PointMC: Multi-view Consistent Encoding and Center-Global Feature Fusion for Point Clouds Understanding

Xinxing Yu[1], Ajian Liu[13*], Sunyuan Qiang[2], Yuzhong Wang[1], Hui Ma[14†], Yanyan Liang[1*]

[1]Faculty of Innovation Engineering, Macau University of Science and Technology, Macau, China
[2]Southwest Institute of Technical Physics, Chengdu, China
[3]MAIS, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[4]School of Computing and Information Technology, Great Bay University, Dongguan, China
{astrumyu, yzwangjoseph, mahuilight}@gmail.com, qiangsunyuan2025@163.com, ajian.liu@ia.ac.cn, yyliang@must.edu.mo

Abstract

Point cloud tasks have recently benefited from Mamba-based architecture, which leverage state space modeling to achieve strong performance. Previous studies have primarily focused on network design while overlooking the importance of position encoding and relying on coarse-grained geometric feature aggregation. The former leads to *semantic ambiguity* due to inconsistent spatial relationships, while the latter results in *geometric feature dispersion* by overlooking fine-grained local geometric details. To tackle the above problem, we propose a novel framework, PointMC, including Multi-view Consistent Learnable Position Encoding (MCLPE) and Center-Global Feature Fusion (CGFF), to provide semantically coherent positional guidance for inter-patch and enable fine-grained geometric structure aggregation within intra-patch regions. Specifically, the proposed MCLPE module is inspired by a spatial structure modeling mechanism guided by physical constraints, leverages multi-view virtual reconstruction and a learnable strategy to dynamically constrain spatial relationships along patch boundaries, thereby enhancing the semantic consistency and representational clarity across inter-patch regions. Furthermore, considering the lack of local structural information within each patch, the CGFF module employs a dual-guidance mechanism based on center and global structures to effectively promote the aggregation of local geometric features. Extensive experiments on multiple benchmark datasets validate the effectiveness of PointMC, consistently outperforming existing state-of-the-art methods, and demonstrating superior capability in capturing both inter-patch semantic consistency and intra-patch geometric details.

Introduction

Point clouds consist of numerous discrete 3D points with spatial coordinates (x, y, z) and attributes describing object surface geometry. Currently, point cloud has wide application (Liu et al. 2025a; Yan et al. 2025; Liu et al. 2025b; Yu et al. 2025b; Wang 2024). The sparsity, irregularity, and complexity of point clouds pose challenges for existing models to effectively capture geometric features, limiting their performance in 3D tasks.

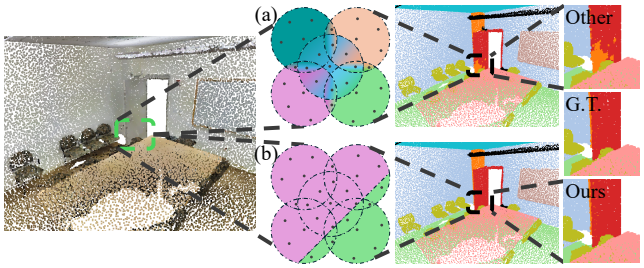

Figure 1: Comparison of position encoding. (a) Overlapping point patches receive multiple position encodings, hindering their ability to distinguish nearby points. (b) Our method effectively mitigates this overlap impact while accurately distinguishing between proximity points.

Mamba (Gu and Dao 2024) or transformer (Vaswani et al. 2017) based methods (Zhao et al. 2021; Wu et al. 2022; Han et al. 2024; Wang et al. 2024) have been proposed for point cloud processing. These methods divide point clouds into patches using Farthest Point Sampling (FPS) to select patch centers, followed by K-Nearest Neighbor grouping to form patches around each center. The aforementioned approaches use patch information as inputs for learnable positional encodings and coarse-grained local structure extractors, aiming to fulfill the demand for positional and local feature modeling. Nevertheless, current approaches exhibit two notable limitations: (1) semantic ambiguity arising from inconsistencies in spatial relationships governed by position encoding, and (2) the degradation of fine-grained spatial details due to insufficient modeling of local geometric information, which results in dispersed geometric features.

Semantic ambiguity from position encoding occur due to overlapping regions between patches. Position encoding of patch centers or patch points causes interference from multiple encodings within overlapping areas. An interesting phenomenon is observed, as shown in Figure 1(a): the non-unique construction of position encoding results in inconsistent modeling of inter-patch spatial relationships, which compromises the clarity of semantic boundaries and subsequently induces semantic ambiguity. Furthermore, employing absolute position encoding based on either patch centers

*Corresponding author
†With the assistance of Hui Ma's writing guidance.

or individual points not only impedes the inference of relative spatial relationships but also compromises the ability of the network to model inter-patch dependencies. Approaches such as PTV3 (Wu et al. 2024) and PointMamba (Liang et al. 2025) use methods that include the Hilbert curve or Z-order curve to serialize point clouds in an attempt to encode relative positional information, but these suffer from proximity distortion where spatially close points become distant in the sequence, degrading spatial locality. This occurs when two similar vectors are very close in the embedding space but are split at opposite ends of the sequence after mapping. However, degraded spatial locality prevents position encoding from achieving remote decay functionality.

