# LoCoFormer: Rethinking Local Geometric and Semantic Context Representation for Point Cloud Transformers

**Anonymous authors**
Paper under double-blind review

## Abstract

Synergistically capturing local geometric structures and semantic features remains a critical challenge in 3D point cloud analysis. Recent attempts to integrate local point operators into point cloud Transformers have made impressive progress, yet they still struggle to fully exploit local contexts. In this paper, we present LoCoFormer, an effective framework for comprehensive local geometric and semantic representation. Specifically, we identify the limitations of conventional Softmax attention and propose Bilateral Context Attention (BCA). BCA adopts delicate geometric and semantic context branches to separately capture spatial structures and implicit features, and utilizes a bilateral fusion mechanism to modulate attention weights and provide geometric priors, establishing enhanced local contextual dependencies. Furthermore, motivated by the observation that architectural design affects information acquisition and propagation, we modernize the network architecture from three aspects: *Spatial Proximity*, *Raw-point Embedding*, and *Pooling Strategy*, which respectively determine receptive field size, feature initialization, and spatial alignment, thereby improving local context representation. Experimental results demonstrate that LoCoFormer achieves superior performance across multiple challenging benchmarks, including ModelNet40 for shape classification and S3DIS, ScanNet v2, and ScanNet200 for semantic segmentation.

## 1 Introduction

Understanding 3D point clouds is essential for a wide range of real-world applications, such as robot navigation Chen et al. (2024), autonomous driving Huang et al. (2025), and augmented reality Liu et al. (2024). Unlike images composed of regularly arranged pixels, point clouds are inherently sparse and irregular, which brings unique challenges to algorithm designs. To tackle the problems, numerous approaches have been proposed, which can be broadly categorized into transformation-based Su et al. (2018); Xu et al. (2020); Ha et al. (2025) and point-based methods Charles et al. (2017); Wu et al. (2025a); Qu et al. (2026). Recently, Transformers Vaswani et al. (2017); Dosovitskiy et al. (2020); Cocchi et al. (2025) have attracted increasing attention from the point cloud community due to their powerful capability to model long-range dependencies. Point Transformer Zhao et al. (2021) employs vector self-attention to encode long-range context. Inspired by Swin Transformer Liu et al. (2021), Stratified Transformer Lai et al. (2022) applies self-attention to cubic 3D windows with key sampling strategies. Fast Point Transformer Park et al. (2022) utilizes a voxel hashing-based attention module to handle large-scale point clouds.

Despite achieving decent performance, point cloud Transformers remain limited in modeling local geometric priors, due to the inherent constraints of their core operator, Softmax attention Vaswani et al. (2017). To be specific, 1) Softmax attention learns queries, keys, and values solely through linear projections, lacking the capability to explicitly characterize point-pair relationships within local regions; 2) The element-wise addition of attention maps and positional encoding biases fails to fully facilitate the interaction between semantic features and geometric structures. To mitigate these issues, several methods Duan et al. (2023); Wang et al. (2024); Wan et al. (2025) attempt to integrate local point operators into the attention mechanism. However, such approaches treat local structure enhancement and feature aggregation as relatively separate operations, with local point operators are added before or after self-attention. As a result, geometric and semantic contexts are not



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

jointly exploited during the attention calculation, which still struggles to capture sufficient local priors. Furthermore, we observe that architecture modernization has flourished in MLP-based networks Qian et al. (2022); Thomas et al. (2024); Zeng et al. (2025), demonstrating its potential to provide improved local contextual information. This inspires us to systematically explore architectural design for effective local context representation and propagation in point cloud Transformers, thereby promoting the generalization ability.

To harvest comprehensive **Lo**cal geometric and semantic **Co**ntexts, we present **LoCoFormer**, a novel framework motivated by our analysis (see Section 3.1) of point cloud Transformers from the perspectives of attention mechanism and architectural design. Specifically, we propose Bilateral Context Attention (BCA), which adopts a dual-branch design. The geometric context branch operates on point coordinates $\mathcal{P}$ to capture spatial distributions of the point cloud, while the semantic context branch utilizes the refined query $Q_{\text{rfn}}$ and vanilla query $Q$ to progressively aggregate features $\mathcal{F}$, extracting implicit semantic cues. By encoding two distinct input elements, our BCA characterizes point clouds with complementary knowledge, resulting in robust local context representation. During this process, a bilateral fusion mechanism is introduced for effective interaction between geometric and semantic contexts. It employs learnable geometric encodings to modulate attention weights computed from the semantic context and injects geometric priors during feature aggregation. This enables BCA to jointly leverage $\mathcal{P}$ and $\mathcal{F}$ to model enhanced contextual dependencies, achieving strong expressive power.

Moreover, we modernize the network architecture to improve local context representations, which consists of three aspects. 1) *Spatial Proximity:* We introduce a simple yet efficient Point Serialization Strategy (PSS), referring to serialization-based methods Liu et al. (2023); Wang (2023); Wu et al. (2024), to tackle the challenge of receptive field scaling caused by treating point clouds as unordered sets. It organizes unordered point clouds into structured sequences following specific patterns while retaining certain spatial proximity. Surprisingly, as an alternative to the traditional k-NN, PSS expands the receptive field from a small constant (e.g., 16 points) to 1024 points. 2) *Raw-point Embedding:* We advocate aggregating local neighbor features for each point during raw-point embedding to achieve informative feature initialization. Hence, we utilize sparse convolution to project input points into a latent space, leveraging its capability of encoding spatial priors while maintaining computational and memory efficiency. As illustrated in Figure 2, this approach enables faster convergence and achieves superior performance. 3) *Pooling Strategy:* To address the spatial misalignment caused by prevalent pooling strategies based on the sampling-grouping paradigm Qi et al. (2017), we employ partition-based pooling Wu et al. (2022) to summarize local geometric and semantic contexts. It operates on non-overlapping partitions obtained by dividing the point cloud and aggregates point features within each partition. Table 6 shows the performance gain, indicating that this approach preserves more faithful information.

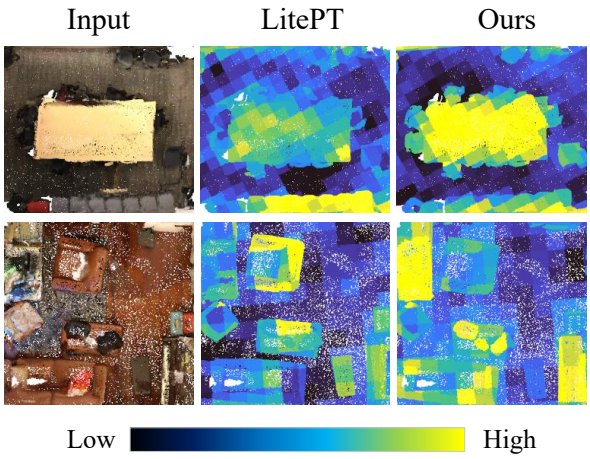


Figure 1: Visualization of feature response scores on ScanNet v2. Compared with the state-of-the-art network LitePT Yue et al. (2026), our LoCoFormer exhibits more concentrated and structured feature responses that align with the target semantics.

Compared with the state-of-the-art method, our LoCoFormer exhibits more discriminative feature responses, benefiting from enhanced local context learning, as shown in Figure 1. In summary, our main contributions include:

- We propose Bilateral Context Attention (BCA), which adopts a dual-branch design with a bilateral fusion mechanism to effectively utilize local spatial structures and implicit features for establishing enhanced contextual dependencies.

