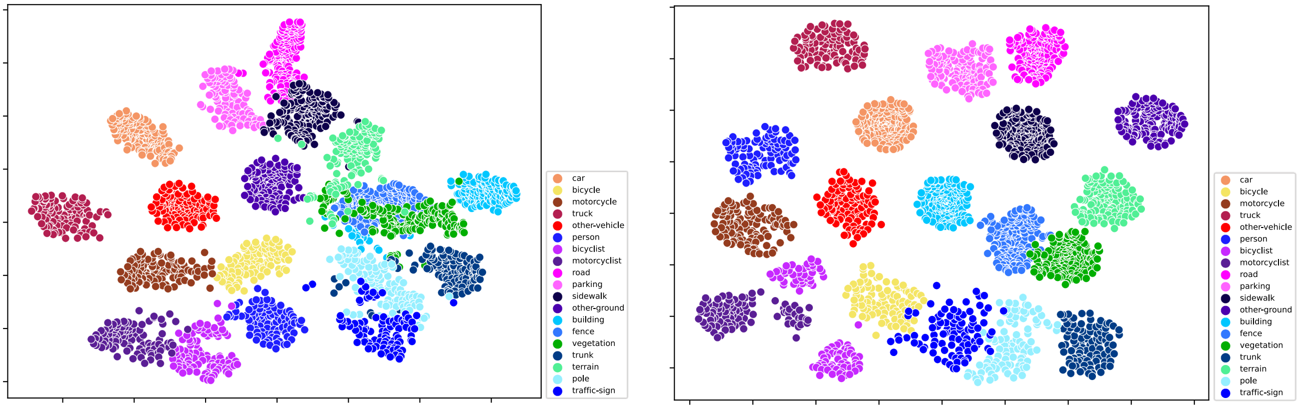


Fig. 11. Visualization analysis of contrastive learning (the right one uses contrastive loss).

Meanwhile, the S3DIS dataset can also lead to an increase ranging from 0.7% to 2.0% in mIoU. We conclude that the utilization of inter-category information in contrastive learning is of great significance in both large-scale indoor and outdoor scenarios.

I. Visualization in Latent Feature Space of Contrastive Learning

In this section, to further illustrate the clustering results of contrastive learning more explicitly and vividly, comparative visualizations of whether contrastive loss is introduced during iteration are performed using t-distributed stochastic neighbor embedding techniques. As shown in Fig. 11, after applying contrastive loss, points belonging to the same category tend to be gathered together in latent feature space, while those belonging to different categories are forced to stay apart. In brief, we can conclude that feature representations are prompted to be more discriminative in the process of minimizing the category-specific distances, which would be beneficial to the determination of semantics and enhance the accuracy of multitask classification results.

V. Conclusion

With the rapid development of 3-D scanners, the semantic segmentation of LiDAR point clouds is the foundation for spatial intelligent perception and has been a trending topic in recent years. Hence, in this article, we develop a contrastive-category guided learning graph convolutional neural network for the semantic segmentation of LiDAR point clouds. First, the detailed local geometric structures are designed to extract the inherent geometric information and combine it from different receptive fields. Then, a GSAGCM utilizes the multistacked PEConvs and attention pooling to achieve the extraction and transmission of neighboring semantic relationship information, which aggregates newer semantic features per point in parallel. Finally, by introducing contrastive loss, the semantic features generated from the previous encoder-decoder architecture could become more discriminative, benefiting the transformation to the pointwise classification score in the subsequent classification layer. Experiments on the SemanticKITTI and S3DIS dataset have shown that our CGGC-Net performs well in both large-scale outdoor and indoor scenarios and is capable of classifying small and even incomplete instances.

Nevertheless, the semantic segmentation of large-scale point clouds in fully-supervised tasks requires time-consuming and laborious dense annotation. Therefore, in the future, we will explore weakly supervised point cloud semantic segmentation.

References

[1] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3d object detection methods for autonomous driving applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3782–3795, Oct. 2019.

[2] S. A. Bello, S. Yu, C. Wang, J. M. Adam, and J. Li, "Deep learning on 3D point clouds," *Remote Sens.*, vol. 12, no. 11, 2020, Art. no. 1729.

[3] K. Lai and D. Fox, "Object recognition in 3D point clouds using web data and domain adaptation," *Int. J. Robot. Res.*, vol. 29, no. 8, pp. 1019–1037, 2010.

[4] I. Colomina and P. Molina, "Unmanned aerial systems for photogrammetry and remote sensing: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 92, pp. 79–97, 2014.

[5] M. Jaboyedoff et al., "Use of LIDAR in landslide investigations: A review," *Natural Hazards*, vol. 61, no. 1, pp. 5–28, 2012.

[6] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-D mapping with an RGB-D camera," *IEEE Trans. Robot.*, vol. 30, no. 1, pp. 177–187, Feb. 2014.

