LGFormer: Lightweight Local-Global Transformer for Indoor Point Cloud Segmentation

Yuwei Zhang[1], Fashuai Li[2, *], Yiyi Liu[1], Ping Wang[1], Yuwei Chen[3], Biao Xiong[1, *]

[1] Wuhan University of Technology, Wuhan, China - (zhangyuwei, yiyiliu, wangping, b.xiong)@whut.edu.cn
[2] The Advanced Laser Technology Laboratory of Anhui Province, Hefei, China - lifashuai@gmail.com
[3] University of Chinese Academy of Sciences, Hangzhou, China – yuwei.chen@ucas.ac.cn
[*] Corresponding Authors

Keywords: Point Cloud, Semantic Segmentation, Transformer, Graph Convolution Network.

Abstract

Semantic segmentation of indoor point clouds is a fundamental task in 3D scene understanding, supporting applications such as virtual reality, indoor navigation, and building management. Point-based transformer models achieve high accuracy but require substantial computational resources, while superpoint-based methods are more efficient yet often less precise. To address this trade-off, we propose **LGFormer**, a lightweight framework that integrates Graph Convolutional Networks (GCN) and transformers to jointly capture local and global contextual features. The method constructs a superpoint-based topology graph, where local features are extracted using GCN and global dependencies are modeled through transformer layers. Experiments on the S3DIS and ScanNet++ datasets demonstrate that LGFormer achieves 90.7% and 88.5% segmentation accuracy, respectively, while reducing inference time by more than 99% compared with point-based transformers. By effectively leveraging superpoints and local-global feature fusion, LGFormer delivers competitive accuracy with significantly lower computational cost, making it suitable for large-scale indoor scene analysis.

1. Introduction

Recent advancements in 3D sensing technologies have significantly improved the digital representation of real-world environments(Xiao et al., 2023). Point clouds, as detailed 3D geometric descriptions, are now essential for applications such as virtual reality, autonomous navigation, robotic perception, and building lifecycle management (Xiong et al., 2023, Zhu et al., 2024). A key challenge in exploiting point cloud data lies in semantic segmentation, which assigns class labels to individual points to capture structural and contextual information crucial for higher-level scene understanding (Qian et al., 2022). Although recent research trends increasingly explore end-to-end systems that directly produce task-specific outputs (Kim et al., 2025, Liu et al., 2025), semantic segmentation remains indispensable for achieving fine-grained understanding and interpretability in complex 3D environments (He et al., 2025).

Existing research on point cloud semantic segmentation can be broadly categorized into projection-based, voxel-based, point-based, and superpoint-based approaches. Projection-based methods (Boulch et al., 2018) convert 3D data into 2D images to exploit mature convolutional networks, but this transformation inevitably leads to the loss of geometric integrity and spatial consistency. Voxel-based methods (Zhou and Tuzel, 2018) discretize the space into regular 3D grids to enable volumetric convolutions, yet they suffer from excessive memory consumption and quantization artifacts. Point-based methods (Qian et al., 2022) directly process raw point sets, preserving geometric detail and permutation invariance, but their local receptive fields limit the modeling of long-range dependencies.

In contrast, superpoint-based methods (Robert et al., 2023) partition point clouds into geometrically homogeneous regions and construct graph representations to enhance computational efficiency. However, they primarily emphasize node-level features while overlooking informative edge attributes and long-range contextual dependencies essential for comprehensive scene understanding, thereby limiting segmentation accuracy. This trade-off between accuracy and efficiency remains a key challenge, particularly for large-scale or resource-constrained applications that demand lightweight yet precise solutions.

To address these challenges, we present **LGFormer**, a lightweight framework that jointly captures local and global contextual features for indoor point cloud segmentation. The input point cloud is partitioned into superpoints represented as graph nodes. The *local context unit* employs GCN model to extract node and edge attributes, while the *global context unit* leverages transformer layers to model long-range dependencies. The fused representations are encoded into graph nodes, and semantic categories are predicted through a fully connected layer. LGFormer achieves robust and efficient scene understanding, attaining 90.7% and 88.5% segmentation accuracy on the S3DIS and ScanNet++ datasets, respectively, while reducing inference time by more than 99% compared with point-based transformers. The main contributions of this paper are as follows:

- **LGFormer**: a lightweight framework integrating GCNs and Transformers for local-global feature learning.
- A **GCN–GRU** module that strengthens long-range feature aggregation among superpoints.
- State-of-the-art performance on S3DIS and ScanNet++ with minimal computational cost.

