# A 3D Semantic Segmentation Method for Large-Scale Point Cloud on Deep Learning

Sihan Liu, Wenyu Zhang*, Yujun Zhang, Zhijian Wang, Dongxiang Gao

*Abstract*—**Point cloud data's ability to preserve precise geometric details makes point cloud semantic segmentation crucial for 3D visual perception. Within a perception system, the real-time performance of the model holds pivotal significance, particularly concerning its ability to conduct semantic segmentation on large-scale point cloud data. RanSeNet, a lightweight neural architecture based on attention mechanisms, directly operates on each individual point in the point cloud data, eliminating the need for preliminary processing steps. The experiment shows that RanSeNet achieves fast processing, high segmentation efficiency, and handles millions of points simultaneously. Compared to existing results, the proposed method achieves 88.6% Overall Accuracy (OA) and 64.27% Mean Intersection over Union (MIoU) in area 5 of the S3DIS dataset, which is a challenging large-scale semantic scene segmentation task.**

*Index Terms*—**3D point cloud, semantic segmentation, random sampling, Attention mechanism**

## I. Introduction

EFFICIENT semantic segmentation method of large-scale 3D point cloud is a fundamental and indispensable attribute of real-time intelligent systems, such as indoor AI robots and augmented reality. The employment of generalized point clouds facilitates the achievement of autonomous robot navigation and environmental perception, consequently elevating the overall cognitive prowess of robotic systems. However, as the complexity of network structures escalates, computational efficiency diminishes, thereby impinging upon the practical applicability of these methods. Compared with the 2D image segmentation task, the 3D point cloud data acquired through depth sensors exhibits inherent traits of irregularity, with dense clustering in close proximity juxtaposed with sparse dispersion in distant regions, resulting in an unstructured nature. While deep convolutional neural networks have showcased remarkable prowess in the realm of two-dimensional computer vision tasks, their direct transposition to the domain of unstructured point cloud data presents inherent challenges.

At the same time, several methods exist for semantic segmentation of point clouds, such as the random sampling consensus method (RANSAC), which exhibits higher robustness and efficiency for plane, sphere, cylinder, etc.; the Euclidean segmentation algorithm based on clustering; the Region growth segmentation algorithm; segmentation based on normal differentiation; and Supervoxel-based segmentation, among others [1]. However, due to the inherent characteristics of the aforementioned methods, when dealing with increasing amounts of point cloud data, there are challenges such as high computing power requirements, long processing times, lower precision, vulnerability to noise interference, and difficulties in handling large-scale indoor and outdoor data. As a result, point cloud semantic segmentation based on deep learning has emerged as a fervently discussed subject in contemporary research, given its remarkable high-level semantic comprehension capabilities.

The projection-based deep learning segmentation method involves projecting point clouds onto 2D images and using traditional 2D CNN for segmentation. However, this approach requires extensive preprocessing steps and may lead to the loss of crucial spatial information [2]. Similarly, voxel-based deep learning segmentation method uses 3D convolution, which is not only slow in calculation, but also produces a large number of invalid calculations [3]. To address these issues, a direct point-to-point processing method is needed, which can bypass the need for pre-processing steps [4].

PointNet, proposed by Charles [5-6] et al., is a deep learning-based model that directly processes point clouds for point cloud segmentation. This method utilizes shared multi-layer perceptrons (MLPs) to learn the features of individual points, making it computationally efficient. However, it has limitations in capturing broader context information for each point. To address this limitation, the authors then proposed PointNet++, which enhances the model's ability to capture local context information at different scales. In order to learn richer local structures, many specialized neural modules have been introduced by relevant scholars [7-8]. While these methods have achieved favorable semantic segmentation results, most of them are limited to tiny scale 3D point clouds (e.g., 4000 points or 1×1 meter blocks for indoor scenes) and cannot be directly extended to larger point clouds (e.g., millions of points and 200×200 meter blocks for larger scale scenario) [9]. There are three reasons for these limitation. (1) Point sampling methods commonly used in these networks are computationally expensive and have low memory efficiency. (2) Most existing local feature learning methods rely on computationally expensive neighborhood search or graph construction, making it challenging to handle a large number of points. (3) For large-scale point clouds, which are usually composed of hundreds of objects, existing local feature learners may struggle to capture the complex

Manuscript received May 20, 2023; revised August 30, 2023.

The work was supported by the Natural Science Foundation of China (No.62272093) and by Liaoning Key Laboratory of the Internet of Things Applicaiton Technology on Intelligent Construction.

Sihan Liu is a graduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (e-mail: 1350907346@qq.com).

Wenyu Zhang is a Professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (Corresponding author, e-mail: zhangwenyu8518@126.com).

Yujun Zhang is a Professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (e-mail: 1997zyj@163.com).

Zhijian Wang is a graduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (e-mail: ew98@foxmail.com).

Dongxiang Gao is a graduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China. (e-mail: 1074501593@qq.com).