- To improve the representation and propagation of local contexts, we modernize the network architecture from three aspects: *Spatial Proximity*, *Raw-point Embedding*, and *Pooling Strategy*, which can enlarge receptive field size, enable informative feature initialization, and mitigate spatial misalignment, respectively.
- We present LoCoFormer for comprehensive local geometric and semantic representation in 3D point cloud analysis. Extensive experiments demonstrate that our method achieves superior performance across multiple downstream task benchmarks, including ModelNet40, S3DIS, ScanNet v2, and ScanNet200.

## 2 Related Work

### 2.1 MLP-based Methods

As pioneering works, PointNet Charles et al. (2017) and PointNet++ Qi et al. (2017) directly encode raw point clouds using shared multi-layer perceptrons (MLPs) and aggregate features through symmetric functions. To enhance local feature representations, numerous studies on point convolution Thomas et al. (2019); Xu et al. (2021); Thomas et al. (2024) have been proposed. PointConv Wu et al. (2019) learns weight functions through MLPs and incorporates density estimation to process point information. PAConv Xu et al. (2021) dynamically constructs convolution kernels to capture local features and geometries. To exploit the full potential of KPConv Thomas et al. (2019), KPConvX Thomas et al. (2024) introduces a scaling mechanism and kernel attention into kernel point convolutions. In addition, several works focusing on training and architectural modernization have also emerged. PointNeXt Qian et al. (2022) revisits PointNet++, adopting a series of improved training strategies and adding inverted residual MLP blocks to the framework. PointMLP Ma et al. (2022) proposes geometric affine modules in conjunction with residual point MLP blocks to deliver competitive performance. DeepLA-Net Zeng et al. (2025) presents a residual local feature extraction block to enable very deep networks while ensuring computational efficiency, and introduces a hybrid deep supervision strategy to facilitate smooth gradient backpropagation.

### 2.2 Point Cloud Transformers

Point Transformer Zhao et al. (2021) introduces vector self-attention and learnable positional encoding to model semantic information. To respect the permutation invariance of input points, PCT Guo et al. (2021) proposes an offset-attention to capture long-range context. PTv2 Wu et al. (2022) presents group vector attention and partition-based pooling to achieve more expressive feature representations. In contrast, PTv3 Wu et al. (2024) focuses on the power of scaling, introducing a serialization method and patch attention. However, these methods generate queries, keys, and values using simple linear projections without modeling local geometric patterns. To alleviate this issue, several works have been proposed. ConDaFormer Duan et al. (2023) introduces a local structure enhancement strategy before and after the attention layer to enrich local geometric representations. GPSFormer Wang et al. (2024) proposes a Local Structure Fitting Convolution to encode local geometric patterns. SP$^2$T Wan et al. (2025) utilizes a local proxy-based dual-stream Transformer to balance local-global feature fusion. Although some progress has been made, existing approaches are still inadequate for capturing comprehensive local contexts. In this work, we focus on developing a delicate attention mechanism and modernizing the network architecture to achieve desired local context representation.

## 3 Methods

### 3.1 Revisiting Point Cloud Transformers

Given a point cloud containing $N$ points, it can be described from two aspects: 1) the intrinsic coordinates $\mathcal{P} = \{p_i | i = 1, 2, ..., N\} \in \mathbb{R}^{N\times 3}$ in a Cartesian space, which indicate the geometric context of points. 2) the corresponding feature map $\mathcal{F} = \{f_i | i = 1, 2, ..., N\} \in \mathbb{R}^{N\times C}$ in $C$-dimensional feature space, which implies the clues about semantic context. From this perspective, jointly leveraging $\mathcal{P}$ and $\mathcal{F}$ is essential for achieving discriminative point cloud representation.

Benefiting from the ability of modeling long-range dependencies, point cloud Transformers Duan et al. (2023); Wu et al. (2024); Wan et al. (2025) have achieved remarkable progress in 3D point cloud

analysis. Let $\mathcal{X} = \{\mathcal{P}, \mathcal{F}\}$ denote a point set fed into the Transformer block. The pivotal Softmax attention Vaswani et al. (2017) operating on point $x_i \in \mathcal{X}$ can be represented as follows:

$$Q = W_q\mathcal{F}, K = W_k\mathcal{F}, V = W_v\mathcal{F}, \tag{1}$$

$$Y_i^{attn} = \text{Softmax}(Q_i(K_{\mathcal{N}_i})^T/\sqrt{D})V_{\mathcal{N}_i}, \tag{2}$$

where $W_q$, $W_k$, $W_v \in \mathbb{R}^{C\times D}$ represent linear projection matrices, $Q$, $K$, $V \in \mathbb{R}^{N\times D}$ denote the query, key, and value, and $\mathcal{N}_i \in \mathbb{R}^M$ indicates the indices of the neighbors at point $x_i$. However, conventional Softmax attention is inherently limited in modeling intricate local geometric patterns. To be specific, the query, key, and value are obtained solely through linear projections, lacking explicit interactions for capturing point-pair relationships within a local region. Moreover, it only considers $\mathcal{F}$, which encodes semantic context, while ignoring $\mathcal{P}$, which retains basic geometric priors. As a remedy, existing methods usually introduce additional positional encoding $\delta$ Zhao et al. (2021); Park et al. (2022); Wan et al. (2025), as formulated below.

$$\widetilde{Y}_i^{attn} = \text{Softmax}(Q_i(K_{\mathcal{N}_i})^T/\sqrt{D} + \delta)V_{\mathcal{N}_i}. \tag{3}$$

Nevertheless, this approach still fails to fully exploit $\mathcal{P}$ and $\mathcal{F}$ for comprehensive context learning due to the simple element-wise addition.

Beyond the attention mechanism, we observe that the architectural design of point cloud Transformers also hinders effective modeling of local geometric and semantic contexts, which mainly stems from three critical aspects. 1) *Spatial Proximity:* Considering the unstructured nature of point clouds, traditional spatial proximity Zhao et al. (2021); Duan et al. (2023); Yang et al. (2025) is defined based on k-Nearest Neighbors (k-NN), which can be formulated as:

$$\mathcal{N}_i^k = \text{KNN}(x_i, \mathcal{X}) = \{x_{i_1}, x_{i_2}, ..., x_{i_k}\}, \tag{4}$$

where $x_{i_j}$ denotes the $j$-th nearest neighbor of $x_i$ measured by Euclidean distance. Although k-NN constructs precise spatial distributions, its efficiency constraint limits the range of receptive fields to a small constant (e.g., 16 points). Since the receptive field size largely determines the local information acquisition, such restricted neighbors prevent the network from utilizing the power of scale to perceive more abundant geometric and semantic contexts. 2) *Raw-point Embedding:* The prevailing raw-point embedding Guo et al. (2021); Wu et al. (2022); Wan et al. (2025) employs MLPs to project input points into a latent space. However, we empirically observe that this design leads to relatively slow convergence and poor performance, as illustrated in Figure 2. Such point-wise embedding merely encodes individual point information without incorporating local geometric or semantic context. As a result, the attention map calculated by the first Transformer block struggles to establish fine-grained relevance between queries and keys, compromising the representation capability of the network. 3) *Pooling Strategy:* Existing pooling strategies typically follow a sampling-grouping paradigm Qi et al. (2017), which suffers from spatial misalignment since the point density and overlap among different grouping sets are not controllable. This results in the aggregated local context being either incomplete or redundant, and such inconsistency is progressively accumulated across hierarchical stages of the network.