2. Related Work

This section reviews key developments in point cloud semantic segmentation, focusing on feature extraction, graph neural networks, and attention-based approaches. For a comprehensive overview of deep learning methods for point clouds, please refer to (Xiao et al., 2023, Zhu et al., 2024, He et al., 2025).

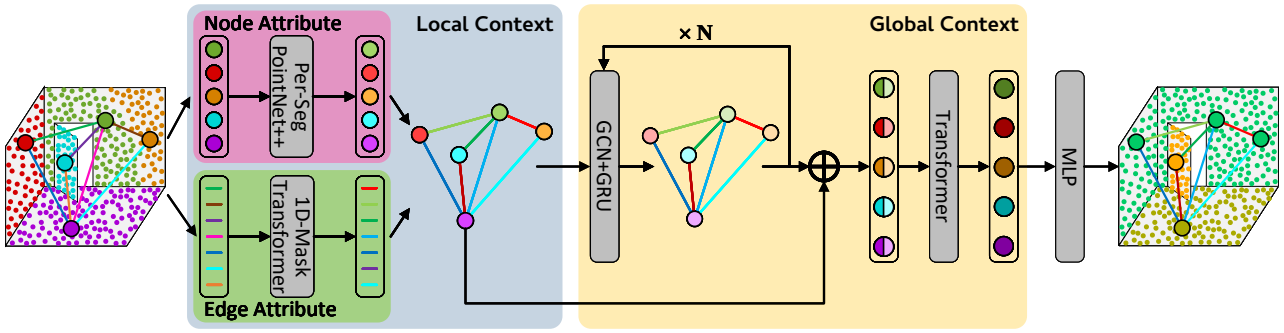

Figure 1. **Overview of LGFormer.** Input point clouds are segmented into superpoints, serving as graph nodes. Using neighboring relationships, a graph topology is constructed and processed by the *local context unit* to extract node and edge attributes locally. The *global context unit* captures long-range contextual information, and features from both local and global contexts are fused and encoded into the nodes. A fully connected layer predicts their semantic categories.

Point Cloud Feature Extraction. Feature extraction is fundamental to point cloud semantic segmentation. Traditional handcrafted methods (Xiong et al., 2014, Xiong et al., 2015) relied on geometric attributes but struggled with complex scenarios due to the limited expressiveness of manually designed features. Deep learning methods addressed these limitations, starting with voxel-based approaches (Maturana and Scherer, 2015, Zhou and Tuzel, 2018), which partition point clouds into regular grids to enable 3D convolutions. While effective, voxelization often results in boundary artifacts and high computational costs.

Point-based methods (Qi et al., 2017) directly process raw points, preserving geometric details and ensuring permutation invariance. PointNeXt (Qian et al., 2022) enhances PointNet++ by incorporating an inverted residual bottleneck design and separable MLPs, enabling efficient and scalable model design. Despite these advancements, point-based methods are inherently limited by the restricted receptive field of convolutional operations, hindering their ability to capture long-range dependencies. Hybrid approaches that combine 3D point cloud features with 2D depth images (Yan et al., 2022) have shown promise in improving segmentation performance, yet effective long-range feature aggregation remains a significant challenge.

GCN-Based Point Cloud Segmentation. GCN (Simonovsky and Komodakis, 2017) provide a natural way to model the unstructured and unordered nature of point clouds by representing them as graphs. DGCNN (Wang et al., 2019b), for instance, uses K-Nearest Neighbor (KNN) to construct dynamic graphs, aggregating features from local neighborhoods. However, as the size of the point cloud grows, the graph complexity increases, impacting efficiency.

Superpoint-based methods, such as Super Point Graph (SPG) (Landrieu and Simonovsky, 2018), cluster point clouds into geometrically homogeneous regions, significantly reducing graph scale and computation. Variants of SPG (Liang et al., 2019, Feng et al., 2023) have explored hierarchical downsampling and object-level graphs to further optimize performance. Despite their advantages, GCN-based methods often struggle to capture long-range dependencies and require extensive preprocessing.

Attention-Based Point Cloud Segmentation. Attention mechanisms offer a powerful tool for modeling relationships between data points by focusing on the most relevant features. Initial efforts applied attention-pooling (Hu et al., 2020), self-attention blocks (Zhang et al., 2020), or graph attention convolutions (Wang et al., 2019a) to enhance local feature aggregation. More recently, transformer-based architectures have emerged as state-of-the-art solutions for point cloud tasks (Vaswani et al., 2017). For example, Point Transformer V3 (PTv3) (Wu et al., 2024) leverages self-attention to model both context relationships effectively.