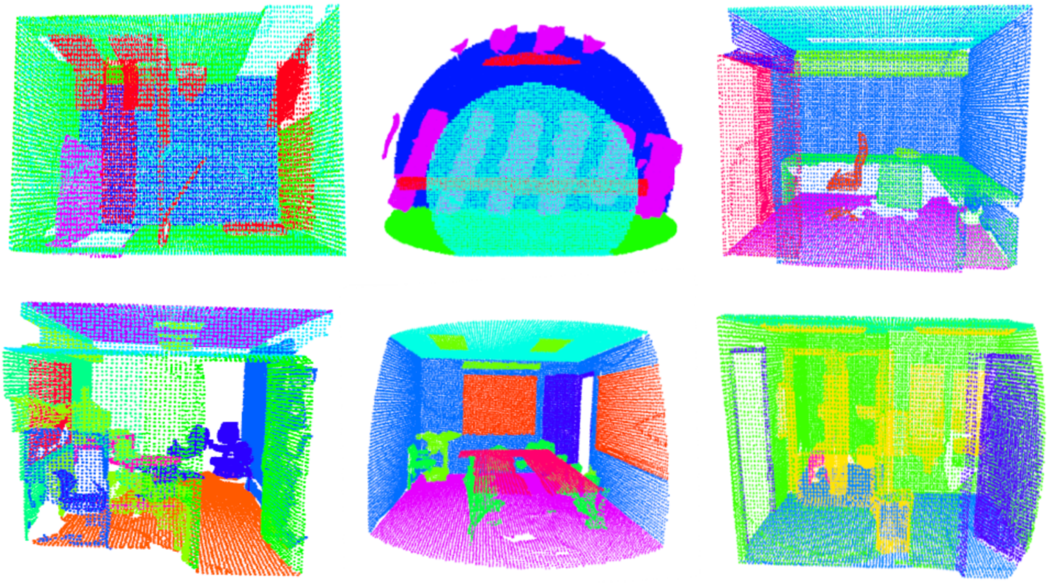

Fig. 1: Semantic segmentation results of RanSeNet on S3DIS

structure efficiently, or their capture efficiency is low due to the limited size of their receptive field.

In recent years, some methods based on projection and voxel techniques have achieved good segmentation accuracy for large-scale scenic point clouds. However, the heavy computation involved in pre-processing and voxelization steps makes these methods unsuitable for real-time applications. On the other hand, some methods based on deep learning have shown promising results, but they are more effective when dealing with small data sets. To address the mentioned issues, Qingyong Hu [10] proposed the RandLA-Net method, which is suitable for semantic segmentation of large-scale point cloud data. The fundamental concept underpinning this approach revolves around the utilization of random point sampling, thereby augmenting data processing velocity and mitigating the loss of information. Nevertheless, the core emphasis of the RandLA-Net architecture predominantly centers upon the refinement of spatial encodings, inadvertently overlooking the interplay amongst channels. This particular constraint gives rise to inadequate feature assimilation within point clouds, consequently exerting a potential dampening effect on the precision of classification and segmentation outcomes.

This paper introduces an enhanced point cloud semantic segmentation model called RanSeNet, which is evaluated on the S3DIS [11], a large indoor scenic point cloud dataset. The objective of this study is to develop robots capable of handling complex tasks similar to humans. The proposed model incorporates an improved Transformer structure, employing a channelwise statistical attention mechanism along with the Gaussian Error Linear Unit (GELU) activation function. The experimental results demonstrate that our model achieves superior segmentation accuracy and faster processing speed compared to other existing methods. For more detailed segmentation results, please refer to Fig. 1.

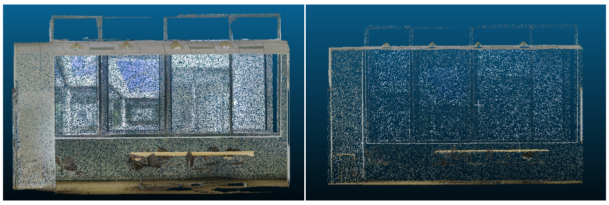

Fig. 2: Grid before and after sampling

## II. Semantic Segmentation Module Overview

### A. Problem Definition

*1) Sampling Procedure:* Dealing with a large number of point clouds at once is unrealistic, given that millions of points cannot be directly fed into the model. To address this, RanSeNet employs raster downsampling of point clouds to reduce the number of points while preserving the spatial structure of the point clouds, as depicted in Fig. 2. This approach enables the model to handle large-scale point cloud data effectively [12].

While raster downsampling ensures the preservation of the geometric structure of point clouds to some extent, it cannot guarantee the consistency of output points. Additionally, even after raster downsampling, the number of points may still be too large for neural networks to handle efficiently. To address this, the RanSeNet model employs random sampling (RS) and continues to downsample points further, reducing the point cloud size to a manageable level for the neural network. This combination of techniques helps optimize the model's performance on large-scale point cloud data. As shown in Fig. 3, a random value $P$ was generated for each point in the subpoint cloud $P_j$, the smallest point $PJ$ was selected,

TABLE I: Efficiency comparison of sampling methods

| **Method**<br>**Index** | FPS | IDIS | RS | GS | CRS | PGS |
|---|---|---|---|---|---|---|
| **efficiency**($10^6$) | 200s | 10s | 0.004s | 12000s | | |
| **effect** | Applies only to small scale point clouds | Sensitive to outliers | | Generate additional calculations | 3000GB memory usage | Survey space $C_{10^6}^{10^5}$ |
| **complexity** | $O(N^2)$ | $O(N)$ | $O(1)$ | | | |

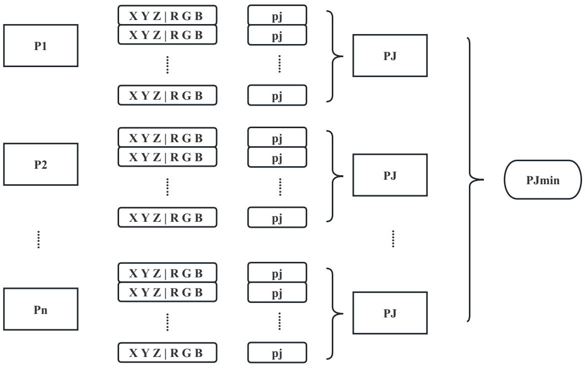

Fig. 3: Random sampling detail schematic

and the smallest point $PJ_{min}$ was chosen from the point $PJ$ optioned by each subpoint cloud. The k-dimensional tree algorithm was carried out to find neighbor points, and the neighbor points $N$ found was used as the sampling point to be input into the model for this round of training.