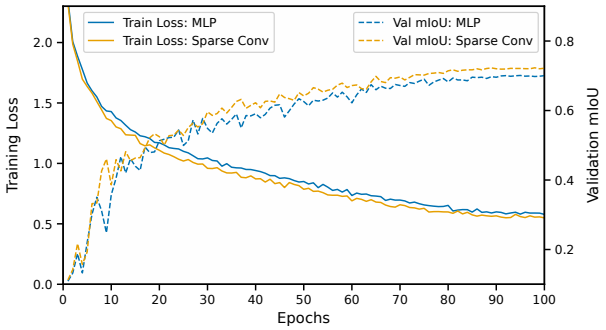


Figure 2: Plots of training loss and validation mIoU on ScanNet v2. We vary only the raw-point embedding, i.e., using MLPs or sparse convolutions, while keeping all other settings identical for comparison.

### 3.2 Bilateral Context Attention

To address the limitations of conventional attention mechanism, we propose Bilateral Context Attention (BCA), which adopts a dual-branch design with a bilateral fusion mechanism, as illustrated in Figure 3 (b).

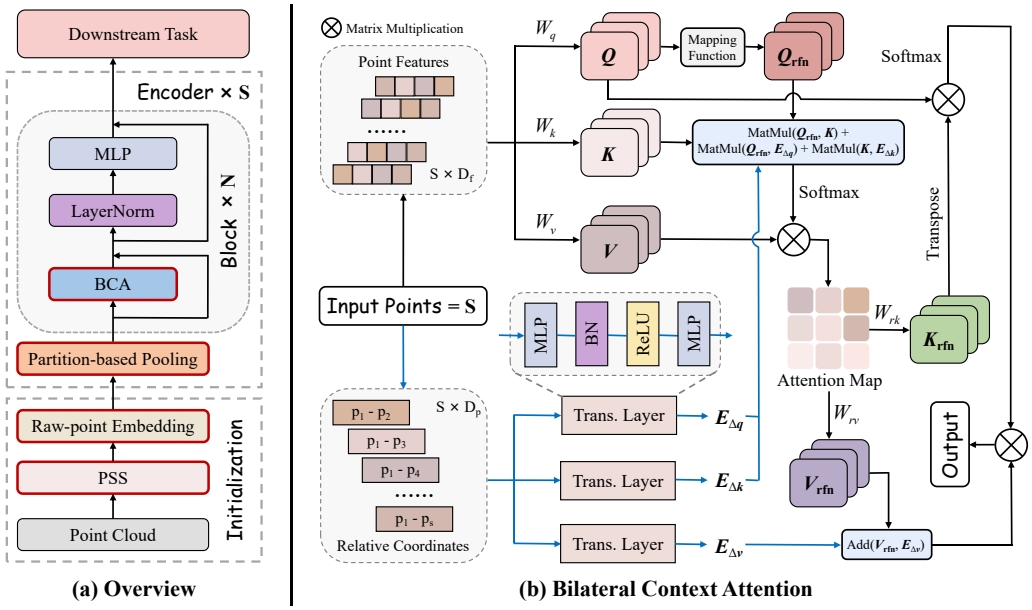


Figure 3: Overall architecture. (a) Framework overview of the LoCoFormer. We highlight our main contributions with red borders. (b) Illustration of the Bilateral Context Attention (BCA). Blue arrows denote the geometric context branch for spatial perception, black arrows indicate the semantic context branch for feature extraction, and light-blue boxes represent the bilateral fusion mechanism that facilitates interaction between geometric and semantic contexts.

**Geometric context branch.** For a patch containing $S$ points, let $\mathcal{P} \in \mathbb{R}^{S\times 3}$ and $\mathcal{F} \in \mathbb{R}^{S\times C}$ represent its coordinates and features. The geometric context branch first computes pairwise relative coordinate offsets to explicitly capture local spatial structures:

$$\Delta p_{ij} = p_i - p_j \in \mathbb{R}^3, \tag{5}$$

where $p_i, p_j \in \mathcal{P}$ are the $i$-th and $j$-th points. Rather than embedding each offset independently, we encode the complete spatial distribution by flattening the $S$ relative coordinate offsets associated with each point into geometric vectors $\Delta M_i$ and stacking them to form a relation matrix $\Delta M$, as formulated below.

$$\Delta M_i = \text{Flatten}\left([\Delta p_{ij}]_{j=1}^{S}\right), \quad \Delta M = \begin{bmatrix} \Delta M_1 \\ \vdots \\ \Delta M_S \end{bmatrix} \in \mathbb{R}^{S\times 3S}. \tag{6}$$

Then, the transformation layer constructed by MLP, Batch Normalization, and ReLU activation is employed to derive learnable geometric encoding, which can be formulated as follows.

$$E_{\Delta q} = L_q(\Delta M), \quad E_{\Delta k} = L_k(\Delta M), \quad E_{\Delta v} = L_v(\Delta M). \tag{7}$$

In particular, the weights of transformation layers for generating $E_{\Delta q}$, $E_{\Delta k}$, and $E_{\Delta v}$ are not shared, so we distinguish them by assigning different subscripts.

**Semantic context branch.** To capture underlying semantic cues, the semantic context branch employs two queries to aggregate features across different scales. To be specific, it first derives the vanilla query $Q$, key $K$, and value $V$ by projecting the input features:

$$Q = W_q\mathcal{F}, K = W_k\mathcal{F}, V = W_v\mathcal{F}, \tag{8}$$

where $W_q, W_k, W_v \in \mathbb{R}^{C\times D}$ indicate projection matrices. Then, a mapping function is introduced to generate the refined query $Q_{\text{rfn}}$, which summarizes representative features from the vanilla query $Q$. We perform attention calculations in two successive stages to progressively aggregate fine-grained features. For the $h$-th attention head, it is formulated as:

$$Attn^h_{\text{rfn}} = \text{Softmax}(Q^h_{\text{rfn}}(K^h)^T/\sqrt{D/H})V^h, \tag{9}$$

$$Attn^h_{\text{out}} = \text{Softmax}(Q^h(W^h_{rk}Attn^h_{\text{rfn}})^T/\sqrt{D/H})(W^h_{rv}Attn^h_{\text{rfn}}), \tag{10}$$

where $H$ is the number of attention heads, and $W^h_{rk}$ and $W^h_{rv}$ are projection matrices for generating $K^h_{\text{rfn}}$ and $V^h_{\text{rfn}}$, respectively. In practice, the refined query $Q_{\text{rfn}}$ can be obtained through various mapping functions, such as point convolution Thomas et al. (2024) or graph convolution Zhou et al. (2021). To ensure the efficiency of BCA, we adopt max pooling as the mapping function, which performs surprisingly well.

**Bilateral fusion mechanism.** For the fusion of geometric and semantic contexts, an intuitive approach is to directly add the outputs of two branches. However, such a naive fusion strategy fails to fully facilitate the interaction between geometric structures and semantic features. To address this issue, we propose a bilateral fusion mechanism inspired by contextual mode Wu et al. (2021). The bilateral fusion mechanism leverages learnable geometric encodings to modulate attention weights induced by refined queries $Q_{\text{rfn}}$ and injects geometric priors into the refined value $V_{\text{rfn}}$. Formally, this process in the $h$-th attention head can be expressed as:

$$\Delta P^h = Q^h_{\text{rfn}}(\Delta E^h_q)^T + K^h(\Delta E^h_k)^T, \tag{11}$$

$$Attn^h_{\Delta\text{rfn}} = \text{Softmax}((Q^h_{\text{rfn}}(K^h)^T + \Delta P^h)/\sqrt{D/H})(V^h), \tag{12}$$

$$Attn^h_{\Delta\text{out}} = \text{Softmax}(Q(W^h_{rk}Attn^h_{\Delta\text{rfn}})^T/\sqrt{D/H})(W^h_{rv}Attn^h_{\Delta\text{rfn}} + \Delta E^h_v). \tag{13}$$

Consequently, the geometric and semantic branches provide local context representations with complementary knowledge, while the bilateral fusion mechanism can effectively exploit geometric structures $\mathcal{P}$ and semantic features $\mathcal{F}$ to model contextual dependencies.