Superpoint Transformer (SPT) (Robert et al., 2023) integrates attention mechanisms into a superpoint-based graph framework, using hypergraphs to represent multi-scale segmentations. This approach extracts contextual features from superpoints but remains computationally expensive. Similar methods, such as OctFormer (Wang, 2023) and FlatFormer (Liu et al., 2023), have demonstrated the potential of transformers in enhancing segmentation accuracy. However, the high computational overhead of these models limits their practicality.

Summary. While GCNs and transformers excel at capturing local and global features, respectively, they are often employed in isolation. Our proposed method, LGFormer, addresses these limitations by integrating GCNs and transformers into a unified framework. This approach balances efficiency and accuracy, enabling robust segmentation of indoor point clouds with reduced computational costs.

3. Method

The spatial distribution of point clouds reveals essential surface properties, providing critical contextual information for semantic understanding. To leverage this, we propose a framework that efficiently captures local and global contextual features, enabling effective representation of complex, high-dimensional data with a lightweight network. This section outlines our methodology, integrating local and global context learning for robust semantic segmentation.

3.1 Overview

As illustrated in Figure 1, our method employs two modules to capture *local* and *global* contextual information from point clouds. The local context module extracts features from nodes and their adjacent edges in localized regions using spatial coordinates. The global context module combines a GCN module with an Transformer module to learn broader relationships, enabling a comprehensive understanding of the scene. The features from these modules are integrated into node representa-

tions, which are then processed through a fully connected layer to predict semantic categories.

Prior to model input, the point cloud is preprocessed into a structured format. It is segmented into superpoints, and an adjacency graph $G(V, E)$ is constructed following the SPG method (Landrieu and Simonovsky, 2018). Nodes $v \in V$ represent superpoints, while edges $e \in E$ encode adjacency relationships. For clarity, the terms ”node” and ”patch” are used interchangeably.

3.2 Local Context Unit

The local context unit analyzes a topology graph derived from preprocessing, where nodes and edges possess distinct features. This unit extracts features from both nodes and edges to capture the intricacies of local spatial relationships, enabling the identification of hidden patterns.

3.2.1 Node Features. Each node in the topology graph encompasses a set of 3D points, varying in number and spatial distribution. Extracting meaningful features from these nodes requires standardization of this point set into feature vectors. We utilize PointNet++ (Qi et al., 2017) to condense the 3D coordinates of points within each node into a 32-dimensional feature vector.

Our approach exclusively uses spatial coordinates, excluding color and intensity attributes due to their inconsistency across datasets. This focused strategy ensures robustness across diverse scenarios. Furthermore, the feature extraction process is strictly confined to individual nodes, capturing critical local information, including boundary details, without interference from neighboring nodes.

3.2.2 Edge Features. Edges in the topology graph represent connections between nodes rather than groups of points, posing challenges for direct feature extraction. To address this, we manually design edge features that capture spatial relationships and connection strengths between adjacent nodes. These features are subsequently transformed into high-dimensional representations for further analysis.

Initialization of Edge Features. Shown in Table 1, The initial edge feature vector $F_{e_{(u,v)}}$ comprises two main components: (1) **Patch Difference (PD):** Captures the relative geometric relationship between nodes, including differences in length, surface area, volume, and point counts, along with angular and centroidal distances; (2) **Patch Connection (PC):** Quantifies the connection strength between nodes by analyzing the spatial distribution of connecting points, including attributes like point count, centroid, and eigenvalues. This 23-dimensional vector encapsulates spatial and connection attributes, normalized and used as the input for subsequent feature transformation.

Local Attention for Edge Features. To effectively extract high-dimensional contextual information between edges, we employ a masked attention mechanism (Figure 2). This module projects the 23-dimensional edge feature vector into a 512-dimensional space using Multi-Layer Perceptrons (MLPs), which is then divided into four subspaces for multi-head attention processing. Each subspace independently calculates the weights of adjacent edges, preserving local correlations while ignoring irrelevant global connections through a 1D masking mechanism. The resulting 512-dimensional feature vector represents enhanced edge features optimized for contextual learning.

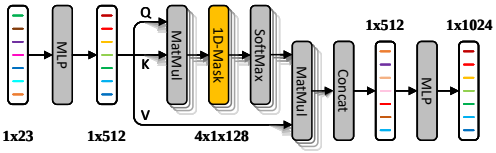

Figure 2. Edge feature extraction with masked attention.