To achieve semantic segmentation of point cloud, the choice of sampling method is crucial. Existing point sampling approaches can be roughly classifies into heuristic and based on learning approachs. However, there is still no standard sampling strategy that is suitable for large scale point clouds. Subsequently, we analyze and compare the relative merits and complexities of different sampling methods as shown in Table I. The methods included are: FPS (farthest point sampling), IDIS (inverse density importance sampling), GS (Generator-based Sampling), CRS (Continuous Relaxation based Sampling), and PGS (Policy Gradient based Sampling). Regardless of the size of the input point cloud, RS (random sampling) exhibits the highest computing efficiency [13]. When $10^6$ points are processed, only 0.004s is needed, and its computational complexity is $O(1)$.

*2) Spatial Location Coding —Extract Feature:* If the location information after point aggregation only includes the location of the point itself and the location of the central point, the network may exhibit limitations in comprehending the local geometric model [14]. Therefore, the method incorporates relative position information and Euclidean distance, which has been verified to achieve the optimal effect. The process of this module is illustrated in Fig. 4. By incorporating additional geometric information, the model gains a better understanding of the local geometric characteristics, leading to improved performance in semantic segmentation.

Given a child point cloud $P$, after random sampling, we get $p$, and sum total $N$. Each point $p_i (1 \leq i \leq N)$ in $p$ carries a point feature $f_i$ and its position information $(x, y, z)$. For the $i$th point $i^{th}$ in $p$, firstly, a simple nearest neighbor (KNN) algorithm is used to collect $K$ adjacent points to improve efficiency.

For each $K$ nearest point $\{p_i^1 \cdots p_i^k \cdots p_i^k\}$ of the central point $p_i$, the position coding $r_i^k$ is shown in Formula (1):

$$r_i^k = MLP\left(p_i \oplus p_i^k \oplus (p_i - p_i^k) \oplus \|p_i - p_i^k\|\right) \quad (1)$$

where $p_i$ and $p_i^k$ are the position of the points, $\oplus$ is the join operation, and $\| \bullet \|$ calculates the Euclidean distance between the adjacent points and the center point. For each adjacent point $p_i^k$, the relative point position $r_i^k$ of the code is connected with its corresponding point feature $f_i^k$ to obtain an augmented feature vector $\hat{f}_i^k$.

The output of this unit is a new set of adjacent feature $\hat{F}_i = \{\hat{f}_i^1 \cdots \hat{f}_i^k \cdots \hat{f}_i^K\}$ that encodes the local geometry of the central point.

The above steps ensure that the corresponding point features always have knowledge of their relative spatial positions. This allows the module to explicitly observe local geometric patterns, ultimately enabling the entire network to effectively learn complex local structures.

*3) Attention Pooling —Weighting the Most Important Adjacent Features:* Given a local feature set: $\hat{F}_i = \{\hat{f}_i^1 \cdots \hat{f}_i^k \cdots \hat{f}_i^K\}$, a shared function $g()$ is designed to learn the unique attention score for each feature [15]. $g()$ consists of a shared MLP and softmax. Its formal definition is as follows:

$$s_i^k = g(f_i^k, W) \quad (2)$$

where $W$ is a learnable weight of a shared MLP. The learned attention score can be viewed as a soft mask layer that automatically selects important features. The weighted sum of these features is calculated as follows:

$$\tilde{f}_j = \sum_{k=1}^{K} (\hat{f}_j^k \cdot s_j^k) \quad (3)$$

This process allows the model to focus on significant and relevant information, enhancing the effectiveness of feature learning and improving the overall performance of the network in semantic segmentation tasks.

Given the input point cloud $p$, for the $j$ th point $p_j$, our local point and attention pool unit learn to aggregate the geometric patterns and features of its $K$ closest points, and finally generate an information feature vector $\tilde{f}_j$, as shown in Fig. 5 below.