[7] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source slam system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[8] L. Luo, S. Y. Cao, B. Han, H. L. Shen, and J. Li, "Bvmatch: Lidar-based place recognition using bird's-eye view images," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 6076–6083, Jul. 2021.

[9] S. Gasperini, M. A. N. Mahani, A. Marcos-Ramiro, N. Navab, and F. Tombari, "Panoster: End-to-end panoptic segmentation of lidar point clouds," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 3216–3223, Apr. 2021.

[10] H. Fang, C. Pan, and H. Huang, "Structure-aware indoor scene reconstruction via two levels of abstraction," *ISPRS J. Photogramm. Remote Sens.*, vol. 178, pp. 155–170, 2021.

[11] J. Zhang, X. Lin, and X. Ning, "SVM-based classification of segmented airborne LiDAR point clouds in urban areas," *Remote Sens.*, vol. 5, no. 8, pp. 3749–3775, 2013.

[12] J. Niemeyer, F. Rottensteiner, and U. Soergel, "Contextual classification of lidar data and building object detection in urban areas," *ISPRS J. Photogramm. Remote Sens.*, vol. 87, pp. 152–165, 2014.

[13] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 922–928.

[14] T. Le and Y. Duan, "Pointgrid: A deep network for 3d shape understanding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9204–9214.

[15] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.

[16] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5099–5108.

[17] M. Welling and T. N. Kipf, "Semi-supervised classification with graph convolutional networks," in 2016, *arXiv:1609.02907*.

[18] L. Wang, Y. Huang, Y. Hou, S. Zhang, and J. Shan, "Graph attention convolution for point cloud semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10296–10305.

[19] L. Jiang, H. Zhao, S. Liu, X. Shen, C. W. Fu, and J. Jia, "Hierarchical point-edge interaction network for point cloud semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 10433–10441.

[20] H. Zhou, Y. Feng, M. Fang, M. Wei, J. Qin, and T. Lu, "Adaptive graph convolution for point cloud analysis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 4965–4974.

[21] W. Li, F. D. Wang, and G. S. Xia, "A geometry-attentional network for ALS point cloud classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 164, pp. 26–40, 2020.

[22] S. Qiu, S. Anwar, and N. Barnes, "Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1757–1767.

[23] Z. Kang and J. Yang, "A probabilistic graphical model for the classification of mobile LiDAR point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 108–123, 2018.

[24] H. Thomas, C. R. Qi, J. E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, "Kpconv: Flexible and deformable convolution for point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6411–6420.

[25] J. Mao, X. Wang, and H. Li, "Interpolated convolutional networks for 3d point cloud understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1578–1587.

[26] A. Vaswani et al., "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 30, pp. 5998–6008, 2017.

[27] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, and Z. Wang, "Randla-net: Efficient semantic segmentation of large-scale point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11108–11117.

[28] D. Li et al., "AGFP-Net: Attentive geometric feature pyramid network for land cover classification using airborne multispectral LiDAR data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 108, 2022, Art. no. 102723.

[29] H. Zhao, L. Jiang, J. Jia, P. H. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 16259–16268.

[30] C. Park, Y. Jeong, M. Cho, and J. Park, "Fast point transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16949–16958.

[31] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, 2019.

[32] C. Liu et al., "Context-aware network for semantic segmentation toward large-scale point clouds in urban environments," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Jun. 2022, Art. no. 5703915.

[33] F. Shuang, P. Li, Y. Li, Z. Zhang, and X. Li, "Msida-Net: Point cloud semantic segmentation via multi-spatial information and dual adaptive blocks," *Remote Sens.*, vol. 14, no. 9, 2022, Art. no. 2187.

[34] J. Behley et al., "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9297–9307.

[35] I. Armeni et al., " 3d semantic parsing of large-scale indoor spaces," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1534–1543.

[36] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[37] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[38] F. J. Lawin, M. Danelljan, P. Tosteberg, G. Bhat, F. S. Khan, and M. Felsberg, "Deep projective 3D semantic segmentation," in *Proc. Int. Conf. Comput. Anal. Images Patterns*, 2017, pp. 95–107.

[39] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1907–1915.

[40] A. Boulch, J. Guerry, B. L. Saux, and N. Audebert, "Snapnet: 3D point cloud semantic labeling with 2D deep segmentation networks," *Comput. Graph.*, vol. 71, pp. 189–198, 2018.

[41] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3D point clouds: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 12, pp. 4338–4364, Dec. 2021.

[42] H. Su et al., "Splatnet: Sparse lattice networks for point cloud processing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2530–2539.

[43] B. Wu, A. Wan, X. Yue, and K. Keutzer, "Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3D lidar point cloud," in *Proc. Int. Conf. Robot. Automat.*, 2018, pp. 1887–1893.