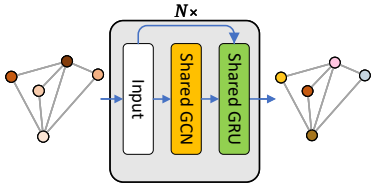

Figure 3. Graph Convolutions with Gated Recurrent Units (GCN-GRU).

Type	Feature Name	Dimension	Description
PD	Length Difference	1	$\mid length(u) - length(v) \mid$
PD	Mean length	1	$(length(u) + length(v))/2$
PD	Surface Discrepancy	1	$\mid surface(u) - surface(v)$
PD	Mean Surface	1	$(surface(u) + surface(v))/2$
PD	Volume Discrepancy	1	$\mid volume(u) - volume(v) \mid$
PD	Mean Volume	1	$((volume(u) + volume(v))/2$
PD	Discrepancy of Point Count	1	$var(\parallel u \parallel, \parallel v \parallel)$
PD	Mean of Point Count	1	$mean(\parallel u \parallel, \parallel v \parallel)$
PD	Angle	1	$cos^{-1} \frac{norm(u) \cdot norm(v)}{\mid norm(u) \mid \cdot \mid norm(v)}$
PC	Distance	1	$\mid mean(u) - mean(v) \mid$
PC	CP Count	1	$count(p), p \in (u \cap v)$
PC	CP Center	3	$mean(p), p \in (u \cap v)$
PC	Eigen Value of CP	3	$(\lambda_1, \lambda_2, \lambda_3) = eigen(u \cap v)$
PC	CP Linear	1	λ_1/λ_2
PC	CP Scatter	1	λ_3/λ_1
PC	Connection Discrepancy	3	$var(u \cap v)$
PC	Connection Distance	1	$mean(\mid u \cap v \mid)$

Table 1. Edge Features for describing Patch Difference (PD) and Patch Connection (PC).

3.3 Global Context Unit

The global context unit addresses two key aspects of the topology graph: connectivity between nodes and the overall spatial relationships among nodes. This unit employs graph convolutions and self-attention mechanisms to extract these features comprehensively.

3.3.1 GCN-GRU model. GCN propagates information through edges, updating node features based on their connections. Successive convolutions expand the receptive field, enabling the model to aggregate information across broader regions. To mitigate information loss during deep convolutions, we integrate Gated Recurrent Units (GRUs) to retain intermediate features and balance feature propagation (Figure 3). Starting with the 32-dimensional features from the local context unit, this process is repeated for 20 iterations to maximize feature integration.

3.3.2 Global Attention Mechanism. To complement the GCNs, a self-attention mechanism processes node features without adjacency constraints. Using an MLP, we derive Query, Key, and Value (QKV) matrices from the high-dimensional node features. The multi-head attention mechanism then extracts global contextual information, enabling nodes to capture long-range dependencies. This global perspective, combined with the local features, enriches the node representations for robust semantic segmentation.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-775-2026 | © Author(s) 2026. CC BY 4.0 License.