### 3.3 Architecture Modernization

In this subsection, we focus on architectural modernization to improve local geometric and semantic context representation, thereby further enhancing the generalization capability of our network. The modernization encompasses three key aspects, as detailed below.

**Spatial Proximity.** Inspired by serialization-based approaches Liu et al. (2023); Wang (2023); Wu et al. (2024), we move beyond the traditional paradigm of treating point clouds as unordered sets and introduce a Point Serialization Strategy (PSS) as an alternative to k-NN for defining spatial proximity. Referring to PTv3 Wu et al. (2024), we employ Z-order curve, Hilbert curve, and their variants to process the 3D space by following different orders of traversal along the x, y, and z axes, organizing point clouds into structured sequences. These space-filling curves provide diverse perspectives on spatial distributions of points, potentially capturing local structures of the point cloud. Benefiting from the efficiency of PSS, we can expand the receptive field from a small constant (e.g., 16 points) to 1024 points. In this manner, our network is able to flexibly scale the range of perception to acquire sufficient geometric and semantic contexts.

**Raw-point Embedding.** To achieve informative feature initialization, we try various approaches for raw-point embedding and find that sparse convolution is an effective choice, as demonstrated in Table 7. In particular, sparse convolution is expressive because it aggregates features from spatially neighboring points. This enables the initial feature representation to encode local geometric and semantic contexts. In addition, sparse convolution is efficient and fast. It inherently saves memory and computation for 3D scans where most of the space is empty, and only computes outputs for predefined coordinates and stores them as a compact sparse tensor. Consequently, the raw-point embedding constructed by sparse convolutions serves as a gateway for learning local spatial relationships, injecting local spatial priors into the subsequent Transformer block.

**Pooling Strategy.** We adopt partition-based pooling Wu et al. (2022) to aggregate local geometric and semantic contexts, motivated by its advantages in spatial alignment and flexible grid sizing. Partition-based pooling summarizes point features within each non-overlapping partition formed by dividing the point cloud. Unlike conventional pooling strategies based on the sampling-grouping paradigm Qi et al. (2017), it avoids information loss or duplication in the grouping set. Meanwhile, partition-based pooling can increase the grid size to reduce memory overhead while hardly affecting the quality of aggregated features. As a result, partition-based pooling produces desired local feature representations while retaining memory efficiency.

### 3.4 FRAMEWORK OF LOCOFORMER

Based on BCAs and modernized architectural designs, we propose LoCoFormer, a framework that can effectively capture comprehensive geometric and semantic contexts, whose architecture is illustrated in Figure 3 (a).

**Backbone structure.** During the initialization phase, we first leverage PSS to organize point clouds in specific patterns, forming a structured sequence. Then, the raw-point embedding constructed by sparse convolutions is employed to project serialized points into a latent space. The encoder in LoCoFormer has five stages, where each stage consists of a partition-based pooling followed by a stack of blocks. Specifically, the grid size multiplier is 2, indicating the expansion ratio relative to the previous pooling stage. The block adopts a residual structure comprising BCA, Layer Normalization, and MLP.

**Output head.** For shape classification, we use global average pooling to aggregate the point-wise features produced by the encoder into a global feature vector, followed by an MLP classifier for prediction. The classifier consists of linear layers interleaved with Batch Normalization and ReLU activation, progressively mapping the dimension of the global feature vector to the number of categories. For semantic segmentation, consistent with previous work Zhao et al. (2021); Duan et al. (2023); Wu et al. (2024), we adopt a U-Net architecture to gradually upsample the features from the encoding stage. Subsequently, an MLP is applied to perform point-wise label predictions.

## 4 EXPIREMENTS

To evaluate our LoCoFormer, we conduct experiments on shape classification and semantic segmentation tasks. We also perform extensive ablation studies to analyze our designs in LoCoFormer, and the experimental results are derived from S3DIS unless otherwise stated. More experimental details and results are available in the Appendix.

### 4.1 SHAPE CLASSIFICATION

For shape classification, we evaluate our LoCoFormer on a widely used public benchemark, ModelNet40 Wu et al. (2015). In Table 1, we report the quantitative results of LoCoFormer against state-of-the-art methods. It is observed that LoCoFormer outperforms the representative model Point Transformer, improving mAcc and OA by 0.7% and 0.5%, respectively. Compared with PointLearner, which adopts a robust attention block, our LoCoFormer still achieves competitive performance. These results demonstrate that enhancing local context learning can facilitate the shape recognition of point clouds.

Table 1: Shape classification results (%) on ModelNet40.

| Method | Input | mAcc | OA |
|---|---|---|---|
| PointNet (Charles et al., 2017) | points | 86.2 | 89.2 |
| PointNet++ (Qi et al., 2017) | points | - | 91.9 |
| DGCNN Wang et al. (2019) | points | 90.2 | 92.9 |
| PointWeb Zhao et al. (2019) | points | 89.4 | 92.3 |
| AdaptConv Zhou et al. (2021) | points | 90.7 | 93.4 |
| Point Transformer Zhao et al. (2021) | points | 90.6 | 93.7 |
| PointNeXt Qian et al. (2022) | points | 90.8 | 93.2 |
| LCPFormer Huang et al. (2023) | points | 90.7 | 93.6 |
| PointMamba Liang et al. (2024) | points | - | 93.6 |
| HydraMamba Qu et al. (2025) | points | - | 94.0 |
| PCM Zhang et al. (2025) | points | 90.7 | 93.4 |
| SAMBLE Wu et al. (2025a) | points | - | **94.2** |
| PointGST Liang et al. (2025) | points | - | 94.1 |
| DC-CCNN Dang et al. (2026) | points | - | 93.5 |
| PCQ Li et al. (2026) | points | - | 94.1 |
| PointLearner Qu et al. (2026) | points | - | **94.2** |
| LoCoFormer | points | **91.3** | **94.2** |

### 4.2 SEMANTIC SEGMENTATION

For semantic segmentation, we conduct experiments on three challenging benchmark: S3DIS Armeni et al. (2016), ScanNet v2 Dai et al. (2017), and ScanNet200 Rozenberszki et al. (2022), as reported in Table 2. Compared with the highly competitive PTv3, LoCoFormer achieves performance gains of 1.1%, 0.7%, and 1.3% on S3DIS, ScanNet v2, and ScanNet200, respectively. Additionally, compared to recent works such as LitePT and Volt-S, LoCoFormer outperforms LitePT by 1.4% and Volt-S by 1.0% on ScanNet v2. Figure 4 and Figure 5 present the qualitative results of LoCoFormer on S3DIS and ScanNet v2, respectively. The visualizations show that the semantic prediction of LoCoFormer is

Table 2: Semantic segmentation results (%) on S3DIS, ScanNet v2, and ScanNet200.