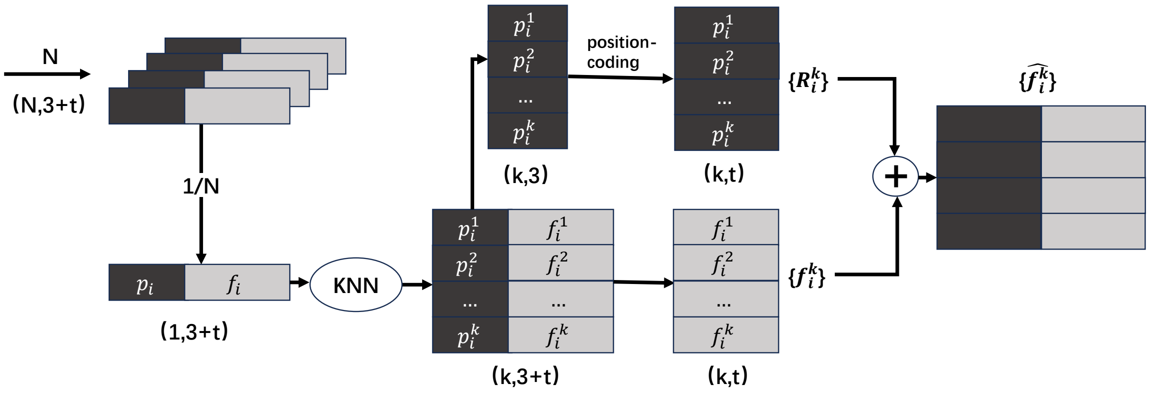

Fig. 4: Local feature aggregation module

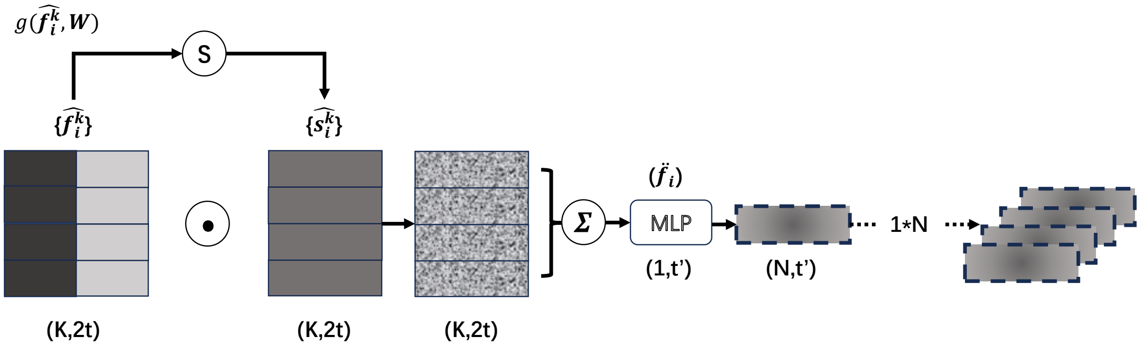

Fig. 5: Attention pooling module

*4) Attention Mechanism (SEPC) —Model the Interdependencies between Feature Channels explicitly:* Different from image data, point cloud data has the characteristics of unstructure and disorder [16], and its characteristics can be expressed as $F \in RB \times N \times 1 \times C$ in neural network, where $B$ represents Batch Size and $C$ is the number of feature channels. In Fig. 6, r stands for the reduction rate, which represents the number of channels reduced at the first fully connected layer. The reduction rate helps to control the dimensionality of the features and can be adjusted to balance the model's capacity and computational efficiency.

Feature layer $U \in R^{N\times1\times C}$ is compressed by global average pooling, the process is as follows:

$$Z_c = F_{sq}(u_c) = \frac{1}{h*w}\sum_{i=1}^{h}\sum_{j=1}^{w} u_c(i,j) \tag{4}$$

where $h*w$ represents the spatial dimensions of the intermediate features.

In the proposed model, each channel learns the activation of specific samples through an optional gate mechanism based on channel dependence. This mechanism allows the model to learn how to use global information and selectively emphasize important information features while suppressing less relevant ones. The activation function is sigmoid and ReLU function is embedded to help network training and limit the complexity of the model. These design choices help in enhancing the model's ability to capture meaningful patterns and improve its overall performance in semantic segmentation tasks.

Through two fully connected layers (FC), the number of $W_1$ neurons in the first fully connected layer is less for dimension reduction, and the number of $W_2$ neurons in the second fully connected layer is the same as that in the input feature layer for dimension enhancement. The weight is fixed between 0 and 1 by Sigmoid, that is, the weight of each channel in the input feature layer (between 0 and 1) is obtained.

$$s = F_{ex}(z_c, W) = \delta(f(z_c, W)) = \delta(W_2\vartheta(W_1 Z_C)) \tag{5}$$

where $\delta$ and $\vartheta$ are the representation of Sigmoid function.

SEPC automatically acquires the importance degree of each channel through learning, and then promotes useful features according to this importance degree and inhibits the features that are not useful for the current task, so as to improve the network expression ability.

*5) Feature Aggregation Module (FAM) —Increment Acceptance Field:* This module connects the above two modules as shown in Fig. 7 to increase the range of the acceptance domain.

Following the initial positioning and pooling operation, each 3D point depicted in the figure perceives $K$ adjacent points, enabling it to gather information from up to $K^2$ adjacent points. The two-hop neighborhood after the second positioning [17]. This is an inexpensive method to extend the receptive domain and the effective neighborhood by feature propagation. This feature propagation approach enhances the model's ability to handle complex and large-scale point cloud data while maintaining computational efficiency.

Given the substantial downsampling of large point clouds,

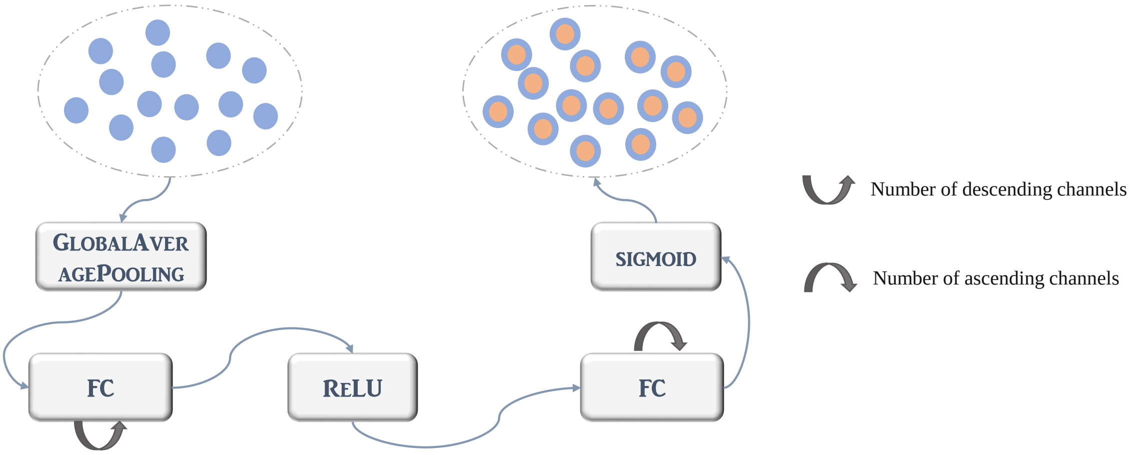

Fig. 6: SEPC Module

it becomes imperative to significantly expand the acceptance domain of each point. This ensures that even if certain points are removed, the geometric intricacies of the input point cloud are more likely to be retained. We use multiple attention pool units with skip connections as extended residual blocks.