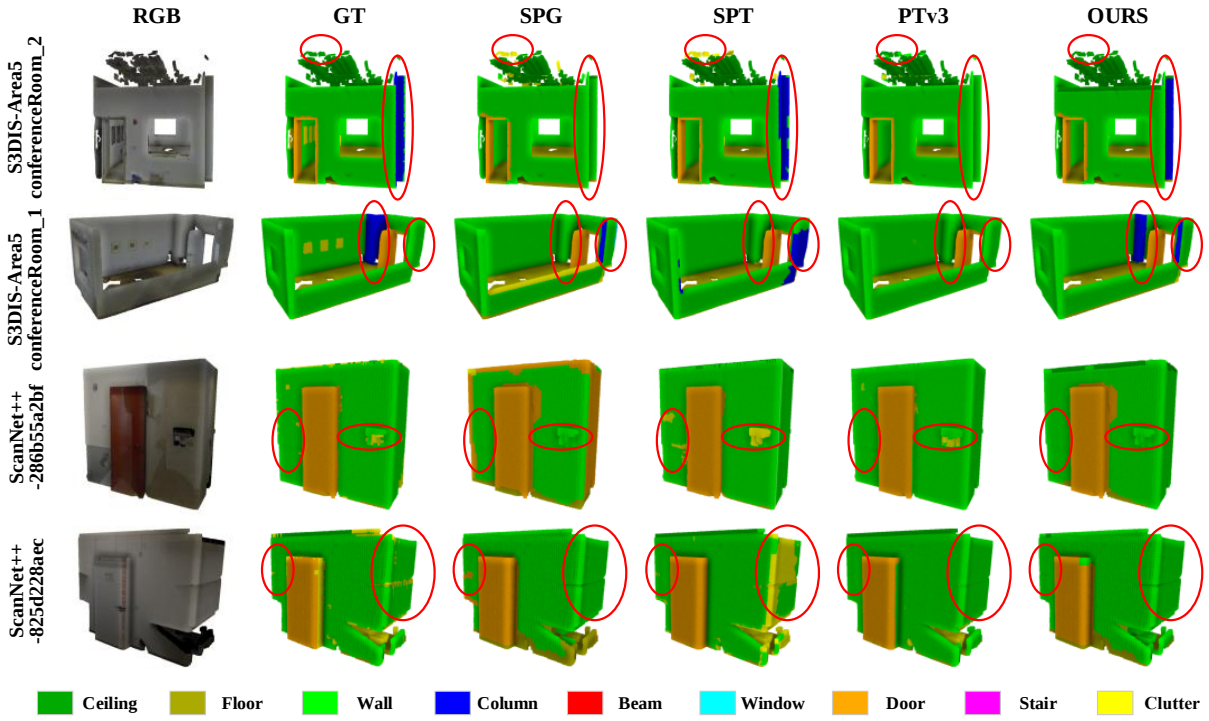

Figure 4. Qualitative comparison of point cloud segmentation on the S3DIS and ScanNet++ test sets. We visualize the segmentation results for 9 different classes and observe that our algorithm performs the best using only XYZ as input. This is particularly obvious for the Ceiling, Floor and Wall categories. It can be attributed to our designed graph structure, which helps clearly define the topological relationships between these categories in indoor scenes. As a result, we are able to extract more consistent features to aid in segmentation.

4. Experiments

4.1 Datasets and Experimental Settings

Datasets. We evaluate our method on two widely used benchmark datasets for indoor point cloud semantic segmentation: ScanNet++ (Dai et al., 2017) and S3DIS (Armeni et al., 2016). ScanNet++ consists of 460 scenes with 74.3 billion points, while S3DIS comprises 272 indoor scenes with 233 million points. Both datasets represent diverse indoor environments, making them ideal for evaluating segmentation performance. For the S3DIS dataset, we adopt a 6-fold cross-validation strategy, where each area serves as a test set while the remaining areas are used for training. For the ScanNet++ dataset, we follow the official split into training, testing, and validation sets, using the validation set for evaluation due to the lack of ground-truth labels in the test set. Our study focuses on seven key categories: *Ceiling*, *Floor*, *Wall*, *Column*, *Beam*, *Window*, *Door*, *Stair* and *Clutter*, which are crucial for applications such as floor plan reconstruction, building management, and indoor navigation. Note that, the test sets in ScanNet++ does not include the category of Column and Stair, which are ignored here.

Evaluation Metrics. We use three standard metrics to assess segmentation performance: overall accuracy (oAcc), mean class accuracy (mAcc), and mean intersection-over-union (mIoU) (Tang et al., 2022). These metrics provide a holistic evaluation, measuring both overall and class-wise accuracy.

Implementation Details. The proposed method is implemented using PyTorch. Training is conducted with the Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$) with an initial learning rate of 1×10^{-3} and a weight decay of 1×10^{-4}. The learning rate is decayed at 180, 260, 340, and 420 epochs using the MultiStepLR scheduler. Each training session spans 500 epochs with a batch size of 4. Weighted cross-entropy is used as the loss function, where the weights are normalized based on the number of superpoints per class. All experiments are conducted on a single NVIDIA A40 GPU, and the adjacency threshold is set to 5 for graph construction.

Model	oAcc (%)	mAcc (%)	mIoU (%)	Params (M)
SPG	86.6	69.1	59.5	0.3
SPT	88.8	82.9	70.2	0.2
PTv3	**92.2**	82.4	**75.4**	46.2
Ours	90.7	**85.1**	67.5	0.8

Table 2. Comparison experiments on S3DIS dataset.

Model	oAcc (%)	mAcc (%)	mIoU (%)	Params (M)
SPG	80.0	66.2	43.2	0.3
SPT	84.4	**74.0**	59.1	0.2
PTv3	79.5	66.6	55.6	46.2
Ours	**88.5**	66.8	**60.2**	0.8

Table 3. Comparison experiments on ScanNet++ dataset.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-775-2026 | © Author(s) 2026. CC BY 4.0 License.