| Method | Input | S3DIS | | ScanNet v2 | ScanNet200 |
|---|---|---|---|---|---|
| | | Area5 | 6-fold | mIoU | mIoU |
| MinkUNet Choy et al. (2019) | voxels | 65.4 | 65.4 | 72.2 | 25.0 |
| KPConv Thomas et al. (2019) | points | 67.1 | 70.6 | 69.2 | - |
| Point Transformer Zhao et al. (2021) | points | 70.4 | 65.4 | 70.6 | 27.8 |
| Stratified Transformer Lai et al. (2022) | points | 72.0 | - | 74.3 | - |
| PointNeXt Qian et al. (2022) | points | 70.5 | 74.9 | 71.5 | - |
| PTv2 Wu et al. (2022) | points | 71.6 | 73.5 | 75.4 | 30.2 |
| OctFormer Wang (2023) | points | - | - | 75.7 | 32.6 |
| PointMetaBase Lin et al. (2023) | points | 72.0 | 77.0 | 72.8 | - |
| ConDaFormer Duan et al. (2023) | points | 72.6 | - | 75.1 | 32.3 |
| PTv3 Wu et al. (2024) | points | 73.4 | 77.7 | 77.5 | 35.2 |
| AMContrast3D++ Chen et al. (2025) | points | 71.6 | - | 71.8 | - |
| LAST-PCL Wu et al. (2025b) | points | 71.3 | - | 73.3 | 33.3 |
| Swin3D Yang et al. (2025) | voxels | 72.5 | 76.9 | 76.4 | - |
| Inv Han et al. (2025) | points | 73.1 | - | 76.4 | - |
| RCSP Han et al. (2026) | points | 73.1 | - | 75.9 | - |
| PointLearner Qu et al. (2026) | points | 74.3 | - | - | - |
| LitePT Yue et al. (2026) | points | - | - | 76.8 | - |
| Volt-S Yilmaz et al. (2026) | points | - | - | 77.2 | 36.2 |
| LoCoFormer | points | **74.5** | **79.4** | **78.2** | **36.5** |

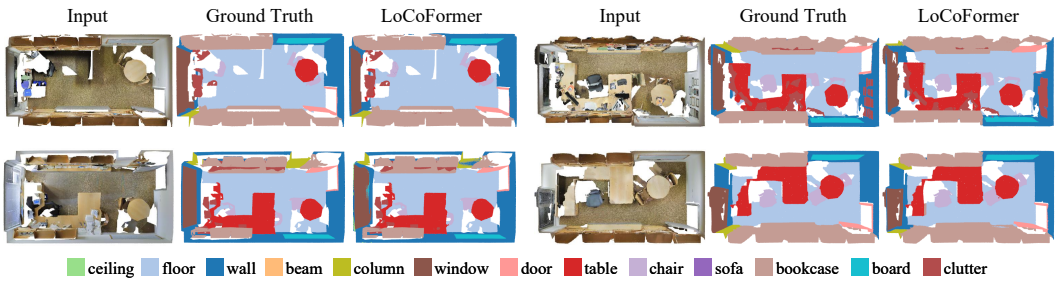


Figure 4: Visualization of semantic segmentation results on S3DIS.

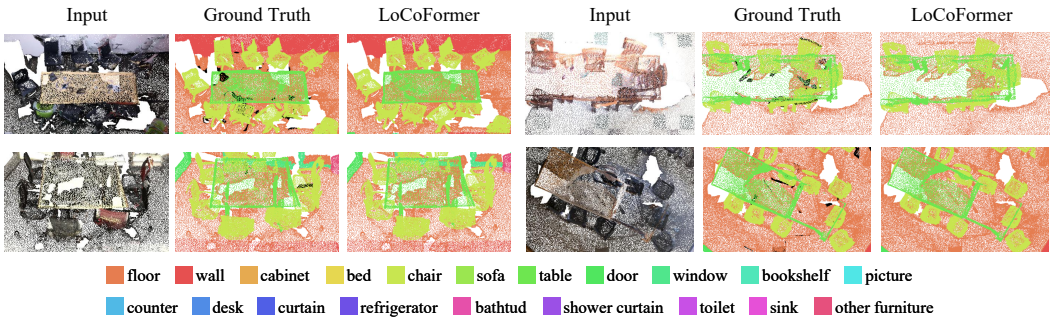


Figure 5: Visualization of semantic segmentation results on ScanNet v2.

extremely close to the ground truth. These results indicate the potential of exploring the interaction between local geometric and semantic contexts in point cloud Transformers.

### 4.3 Ablation Study

**Comparison with conventional Transformer block.** To show the effectiveness of BCA, we conduct experiments and report the results in Table 3. We combine the conventional Softmax attention with recent advanced positional encoding approaches, including RPE, cRPE, CPE, and xCPE, to construct Transformer blocks for comparison. For a fair evaluation, we only replace the Transformer block

in our network while keeping all other settings the same. It is observed that our method achieves consistent performance improvements, with a maximum gain of 1.6%.

**BCA designs.** To maintain the efficiency of the proposed BCA, we explore several simple schemes of the mapping function on ModelNet40 and find that max pooling performs best, as shown in Table 4. Furthermore, we investigate the contributions of applying geometric and semantic interactions to the refined query $Q_{\mathrm{rfn}}$, vanilla key $K$, and refined value $V_{\mathrm{rfn}}$, as reported in Table 5. The results demonstrate that jointly introducing interactions into $Q_{\mathrm{rfn}}$, $K$, $V_{\mathrm{rfn}}$ achieves the optimal performance.

Table 3: Ablation of Transformer block types. Softmax Attn denotes the Softmax attention.

| Self-attention | Positional Encoding | mIoU |
|---|---|---|
| Softmax Attn | RPE | 72.9 |
| | cRPE | 73.4 |
| | CPE | 73.2 |
| | xCPE | 73.5 |
| Ours | Ours | **74.5** |

Table 4: Ablation of the mapping function in BCA. Max and Avg denote pooling operations.

| Mapping Function | mAcc | OA |
|---|---|---|
| Conv1d | 90.6 | 93.6 |
| Max | **91.3** | **94.2** |
| Avg | 90.8 | 93.7 |
| Max & Avg | 91.2 | 93.9 |

**Architectural modernization.** In Table 6, we report the quantitative analysis of our Architecture Modernization. To directly evaluate the improvement brought by our method over traditional point cloud Transformers, we only adopt the widely used Softmax attention as the attention mechanism. We can observe that each architectural modernization approach consistently leads to performance improvements. When these approaches are combined, the network achieves a 2.7% performance gain in mIoU. In Table 7, we explore different schemes for raw-point embedding. It is observed that sparse convolution achieves the best performance.

Table 5: Ablation of the geometric and semantic interaction configuration in BCA.

| ID | $Q_{\mathrm{rfn}}$ | $K$ | $V_{\mathrm{rfn}}$ | mIoU |
|---|---|---|---|---|
| I | ✓ | | | 73.2 |
| II | | ✓ | | 73.0 |
| III | | | ✓ | 73.5 |
| IV | ✓ | ✓ | | 73.6 |
| V | ✓ | | ✓ | 74.1 |
| VI | | ✓ | ✓ | 74.0 |
| VII | ✓ | ✓ | ✓ | **74.5** |

Table 6: Quantitative analysis of our Architecture Modernization. We abbreviate Spatial Proximity, Raw-point Embedding, and Pooling Strategy as SP, RE, and PS, respectively.

| ID | SP | RE | PS | mIoU |
|---|---|---|---|---|
| I | | | | 70.8 |
| II | ✓ | | | 73.0 |
| III | | ✓ | | 72.2 |
| IV | | | ✓ | 71.2 |
| V | ✓ | ✓ | ✓ | **73.5** |

Table 7: Ablation of different raw-point embeddings. Softmax Attn and Sparse Conv denote the Softmax attention and sparse convolution, respectively.

| Raw-point Embedding | MLP | Pooling | Softmax Attn | Sparse Conv |
|---|---|---|---|---|
| mIoU | 74.1 | 71.6 | 73.2 | **74.5** |

## 5 Conclusion

In this work, we propose LoCoFormer to achieve comprehensive local geometric and semantic representation. LoCoFormer utilizes Bilateral Context Attention to capture local contexts with complementary knowledge and effectively facilitate their interaction, thereby establishing enhanced contextual dependencies. Moreover, an Architecture Modernization encompassing *Spatial Proximity*, *Raw-point Embedding*, and *Pooling Strategy* is introduced to respectively expand receptive field size, enable informative feature initialization, and alleviate spatial misalignment, resulting in improved local context representation. Quantitative and qualitative results on multiple challenging benchmarks show that our method consistently delivers performance improvements.