The residual block, comprising SEPC and FAM, effectively enhances the network's capacity for characterization while maintaining the integrity of the point cloud data structure, as depicted in Fig. 8. The channel weights are dynamically adjusted within local features to enhance the network's representation in the given task. Within point cloud data, this attention mechanism can be conceptualized as operating on distinctive channels at each point, thereby accentuating task-relevant information and mitigating the presence of extraneous details.

Utilizing the GELU function as the activating layer for the outputs of the residual module FAM offers a pathway for refining the acquisition of nonlinear features par excellence. This strategic selection finds its rationale in GELU's adept utilization of the probabilistic statistics inherent to input data, thereby upholding the integrity of input information while seamlessly accommodating the inherent stochastic regularities within the activation function itself. The mathematical formulation of the GELU function is expressed as follows:

$$GELU(X) = X * P(X \leq x) = X * \Psi(x) \tag{6}$$

where $\Psi(x)$ refers to the cumulative function of the Gaussian normal distribution of $x$.

In RanSeNet, two sets of local pools are deployed along with an attention pool to create standard residual blocks, striking a favorable balance between efficiency and effectiveness. This module takes into account the adjacent geometry and expands the receptive field, effectively preserving the complex local structure within the point cloud data.

### B. Network Structure

As depicted in Fig. 9, the extended residual block structure is denoted as FAM. RS signifies random sampling, NL represents near point interpolation (used for image upsampling), SEPC refers to the attention mechanism module. $C_{in}$ represents the input channel of each module, $C_{out}$ represents the output channel of each module, and Concat denotes the concatenation operation. The model in RanSeNet fully integrates feature information from different dimensions, which involves four rounds of downsampling and four rounds of upsampling.

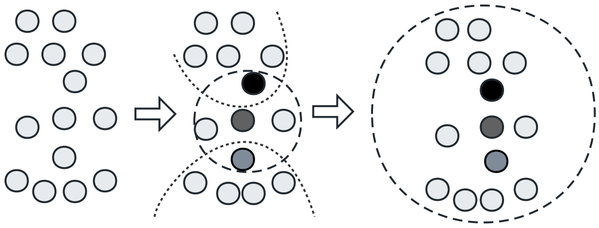

Fig. 7: FAM: Feature Aggregation Module

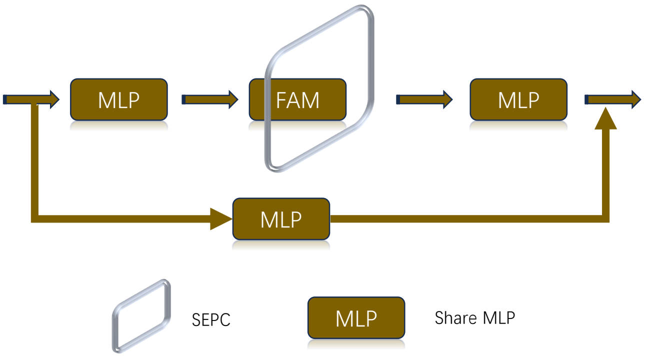

Fig. 8: Overall architecture of RanSeNet's feature extraction

## III. Experimental Design and Implementation

### A. Design and Implementation

In this chapter, we evaluate the overall efficiency of RanSeNet on large-scale semantic segmentation point clouds in real-world scenarios. To assess the performance of RanSeNet, we conducted evaluations on the S3DIS dataset, which is a widely-used large-scale indoor point cloud dataset. For evaluation, we use standard metrics such as Mean Intersection over Union (mIoU) and overall accuracy (OA) of