Model	mIoU	ceiling	floor	wall	beam	window	door	clutter
SPG	43.19	72.17	93.03	63.71	16.97	29.89	24.37	2.18
SPT	59.12	74.45	91.55	71.61	0.63	48.18	**57.21**	**70.24**
PTv3	55.64	73.56	87.33	60.99	0.00	**52.28**	51.22	64.07
Ours	**60.24**	**83.40**	**95.30**	**77.99**	**50.22**	43.86	24.91	46.00

Table 4. Class-wise quantitative comparison on ScanNet++.

4.2 Results

Quantitative Results. Tables 2 and 3 present the results on the S3DIS and ScanNet++ datasets, respectively. Our method demonstrates competitive or superior performance compared to state-of-the-art approaches such as SPG (Landrieu and Simonovsky, 2018), SPT (Robert et al., 2023), and PTv3 (Wu et al., 2024). While several advanced methods, including PointNext (Qian et al., 2022) and Swin3D (Yang et al., 2024), achieve results comparable to PTv3 with similar parameter complexity, we focus our comparison on PTv3 for clarity. PTv3 is a representative architecture that captures global contextual information.

Although several advanced methods—such as PointNext (Qian et al., 2022) and Swin3D (Yang et al., 2024)—achieve performance comparable to PTv3 with a similar level of parameter complexity, we center our comparison on PTv3 for clarity. PTv3 serves as a representative model that incorporates global contextual information without relying on full global attention.

On the S3DIS dataset (Table 2), our model achieves an mAcc of 85.1%, outperforming all other methods. Although PTv3 achieves a slightly higher oAcc of 92.2%, it requires 46.2M parameters compared to our lightweight model with only 0.8M parameters. This highlights the efficiency of our approach in delivering high accuracy with significantly fewer resources. The mAcc of 85.1%, surpassing SPG, SPT, and PTv3, further demonstrates our method's balanced segmentation performance across all classes. Our mAcc exceeds that of PTv3 primarily because the lighter model architecture leads to a higher number of false-positive predictions. In particular, background clutter is frequently misclassified as structural classes such as walls or floors, which inflates point-wise accuracy while degrading region-level consistency.

On the ScanNet++ dataset (Table 3), our method achieves the highest mIoU of 60.2%, surpassing SPG, SPT, and PTv3. Additionally, we achieve the highest oAcc of 88.5%, indicating robust performance in diverse indoor scenes. These results underscore the effectiveness of our graph-based framework in capturing both local and global contextual information.

Class-Wise Performance. Table 4 presents a detailed class-wise comparison on ScanNet++. Note that, the test sets in ScanNet++ does not include the category of *Column* and *Stair*, which are ignored here. Our method achieves superior performance in key structural categories, such as *Ceiling* (83.40%), *Floor* (95.30%), and *Wall* (77.99%). This improvement is attributed to the GCN module, which effectively captures geometric structures and adjacency relationships. For challenging categories like *Beams*, our method achieves 50.22%, significantly outperforming other methods. While categories such as *Windows* and *Doors* show lower performance due to their ambiguity and overlap with adjacent classes, our method demonstrates competitive performance overall.

Efficiency. Table 5 compares the computational efficiency of different methods. Our approach achieves competitive preprocessing (2.4h) and training times (2.3h) while maintaining the fastest inference speed (2.9s). In contrast, PTv3 eliminates preprocessing but incurs significantly higher training and inference times (15.2h and 1375.3s, respectively). Despite its smaller model size, SPT has slower inference times (4.4s) compared to our method, due to the complexities of its transformer architecture. These results demonstrate the practicality of our method for real-world applications.

Qualitative Results. Figure 4 illustrates segmentation results on the S3DIS and ScanNet++ datasets. Our method excels in identifying structural categories such as *Ceiling*, *Floor*, and *Wall*, with visibly more accurate and consistent segmentation compared to other methods. The improved performance is attributed to our graph-based framework, which emphasizes structural relationships and topological consistency.

4.3 Discussion

Strengths and Limitations. Our method excels in recognizing structural components like *Ceiling* and *Wall* due to the effective integration of GCN and GRU modules, which encode topological relationships and global features. This performance benefits applications such as floor plan reconstruction and indoor navigation. However, lower accuracy on ambiguous classes like *Windows* and *Doors* arises from overlaps with adjacent categories. Future work could address this by incorporating RGB features for richer context.