### AI USE STATEMENT

In this work, we used generative AI tools solely to assist with translation, including grammar and spelling checks. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, support qualitative and thematic data analysis, or interpret results. Generating synthetic data sets and cleaning or reformatting dataset are not applicable to this work. We take full responsibility for the final content of this submission.

## REFERENCES

Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1534–1543, 2016.

Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4413–4421. IEEE, 2018.

R Qi Charles, Hao Su, Mo Kaichun, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 77–85. Ieee, 2017.

Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding, reasoning, and planning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26418–26428. IEEE, 2024.

Yang Chen, Yueqi Duan, Haowen Sun, Jiwen Lu, and Yap-Peng Tan. Ambiguity-aware point cloud segmentation by adaptive margin contrastive learning. *IEEE Transactions on Multimedia*, 2025.

Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3075–3084, 2019.

Federico Cocchi, Nicholas Moratelli, Davide Caffagni, Sara Sarto, Lorenzo Baraldi, Marcella Cornia, and Rita Cucchiara. Llava-more: A comparative study of llms and visual backbones for enhanced visual instruction tuning. In *2025 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 4337–4347. IEEE, 2025.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 2432–2443. IEEE, 2017.

Jisheng Dang, Delin Deng, Bimei Wang, Jingze Wu, Hui Zhang, Haijiang Li, Jingmei Jiao, Dengyue Pan, Mangang Xie, and Jizhao Liu. Primary visual cortex inspired point cloud analysis framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 3506–3514, 2026.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Lunhao Duan, Shanshan Zhao, Nan Xue, Mingming Gong, Gui-Song Xia, and Dacheng Tao. Condaformer: Disassembled transformer with local structure enhancement for 3d point cloud understanding. *Advances in Neural Information Processing Systems*, 36:23886–23901, 2023.

Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational visual media*, 7(2):187–199, 2021.

Juhyung Ha, Vibhas Kumar Vats, Soon-heung Jung, Alimoor Reza, and David J Crandall. Hvpunet: hybrid-voxel point-cloud upsampling network. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 29153–29162. IEEE, 2025.

Jiawei Han, Kaiqi Liu, Wei Li, Feng Zhang, and Xiang-Gen Xia. Generating inverse feature space for class imbalance in point cloud semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Jiawei Han, Kaiqi Liu, Wei Li, and Musen Lin. Accelerating training convergence for point cloud semantic segmentation of large-scale urban scenes with scene-ensemble prototypes. *IEEE Transactions on Geoscience and Remote Sensing*, 2026.

Xun Huang, Ziyu Xu, Hai Wu, Jinlong Wang, Qiming Xia, Yan Xia, Jonathan Li, Kyle Gao, Chenglu Wen, and Cheng Wang. L4dr: Lidar-4dradar fusion for weather-robust 3d object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 3806–3814, 2025.

Zhuoxu Huang, Zhiyou Zhao, Banghuai Li, and Jungong Han. Lcpformer: Towards effective 3d point cloud analysis via local context propagation in transformers. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4985–4996, 2023.

Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 8490–8499. IEEE, 2022.

Hongxuan Li, Wencheng Zhu, Huiying Xu, Xinzhong Zhu, and Pengfei Zhu. Point cloud quantization through multimodal prompting for 3d understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 6145–6153, 2026.

Dingkang Liang, Xin Zhou, Wei Xu, Xingkui Zhu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, and Xiang Bai. Pointmamba: A simple state space model for point cloud analysis. *Advances in neural information processing systems*, 37:32653–32677, 2024.

Dingkang Liang, Tianrui Feng, Xin Zhou, Yumeng Zhang, Zhikang Zou, and Xiang Bai. Parameter-efficient fine-tuning in spectral domain for point cloud learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Haojia Lin, Xiawu Zheng, Lijiang Li, Fei Chao, Shanshan Wang, Yan Wang, Yonghong Tian, and Rongrong Ji. Meta architecture for point cloud analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17682–17691, 2023.

Fengqi Liu, Jingyu Gong, Qianyu Zhou, Xuequan Lu, Ran Yi, Yuan Xie, and Lizhuang Ma. Cloudmix: Dual mixup consistency for unpaired point cloud completion. *IEEE Transactions on Visualization and Computer Graphics*, 31(4):2182–2195, 2024.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9992–10002. Ieee, 2021.

Zhijian Liu, Xinyu Yang, Haotian Tang, Shang Yang, and Song Han. Flatformer: Flattened window attention for efficient point cloud transformer. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1200–1211. IEEE, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *arXiv preprint arXiv:2202.07123*, 2022.

Chunghyun Park, Yoonwoo Jeong, Minsu Cho, and Jaesik Park. Fast point transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16928–16937. IEEE, 2022.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in neural information processing systems*, 35:23192–23204, 2022.

Kanglin Qu, Pan Gao, Qun Dai, and Yuanhao Sun. Hydramamba: Multi-head state space model for global point cloud learning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 333–342, 2025.

Kanglin Qu, Pan Gao, Qun Dai, and Yuanhao Sun. Point-focused attention meets context-scan state space: Robust biological visual perception for point cloud representation. In *International Conference on Learning Representations*, volume 2026, pp. 155663–155689, 2026.

David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *European conference on computer vision*, pp. 125–141. Springer, 2022.

Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pp. 369–386. SPIE, 2019.

Hang Su, Varun Jampani, Deqing Sun, Subhransu Maji, Evangelos Kalogerakis, Ming-Hsuan Yang, and Jan Kautz. Splatnet: Sparse lattice networks for point cloud processing. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 2530–2539. IEEE, 2018.

Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6410–6419. IEEE, 2019.

Hugues Thomas, Yao-Hung Hubert Tsai, Timothy D Barfoot, and Jian Zhang. Kpconvx: Modernizing kernel point convolution with kernel attention. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5525–5535. IEEE, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Jiaxu Wan, Hong Zhang, Ziqi He, Yangyan Deng, Qishu Wang, Ding Yuan, and Yifan Yang. Sp 2 t: Sparse proxy attention for dual-stream point transformer. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27885–27895. IEEE, 2025.

Changshuo Wang, Meiqing Wu, Siew-Kei Lam, Xin Ning, Shangshu Yu, Ruiping Wang, Weijun Li, and Thambipillai Srikanthan. Gpsformer: A global perception and local structure fitting-based transformer for point cloud understanding. In *European conference on computer vision*, pp. 75–92. Springer, 2024.

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

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38(5): 1–12, 2019.

Chengzhi Wu, Yuxin Wan, Hao Fu, Julius Pfrommer, Zeyun Zhong, Junwei Zheng, Jiaming Zhang, and Jürgen Beyerer. Samble: Shape-specific point cloud sampling for an optimal trade-off between local detail and global uniformity. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1342–1352. IEEE, 2025a.