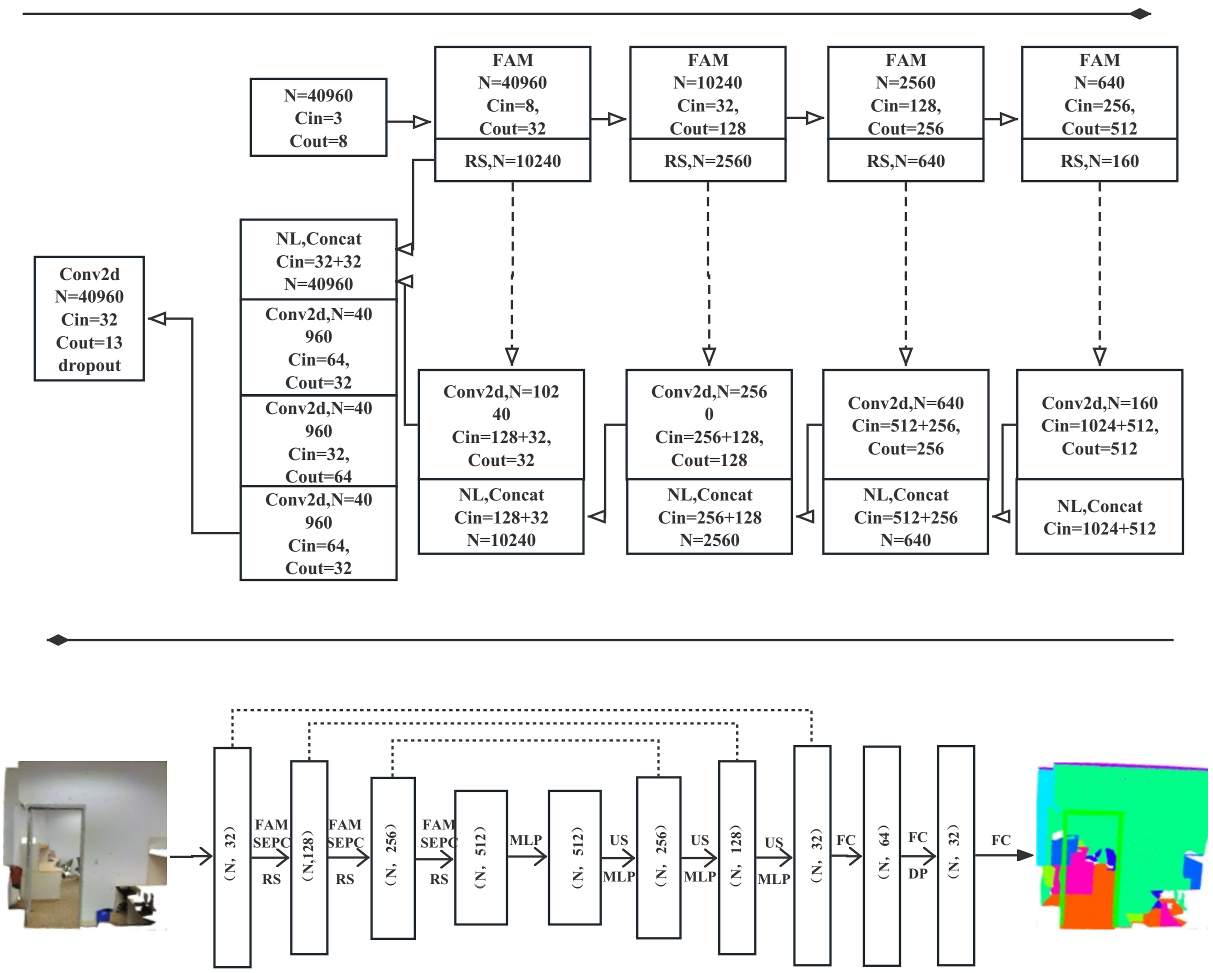

Fig. 9: Overall architecture of RanSeNet

TABLE II: The result of semantic segmentation on S3DIS area 5

| | OA | mIoU | ceiling | floor | wall | beam | column | window | door | table | chair | sofa | bookcase | board | clutter |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **PointNet** | - | 41.1 | 88.8 | 97.3 | 69.8 | 0.1 | 3.9 | 46.3 | 10.8 | 59.0 | 52.6 | 5.9 | 40.3 | 26.4 | 33.2 |
| **SegCloud** | - | 48.9 | 90.1 | 96.1 | 69.9 | 0.0 | 18.4 | 38.4 | 23.1 | 70.4 | 75.9 | 40.9 | 58.4 | 13.0 | 41.6 |
| **PointCNN** | 85.9 | 57.3 | 92.3 | 98.2 | 79.4 | 0.0 | 17.6 | 22.8 | 62.1 | 74.4 | 80.6 | 31.7 | 66.7 | 62.1 | 56.7 |
| **SPGraph** | 86.4 | 58.0 | 89.4 | 96.9 | 78.1 | 0.0 | 42.8 | 48.9 | 61.6 | 84.7 | 75.4 | 69.8 | 52.6 | 2.1 | 52.2 |
| **PointWeb** | 87.0 | 60.3 | 92.0 | 98.5 | 79.4 | 0.0 | 21.1 | 59.7 | 34.8 | 76.3 | 88.3 | 46.9 | 69.3 | 64.9 | 52.5 |
| **RandLA-Net** | 87.2 | 63.17 | 91.2 | 97.4 | 81.7 | 0.0 | 30.5 | 58.1 | 49.6 | 78.4 | 83.9 | 54.5 | 72.3 | 70.3 | 53.36 |
| **RanSeNet** | 88.67 | 64.27 | 91.65 | 97.20 | 80.99 | 0.0 | 21.83 | 61.06 | 47.67 | 78.57 | 86.97 | 74.28 | 71.92 | 70.96 | 52.43 |

all classes. These metrics provide a comprehensive measure of the model's ability to accurately segment different classes in the point cloud data. The results of the evaluation provide insights into the effectiveness and efficiency of RanSeNet in handling large-scale point cloud data for semantic segmentation tasks, which is crucial for its real-world applicability in various indoor environments.

### *B. The Data Set S3DIS*

The S3DIS dataset is a large 3D indoor dataset designed for semantic segmentation tasks. It consists of 271 rooms distributed across six areas, including teaching buildings and office areas. Each room contains multiple objects such as tables, chairs, windows, and more. The primary goal of the dataset is to locate and segment semantic objects from these large-scale three-dimensional point clouds. The dataset covers an impressive area of more than 6,000 square meters, containing over 215 million points. It comprises 13 meaningful categories, including ceilings, floors, walls, beams, columns, windows, doors, tables, chairs, sofas, bookcases, blackboards, and sundry objects.