Ablation Studies. Ablation results on the S3DIS dataset (Figure 5) highlight the synergy between local and global feature units, with significant performance drops when either is removed. In the global unit, removing the Transformer leads to the largest performance drop, underscoring its role in capturing long-range dependencies, while the GCN remains critical for feature extraction.

Impact of GRU Iterations. Our model integrates GCNs with GRUs, using intermediate convolutional features as hidden states to capture global node connectivity across receptive fields (Figure 3). Table 6 shows that the GRU module consistently improves segmentation performance (oAcc, mAcc, and mIoU) by adaptively preserving relevant features. Accuracy increases steadily up to 20 iterations, effectively modeling long-range dependencies. Beyond this point, improvements plateau, indicating diminishing returns due to dataset constraints and the model's receptive field limits.

5. Conclusion

We proposed **LGFormer**, a lightweight framework for indoor point cloud semantic segmentation that integrates transformers and GCN modules to effectively learn local-global contextual information. LGFormer achieves state-of-the-art performance,

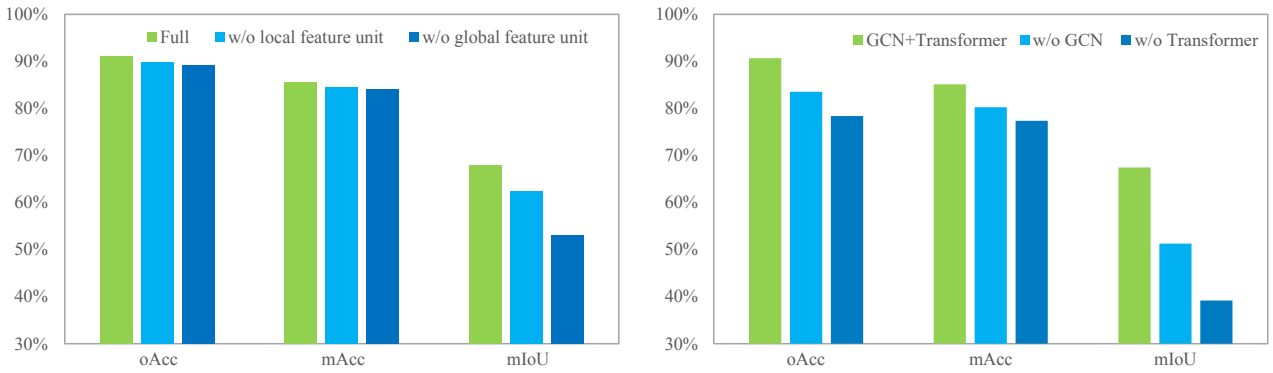

Figure 5. Ablation study evaluating the contributions of the local feature unit and global feature unit (left), and the roles of GCN and Transformer in global feature extraction (right).

Model	Preprocessing (h)	Training (h)	Inference (s)
SPG	2.4	2.8	3.4
SPT	2.9	**1.6**	4.4
PTv3	-	15.2	1375.3
Ours	**2.4**	2.3	**2.9**

Table 5. Efficiency comparison on ScanNet++ test sets.

GRU Times	5	10	15	20	25
oAcc (%)	87.82	89.51	90.08	**90.70**	90.29
mAcc (%)	83.83	84.19	84.31	**85.13**	84.67
mIoU (%)	60.65	64.16	63.57	**67.47**	66.76

Table 6. The influence of GRU module.

with an oAcc of 90.7% and mAcc of 85.1% on S3DIS, and an oAcc of 88.5% and mIoU of 60.2% on ScanNet++, outperforming prior methods in both accuracy and efficiency. Ablation studies confirm the effectiveness of the local-global transformer and GRU modules, underscoring their contributions to robust feature extraction and segmentation precision.

While our method demonstrates strong performance, challenges remain in handling errors from superpoint construction and reliance on hand-crafted edge features. Future work will focus on developing boundary-aware superpoint construction and exploring adaptive feature learning to further improve segmentation accuracy. Additionally, extending LGFormer to outdoor environments with more complex topologies represents an exciting direction for future research.

References

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S., 2016. 3d semantic parsing of large-scale indoor spaces. *CVPR*, 1534–1543.

Boulch, A., Guerry, J., Le Saux, B., Audebert, N., 2018. SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks. *Computers & Graphics*, 71, 189–198.

Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., Nießner, M., 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*.

Feng, M., Hou, H., Zhang, L., Wu, Z., Guo, Y., Mian, A., 2023. 3d spatial multimodal knowledge accumulation for scene graph prediction in point cloud. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9182–9191.