Kan Wu, Houwen Peng, Minghao Chen, Jianlong Fu, and Hongyang Chao. Rethinking and improving relative position encoding for vision transformer. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 10013–10021. IEEE, 2021.

Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9613–9622. Ieee, 2019.

Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. *Advances in neural information processing systems*, 35:33330–33342, 2022.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4840–4851. IEEE, 2024.

Yanmin Wu, Qiankun Gao, Renrui Zhang, Haijie Li, and Jian Zhang. Language-assisted 3d scene understanding. *IEEE Transactions on Multimedia*, 27:3869–3879, 2025b.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1912–1920, 2015.

Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation. In *European Conference on Computer Vision*, pp. 1–19. Springer, 2020.

Mutian Xu, Runyu Ding, Hengshuang Zhao, and Xiaojuan Qi. Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds. *arXiv preprint arXiv:2103.14635*, 2021.

Yu-Qi Yang, Yu-Xiao Guo, Jian-Yu Xiong, Yang Liu, Hao Pan, Peng-Shuai Wang, Xin Tong, and Baining Guo. Swin3d: A pretrained transformer backbone for 3d indoor scene understanding. *Computational Visual Media*, 11(1):83–101, 2025.

Kadir Yilmaz, Adrian Kruse, Tristan Höfer, Daan de Geus, and Bastian Leibe. Volume transformer: Revisiting vanilla transformers for 3d scene understanding. *arXiv preprint arXiv:2604.19609*, 2026.

Yuanwen Yue, Damien Robert, Jianyuan Wang, Sunghwan Hong, Jan Dirk Wegner, Christian Rupprecht, and Konrad Schindler. Litept: Lighter yet stronger point transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24173–24184, 2026.

Ziyin Zeng, Mingyue Dong, Jian Zhou, Huan Qiu, Zhen Dong, Man Luo, and Bijun Li. Deepla-net: Very deep local aggregation networks for point cloud analysis. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1330–1341. IEEE, 2025.

Tao Zhang, Haobo Yuan, Lu Qi, Jiangning Zhang, Qianyu Zhou, Shunping Ji, Shuicheng Yan, and Xiangtai Li. Point cloud mamba: Point cloud learning via state space model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10121–10130, 2025.

Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5565–5573, 2019.

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16239–16248. IEEE, 2021.

Haoran Zhou, Yidan Feng, Mingsheng Fang, Mingqiang Wei, Jing Qin, and Tong Lu. Adaptive graph convolution for point cloud analysis. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4965–4974, 2021.

# A Appendix

## A.1 Datasets and Metrics

ModelNet40 Wu et al. (2015) is a large-scale collection of manufactured point clouds, consisting of 12,311 CAD models across 40 categories. Following a common protocol, we allocate 9,843 models for training and the remaining 2,468 models for testing. Following standard practice in the community, we utilize the mean of class accuracy (mAcc) and overall accuracy (OA) for shape classification evaluation. S3DIS Armeni et al. (2016) for semantic scene parsing comprises 271 rooms across six areas from three buildings, where each point are densely sampled on mesh surfaces and annotated with 13 categories. Adhering to the official protocol, we withhold area 5 for testing while using the remaining areas for training, and also perform a 6-fold cross-validation. ScanNet v2 Dai et al. (2017) contains 1,513 room scans reconstructed form RGB-D frames. We follow the official train-test split, where 1,201 scenes are used for training and 312 scenes for testing. ScanNet200 Rozenberszki et al. (2022) shares the same input point clouds as ScanNet v2 but extends the annotation labels from 20 to 200 categories, encompassing fine-grained object classes as well as previously unrepresented categories. It follows the original data split of ScanNet v2 for training and testing. In accordance with standard practice in the community, we utilize the mean class-wise intersection over union (mIoU) as the evaluation metric for semantic segmentation.

Table A: Shape classification settings.

| Config | Value |
|---|---|
| optimizer | AdamW Loshchilov & Hutter (2017) |
| scheduler | OneCycleLR Smith & Topin (2019) |
| criteria | CrossEntropy<br>Lovasz Berman et al. (2018) |
| lr rate | 1e-3 |
| block lr rate | 1e-4 |
| weight decay | 1e-2 |
| batch size | 16 |
| epochs | 300 |
| warmup epochs | 15 |

Table B: Semantic segmentation settings.

| Config | Value |
|---|---|
| optimizer | AdamW Loshchilov & Hutter (2017) |
| scheduler | OneCycleLR Smith & Topin (2019) |
| criteria | CrossEntropy<br>Lovasz Berman et al. (2018) |
| lr rate | 6e-3 |
| block lr rate | 6e-4 |
| weight decay | 5e-2 |
| batch size | 4 |
| epochs | 800 |
| warmup epochs | 40 |

## A.2 Implementation Details

**Model setting.** For shape classification, the depth of raw-point embedding is set to 1, and the embedding channel is 32. The channel dimensions of the five stages in our network are set to [C, 2C, 4C, 8C, 16C], where C is 32. The block depths are [1, 1, 1, 1, 1]. The receptive field sizes of the encoder are set to [1024, 1024, 1024, 1024, 1024]. The attention mechanism in all stages adopts the proposed BCA. For semantic segmentation, the raw-point embedding and the channel dimensions are set the same as in the classification task. The block depths are set to [2, 2, 2, 6, 2]. To save memory consumption and mitigate overfitting, the receptive field sizes of the encoder are configured to [128, 128, 1024, 1024, 1024]. We employ our BCA in the first two stages and leverage the commonly used Softmax attention in the remaining stages.

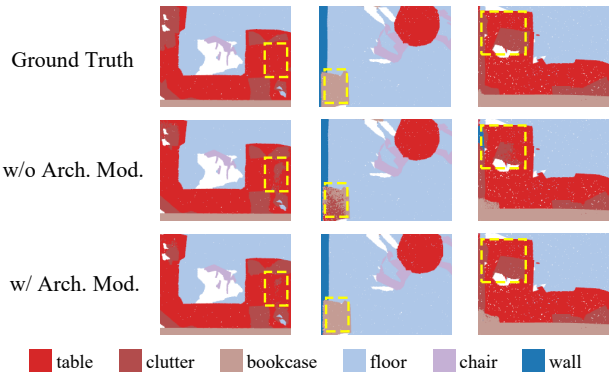


Figure A: Qualitative analysis of Architecture Modernization on S3DIS. w/ and w/o Arch. Mod. indicate our network with and without the proposed Architecture Modernization, respectively. The differences between these two settings are highlighted by yellow dashed boxes.

**Training setting.** All experiments are performed on 4 RTX 4090 GPUs unless otherwise specified. In accordance with common practice Wu et al. (2022; 2024); Yue et al. (2026), we downsample the point cloud into grids, where the grid sizes for shape classification and semantic segmentation are set to 0.01m and 0.02m, respectively. The detailed training configurations for shape classification are outlined in Table A. For semantic segmentation, we detail the training configurations in Table B.