The S3DIS dataset stands as a pivotal resource, indispensable for the rigorous assessment and benchmarking of semantic segmentation algorithms within authentic indoor settings. With its vast scale and diverse array of object categories, a challenging yet invaluable dataset is presented, one that serves as a catalyst for the advancement of research in the domains of 3D point cloud semantic segmentation.

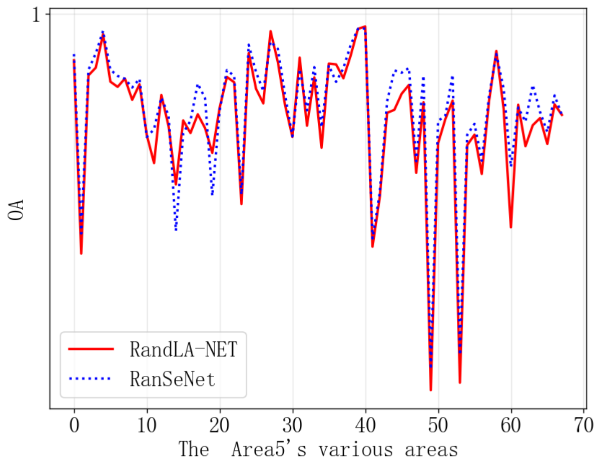

Fig. 10: Area5's subregion Semantic segmentation results

### C. Experimental Results

The RanSeNet model was trained for 100 cycles during the experiment. The test set consisted of Area5, known for its intricate spatial structure within the S3DIS dataset, while the remaining five regions served as the verification set. Fig. 10 shows the accuracy of the model in each subarea (meeting room, storage, toilet, and so on) in Area 5, the accuracy comparison results with baseline RandLA-Net are shown in Fig. 10.

During the training of RanSeNet, a four-layer network architecture was used with feature sizes of 32, 128, 256, and 512. The Adam optimizer was employed with a learning rate of 0.01 and a decay rate of 0.05 to optimize the model parameters. Throughout the training process, the grid sampling used a mesh size of 0.04, and the downsampling ratios for each layer were set to 8, 4, 4, and 2, respectively. The parameter K, which determines the number of neighbors in the local neighborhood for feature learning, was set to 16. To train the RanSeNet model, a fixed number of 40,960 points were sampled from each point cloud as input data. The evaluation results indicate that the average mIoU achieved is 64.27%, while the overall accuracy is 88.67%.

TABLE III: Ablation experiments based on model time efficiency were conducted in each room of area 5

| | time (ms) |
|---|---|
| **Baseline** | 332 |
| **RanSeNet** | 156 |

## IV. Result Analysis

### A. Evaluation of Attention Mechanism (SE-PC)

Using traditional point semantic segmentation algorithms, the point cloud may not effectively distinguish the importance of different feature channels, leading to inadequate feature extraction and reduced accuracy. To address this issue, the paper introduces an attention mechanism that explicitly models the relationships between different feature channels. With the attention mechanism, the model can adaptively learn different feature channel weights. It assigns higher weights to valid features and lower weights to less relevant ones. This allows the limited computing resources to focus on filtering out more important information for the current task, improving the adequacy of feature expression, and consequently enhancing the accuracy of classification.

By incorporating the attentional mechanism, the model gains the ability to emphasize relevant features and suppress less useful ones, enabling it to make more informed decisions during the semantic segmentation process. This approach effectively addresses the challenges of feature extraction in point cloud data, leading to improved performance in various classification tasks.

### B. Evaluate the details of the Transformer structure

With an increase in the number of transformer layers, the network's capacity becomes larger, and its expression capacity becomes stronger. However, this may also lead to issues such as slow network convergence and gradient disappearance [18]. To address these challenges, RanSeNet utilizes point subsampling when encoding the point cloud data. We have improved the layers and sampling ratio of the Transformer model and conducted multiple comparison experiments. The results demonstrate that RanSeNet outperforms other configurations, showing the best overall performance.

By carefully selecting the number of layers and optimizing the point subsampling approach, RanSeNet strikes a balance between network capacity and computational efficiency, resulting in superior results for point cloud semantic segmentation. This highlights the significance of the design choices made in RanSeNet, which contributes to its effectiveness in handling large-scale point cloud data.

Indeed, point cloud data is a unique data structure with a large number of points and a specific spatial arrangement, making it challenging to process efficiently. However, point cloud data holds great practical significance for research on scene perception and understanding.

In the context of point cloud semantic segmentation, real-time performance is a critical aspect as it directly impacts the usability of the technology in various applications. The speed at which the model can process point clouds is a key factor in making it practical for real-world scenarios.

The proposed method in RanSeNet has demonstrated significant improvements in processing speed, as shown in Table III. By leveraging attention mechanisms and optimizing the transformer layers and sampling ratio, RanSeNet achieves a good balance between accuracy and efficiency. The results demonstrate the model's capability to process large-scale point cloud data in real-time, making it a valuable tool for various applications in scene perception and other related fields.

### C. Ablation analysis

In order to demonstrate the performance of the improved model, ablation experiments were performed, and the influence of the GELU function and the attention module on the segmentation accuracy could be seen.