He, Y., Yu, H., Liu, X., Yang, Z., Sun, W., Anwar, S., Mian, A., 2025. Deep learning based 3D segmentation in computer vision: A survey. *Information Fusion*, 115, 102722.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. Randla-net: Efficient semantic segmentation of large-scale point clouds. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11108–11117.

Kim, M. J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E. P., Sanketi, P. R., Vuong, Q. et al., 2025. Openvla: An open-source vision-language-action model. *Conference on Robot Learning*, PMLR, 2679–2713.

Landrieu, L., Simonovsky, M., 2018. Large-scale point cloud semantic segmentation with superpoint graphs. *CVPR*, 4558–4567.

Liang, Z., Yang, M., Deng, L., Wang, C., Wang, B., 2019. Hierarchical depthwise graph convolutional neural network for 3d semantic segmentation of point clouds. *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 8152–8158.

Liu, Y., Liu, C., Wang, B., Jiao, W., Wu, B., Fan, L., Chen, Y., Li, F., Xiong, B., 2025. Cage: Continuity-aware edge network unlocks robust floorplan reconstruction. *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 1–12.

Liu, Z., Yang, X., Tang, H., Yang, S., Han, S., 2023. Flatformer: Flattened window attention for efficient point cloud transformer. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1200–1211.

Maturana, D., Scherer, S., 2015. Voxnet: A 3d convolutional neural network for real-time object recognition. *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, IEEE, 922–928.

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.

Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H., Elhoseiny, M., Ghanem, B., 2022. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in neural information processing systems*, 35, 23192–23204.

Robert, D., Raguet, H., Landrieu, L., 2023. Efficient 3d semantic segmentation with superpoint transformer. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17195–17204.

Simonovsky, M., Komodakis, N., 2017. Dynamic edge-conditioned filters in convolutional neural networks on graphs. *CVPR*, 3693–3702.

Tang, L., Zhan, Y., Chen, Z., Yu, B., Tao, D., 2022. Contrastive boundary learning for point cloud segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8489–8499.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, L., Huang, Y., Hou, Y., Zhang, S., Shan, J., 2019a. Graph attention convolution for point cloud semantic segmentation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10296–10305.

Wang, P.-S., 2023. Octformer: Octree-based transformers for 3d point clouds. *ACM Transactions on Graphics (TOG)*, 42(4), 1–11.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., Solomon, J. M., 2019b. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38(5), 1–12.

Wu, X., Jiang, L., Wang, P.-S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H., 2024. Point transformer v3: Simpler faster stronger. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4840–4851.

Xiao, A., Huang, J., Guan, D., Zhang, X., Lu, S., Shao, L., 2023. Unsupervised point cloud representation learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Xiong, B., Jancosek, M., Oude Elberink, S., Vosselman, G., 2015. Flexible building primitives for 3D building modeling. *ISPRS Journal of Photogrammetry and Remote Sensing*, 101, 275–290.

Xiong, B., Jin, Y., Li, F., Chen, Y., Zou, Y., Zhou, Z., 2023. Knowledge-driven inference for automatic reconstruction of indoor detailed as-built BIMs from laser scanning data. *Automation in Construction*, 156, 105097.

Xiong, B., Oude Elberink, S., Vosselman, G., 2014. A graph edit dictionary for correcting errors in roof topology graphs reconstructed from point clouds. *ISPRS Journal of photogrammetry and remote sensing*, 93, 227–242.

Yan, X., Gao, J., Zheng, C., Zheng, C., Zhang, R., Cui, S., Li, Z., 2022. 2dpass: 2d priors assisted semantic segmentation on lidar point clouds. *European Conference on Computer Vision*, Springer, 677–695.

Yang, Y.-Q., Guo, Y.-X., Liu, Y., 2024. Swin3D++: Effective Multi-Source Pretraining for 3D Indoor Scene Understanding. *arXiv preprint arXiv:2402.14215*.

Zhang, F., Guan, C., Fang, J., Bai, S., Yang, R., Torr, P. H., Prisacariu, V., 2020. Instance segmentation of lidar point clouds. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 9448–9455.

Zhou, Y., Tuzel, O., 2018. Voxelnet: End-to-end learning for point cloud based 3d object detection. *CVPR*, 4490–4499.

Zhu, Q., Fan, L., Weng, N., 2024. Advancements in point cloud data augmentation for deep learning: A survey. *Pattern recognition*, 153, 110532.