Table C: Synergy analysis of our Architecture Modernization. We abbreviate Spatial Proximity, Raw-point Embedding, and Pooling Strategy as SP, RE, and PS, respectively.

| ID | SP | RE | PS | mIoU |
|---|---|---|---|---|
| I | | | | 70.8 |
| II | ✓ | ✓ | | 73.4 |
| III | ✓ | | ✓ | 73.2 |
| IV | | ✓ | ✓ | 72.7 |

Table D: Ablation of using different receptive field sizes in our network.

| Receptive Field Size | 16 | 32 | 64 | 128 | 256 | 1024 |
|---|---|---|---|---|---|---|
| mIoU | 72.3 | 72.8 | 73.4 | 73.6 | 73.9 | **74.5** |

Table E: Model efficiency analysis. The notation after "/" indicates the defined type of Spatial Proximity. S denotes the serialization-based method.

| Method | Param. | Latency | Memory | mIoU |
|---|---|---|---|---|
| PTv2 / k-NN Wu et al. (2022) | 12.8M | 146ms | 12.3G | 75.4 |
| PTv3 / S Wu et al. (2024) | 46.2M | 51ms | 4.1G | 77.5 |
| Swin3D / k-NN Yang et al. (2025) | 71.2M | 327ms | 8.7G | 76.4 |
| LoCoFormer / S | 46.5M | 67ms | 4.8G | **78.2** |

### A.3 More Experiment Results.

**Synergy analysis.** In Table C, we investigate the synergistic effects of different approaches in Architecture Modernization. It is observed that the joint use of these approaches consistently improves performance, demonstrating the effectiveness of our method.

**Receptive field size.** In Table D, we evaluate the impact of receptive field size on network performance. We scale the receptive field by adjusting the number of points in PSS-constructed patches and observe that using 1024 points yields a 2.2% improvement over 16 points.

**Visualization.** To visually demonstrate the effectiveness of Architecture Modernization, we present visualizations in Figure A. The qualitative results show that the network with Architecture Modernization delivers predictions that more closely align with the ground truth. We also present more qualitative results on semantic segmentation tasks, as shown in Figure B and C. Evidently, the predictions of LoCoFormer closely align with the ground truth, which demonstrates that our method is promising.

### A.4 Model Efficiency Analysis

We evaluate model efficiency on ScanNet v2 in terms of the parameters, latency, and memory overhead in a single inference. The results are measured on a single RTX 4090 GPU, excluding the first iteration to ensure stability. As shown in Table E, we select networks that adopt different spatial proximity definitions for comparison. It is observed that our LoCoFormer achieves a better trade-off between model efficiency and performance.

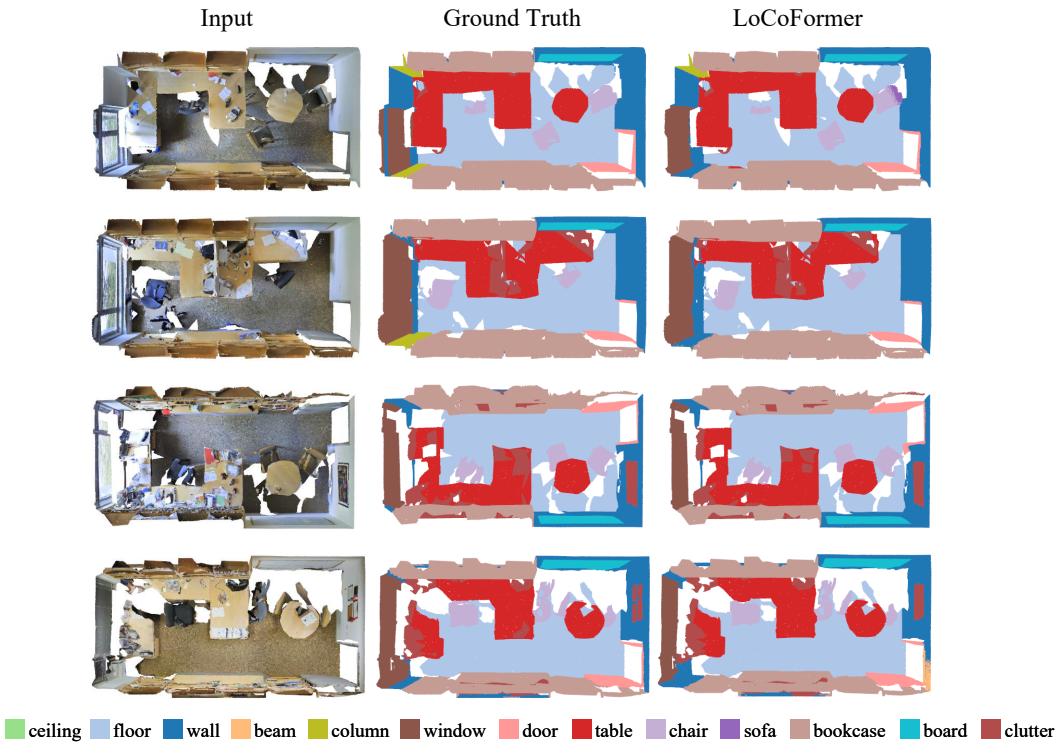


Figure B: Visualization of semantic segmentation results on S3DIS.

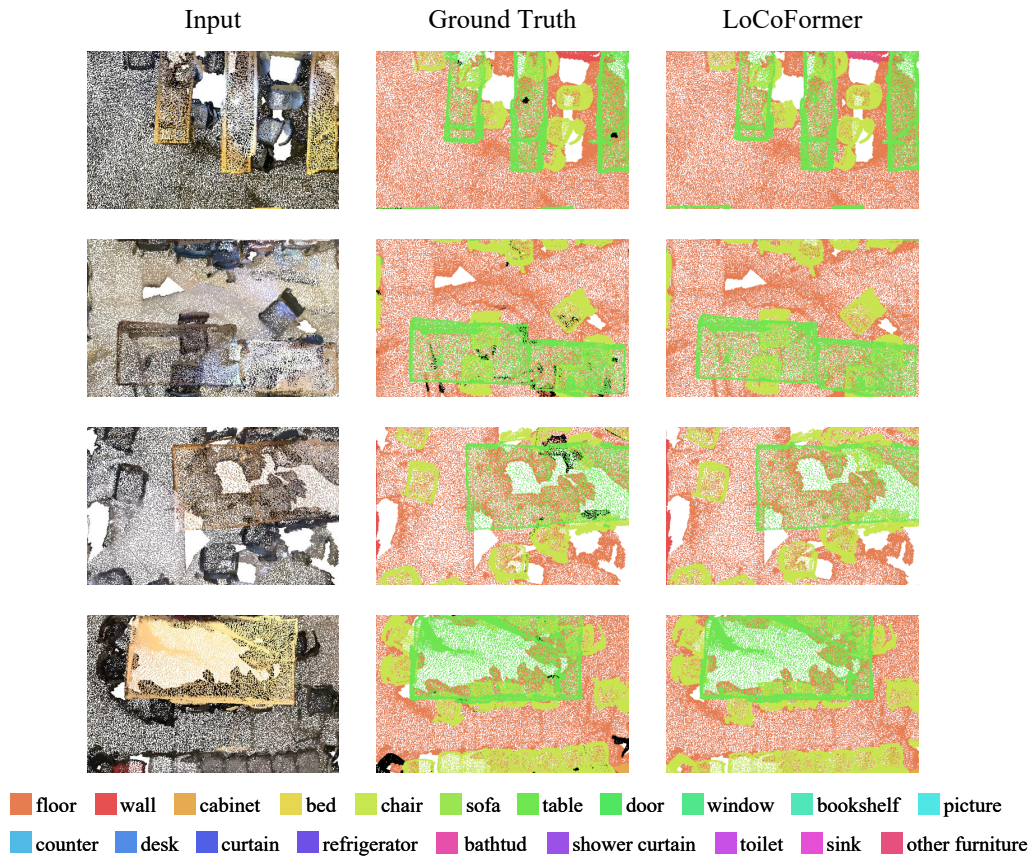


Figure C: Visualization of semantic segmentation results on ScanNet v2.