[44] Y. Zhang et al., "Polarnet: An improved grid representation for online lidar point clouds semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9601–9610.

[45] D. Rethage, J. Wald, J. Sturm, N. Navab, and F. Tombari, "Fully-convolutional point networks for large-scale point clouds," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 596–611.

[46] B. Graham, M. Engelcke, and L. Van Der Maaten, "3d semantic segmentation with submanifold sparse convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9224–9232.

[47] C. Choy, J. Gwak, and S. Savarese, "4d spatio-temporal convnets: Minkowski convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3075–3084.

[48] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009.

[49] Z. Ravichandran, L. Peng, N. Hughes, J. D. Griffith, and L. Carlone, "Hierarchical representations and explicit memory: Learning effective navigation policies on 3D scene graphs using graph neural networks," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 9272–9279.

[50] A. Bessadok, M. A. Mahjoub, and I. Rekik, "Graph neural networks in network neuroscience," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 5833–5848, May 2023, doi: 10.1109/TPAMI.2022.3209686.

[51] G. Bouritsas, F. Frasca, S. Zafeiriou, and M. M. Bronstein, "Improving graph neural network expressivity via subgraph isomorphism counting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 657–668, Jan. 2023.

[52] L. Landrieu and M. Simonovsky, "Large-scale point cloud semantic segmentation with superpoint graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pp. 4558–4567, 2018.

[53] K. J. Joseph, S. Khan, F. S. Khan, and V. N. Balasubramanian, "Towards open world object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 5830–5840.

[54] J. Yang et al., "A laboratory open-set Martian rock classification method based on spectral signatures," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Jun. 2022, Art. no. 4601815.

[55] F. Engelmann, T. Kontogianni, and B. Leibe, "Dilated point convolutions: On the receptive field size of point convolutions on 3D point clouds," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 9463–9469.

[56] Y. Mao et al., "Beyond single receptive field: A receptive field fusion-and-stratification network for airborne laser scanning point cloud classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 188, pp. 45–61, 2022.

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[58] G. Li, M. Muller, A. Thabet, and B. Ghanem, "Deepgcns: Can gcns go as deep as cnns?," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9267–9276.

[59] B. Wu, X. Zhou, S. Zhao, X. Yue, and K. Keutzer, "Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud," in *Proc. Int. Conf. Robot. Automat.*, 2019, pp. 4376–4382.

[60] Q. Huang, W. Wang, and U. Neumann, "Recurrent slice networks for 3d segmentation of point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2626–2635.

[61] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "Pointcnn: Convolution on x-transformed points," *Adv. Neural Inf. Process. Syst.*, vol. 31, pp. 828–838, 2018.

[62] H. Zhao, L. Jiang, C. W. Fu, and J. Jia, "Pointweb: Enhancing local neighborhood features for point cloud processing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5565–5573.

[63] Z. Zhang, B. S. Hua, and S. K. Yeung, "Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1607–1616.

[64] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," *Adv. Neural Inf. Process. Syst.*, vol. 32, pp. 8026–8037, 2019.

Xuzhe Wang is currently working toward the Bachelor's degree with the College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao, China.

His research interests include semantic segmentation of point cloud and indoor modeling.

Juntao Yang received the Ph.D. degree in photogrammetry and remote sensing from the China University of Geosciences, Beijing, China, in 2021.

He was a visiting Ph.D. student with Gottfried Wilhelm Leibniz Universität Hannover, Germany, from 2019 to 2020. He is currently an Associate Professor with the College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao, China. He has authored more than 20 referred journal and conference publications. His research interests include scene understanding of point clouds, planetary structure mapping and analysis.

Zhizhong Kang received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2004.

He was a Postdoctoral Researcher with Delft University of Technology, Delft, The Netherlands, from 2006 to 2008. He is currently a Full Professor and the Vice Dean with the School of Land Science and Technology, China University of Geosciences, Beijing, China. He has authored more than 100 referred journal and conference publications. His research interests include digital photogrammetry, LiDAR data processing, indoor modeling and navigation, and planetary remote sensing.

Dr. Kang was the Chair of ISPRS WG IV/5: Indoor/Outdoor seamless modeling, LBS, mobility. He was the recipient of 2015 ERDAS Award for Best Scientific Paper in Remote Sensing by American Society for Photogrammetry and Remote Sensing and ISPRS President's Honorary Citations 2022.

Junjian Du is currently working toward the Bachelor's degree with the College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao, China.

His research interests include pedestrian trajectory prediction.

Zhaotong Tao is currently working toward the Bachelor's degree with the College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao, China.

Her research interests include semantic segmentation of point clouds.

Dan Qiao is currently working toward the Bachelor's degree with the College of Geodesy and Geomatics, Shandong University of Science and Technology, Qingdao, China.

Her research interests include remote data processing and LAI retrieval.