TABLE IV: The mIoU and OA scores of all ablated networks based on our full RanSeNet.

| | OA(%) | mIoU(%) | ceiling | floor | wall | beam | column | window | door | table | chair | sofa | bookcase | board | clutter |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Replace with RELU** | 88.1 | 62.37 | 92.66 | 97.19 | 79.45 | 0.0 | 25.60 | 60.34 | 30.84 | 78.87 | 85.79 | 61.69 | 70.44 | 68.17 | 52.90 |
| **Remove Attention Mechanism** | 87.5 | 63.39 | 93.00 | 97.36 | 80.04 | 0.0 | 19.98 | 61.57 | 43.67 | 78.03 | 87.74 | 72.56 | 71.21 | 68.13 | 53.46 |
| **The Full framework (RanSeNet)** | 88.67 | 64.27 | 91.65 | 97.20 | 80.99 | 0.0 | 21.83 | 61.06 | 47.67 | 78.57 | 86.97 | 74.28 | 71.92 | 70.96 | 52.43 |

Initially, the RELU activation function was employed in the feature aggregation module FAM. Subsequently, the attention mechanism SE-PC was removed to nullify the influence of the model on the point cloud feature channels.

Table IV presents a comparison of overall accuracy (OA) and mean Intersection over Union (mIoU) scores for all ablation analysis results. From these results, we can observe the effect of using different activation functions on accuracy and how the removal of the attention module affects the performance by not effectively preserving useful features.

The ablation study provides valuable insights into the contributions of the GELU function and the attention module in improving the segmentation accuracy of the model. It demonstrates how the proposed neural units complement each other to achieve excellent performance in point cloud semantic segmentation. This analysis further validates the effectiveness of the enhancements introduced in the RanSeNet model.

## V. Conclusion

In this paper, a lightweight neural network model called RanSeNet has been designed for the efficient semantic segmentation of expansive point cloud datasets within picturesque domains. The empirical findings from the S3DIS dataset demonstrate the adeptness of our model in swiftly and accurately handling substantial data volumes. RanSeNet offers two key advantages: efficient handling of large-scale point cloud data for real-world applications and superior performance compared to other methods, ensuring accurate and real-time semantic segmentation. By incorporating attention mechanisms, point subsampling, and optimized transformer layers, RanSeNet strikes a balance between accuracy and efficiency, making it a valuable solution for various scene perception tasks. The experiments confirm the effectiveness and practicality of our model, making it a promising choice for point cloud semantic segmentation in real-world scenarios.

## References

[1] N. Leal, E. Leal, S. T. German, et al, “A Linear Programming Approach for 3D Point Cloud Simplification,” IAENG International Journal of Computer Science, vol. 44, no.1, pp. 60-67, 2017.

[2] Y. L. Guo, H. Y. Wang, Q. Y. Hu, et al, “Deep Learning for 3D Point Clouds: A Survey [J],” EEE transactions on pattern analysis and machine intelligence, vol. 12, pp. 4338-4364, 2020.

[3] Y. Q. Zhao, X. G. Ma, B. Hu, et al, “A large-scale point cloud semantic segmentation network via local dual features and global correlations [J],” Computers & Graphics, vol. 111, pp. 133-144, 2023.

[4] Z. Fang, B. Y. Xiong, F. Liu, et al, “Sparse point-voxel aggregation network for efficient point cloud semantic segmentation [J],” IET Computer Vision, vol. 16, pp. 644-654, 2022.

[5] C. R. Qi, H. Su, K. Mo, et al. “PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation [C],” IEEE, pp. 77-85, 2017.

[6] C. R. Qi, Y. Li, S. Hao, et al, “PointNet++: deep hierarchical feature learning on point sets in a metric space [C],” NIPS, pp. 5105–5114, 2017.

[7] G. Q. Feng, W. L. Li, X. L. Zhao, et al, “LessNet: Lightweight and efficient semantic segmentation for large-scale point clouds [J],” IET Cyber-Systems and Robotics, vol. 4, pp. 107-115, 2022.

[8] Q. Hu, B. Yang, L. Xie, et al, “Randla-Net: Efficient semantic segmentation of large-scale point clouds [C],” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 11105-11114, 2020.

[9] A. Boulch, B. L. Saux, N. Audebert “Unstructured Point Cloud Semantic Labeling Using Deep Segmentation Networks,” pp. 17-24, 2017.

[10] Q. Y. Hu, B. Yang, L. H. Xie, et al, “RandLA-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds [C],” 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, pp. 11108-11117, 2020.

[11] I. Armeni, S. Sax, A. R. Zamir, et al, “Joint 2D-3D-semantic data for indoor scene understanding,” In CVPR, 2017.

[12] B. Hfle, W. Muck, M. DUTTER, et al, “Detection of building regions using airborne LiDAR : a new combination of raster and point cloud based GIS methods,” proceedings of the geoinformatics forum Salzburg, pp. 66-75, 2009.

[13] B. Kamiński, P. Praat , F. Théberge, et al, “Artificial Benchmark for Community Detection (ABCD)—Fast random graph model with community structure,” Network Science, vol. 9, pp. 1-26, 2021.

[14] X. Li, Z. Dai, L. He, et al, “A K-nearest neighbor indoor fingerprint location method based on coarse positioning circular domain and the highest similarity threshold,” IOP Publishing Ltd, 2022.

[15] T. V. Nguyen, Z. Song , S. Yan, et al, “STAP: Spatial-Temporal Attention-Aware Pooling for Action Recognition,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, pp. 77-86, 2015.

[16] H. Jie, S. Li, S. Gang, et al, “Squeeze-and-Excitation Networks [C],” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) IEEE, 2018.

[17] Q. G. Jia, X. Z. Dong, “Large-factor single image super-resolution based on back projection and residual block,” Twelfth International Conference on Graphics and Image Processing, 2021.

[18] B. Yang, J. N. Wang, Q. Y. Hu, et al, “Learning object bounding boxes for 3D instance segmentation on point clouds,” NeurIPS, 2019.