Addressing the aforementioned issues entails tackling two key challenges: (1). The elimination of patch overlaps considerably reduces the number of spatial regions available for feature extraction, leading to lower information density and consequently constraining the representational capacity of neural networks in point cloud processing. (2). Due to the inherently unordered nature of point clouds, efficiently acquiring reliable relative positional information remains highly challenging, which fundamentally limits the applicability of classical position encoding schemes, such as RoPE (Su et al. 2024), in point cloud representation learning.

In addition, acquiring local geometric features is crucial for point cloud analysis. Existing methods (Zhang, Zhang, and Yan 2024; Zhang et al. 2025b) have designed sophisticated pre-training models and strategies for efficient local feature extraction, while (Li, Wang, and Xu 2024; Wang et al. 2025) developed dedicated neural architectures for processing point cloud features. Although these methods have achieved significant performance, their limited emphasis on local geometric feature extraction constrains their ability to model local topological relationships, thereby resulting in degradation of intra-patch spatial structural information.

To address previous challenges, we introduce a novel framework named PointMC, which incorporates Multi-view Consistent Learnable Position Encoding (MCLPE) and Center-Global Feature Fusion (CGFF) to enhance the semantic boundary clarity across inter-patch regions and improve the geometric structure awareness within intra-patch regions. Inspired by a spatial structure modeling mechanism guided by physical constraints, the MCLPE dynamically regulates inter-patch edge spatial relationships through multi-view virtual reconstruction and a learnable mechanism. This approach effectively models relative positions between patches while mitigating interference from redundant position encodings in overlapping regions in Figure 1(b). Specifically, MCLPE comprises a Multi-view Consistent Position Encoding (MCPE) module for generating initial spatial representations, and a Learnable Position Encoding (LPE) module for refining them with task-adaptive priors. MCPE first models the point cloud from multiple virtual viewpoints to capture consistent positional cues, and then leverages multi-view visibility as a criterion to construct position encodings. The LPE module employs a lightweight MLP-based learnable mechanism to model and adjust spatial relationships between patches during encoding, thereby enhancing the discriminability of semantic across patches.

The CGFF module effectively facilitates the aggregation of local geometric features through a dual-guidance mechanism leveraging both center and global structures, addressing the issue of missing local structural information within intra-patch regions. Specifically, the module performs separate feature mappings on the core region and the global context to extract local structural features and global semantic features, respectively. It then employs a lightweight gating mechanism to adaptively fuse these features through weighted combination, constructing the final feature representation and thereby enabling more effective extraction of structural information from individual patches.

The joint modeling of MCLPE and CGFF contributes to improved discriminability of semantic boundaries across inter-patch regions, while also enhancing the integrity and consistency of spatial feature representations intra-patch. Our key contributions can be summarized as follows:

- We propose Multi-view Consistency Learnable Position Encoding for point clouds, enabling relative position modeling between patches while reducing position semantic ambiguity from patch overlap.
- A Center-Global Feature Fusion is constructed for the purpose of efficiently capturing and extracting fine-grained features of local geometric structures.
- Extensive experiments on ScanObjectNN, ShapeNetPart, S3DIS and ModelNet40 confirm that PointMC consistently surpasses prior state-of-the-art (SOTA) methods.

Related Work

Supervised Learning for Point Cloud Analysis. This section focuses on point-based methods, which are directly relevant to our approach. These methods process raw point clouds without converting them into alternative representations, thereby avoiding information loss (Yu et al. 2025a). Recent studies (Zhang et al. 2022; Park et al. 2022; Ma et al. 2022; Chen et al. 2023; Yin et al. 2024; Li et al. 2024; Park et al. 2023; Deng et al. 2023; Zha et al. 2024; Zou et al. 2024; Bahri et al. 2025; Qu et al. 2025; Zhang et al. 2025a; Su et al. 2025; Wu et al. 2025b,a) primarily focus on designing efficient network modules to advance point cloud understanding. However, as model complexity increases, performance gains have diminished, underscoring the need for novel paradigms beyond conventional architectures.

Multi-view 3D Shape Analysis. Multi-view 3D shape analysis methods project 3D structures onto 2D planes and extract view-specific features using 2D CNNs, which are then fused into global descriptors for downstream tasks. Recent work (Wei, Yu, and Sun 2022; Hamdi, Giancola, and Ghanem 2023; Xu et al. 2024; Hamdi et al. 2025; Sun et al. 2024; Yu and Song 2024) leverage graph convolutions, attention mechanisms, and sequence modeling to capture spatial relationships among views and enhance shape discrimination. However, the sparse nature of point clouds limits the effectiveness of 2D projections due to low feature density. To address this, we propose MCLPE, which transfers the strengths of multi-view modeling such as robust feature representation and geometric relationship encoding into point cloud analysis, overcoming existing limitations.

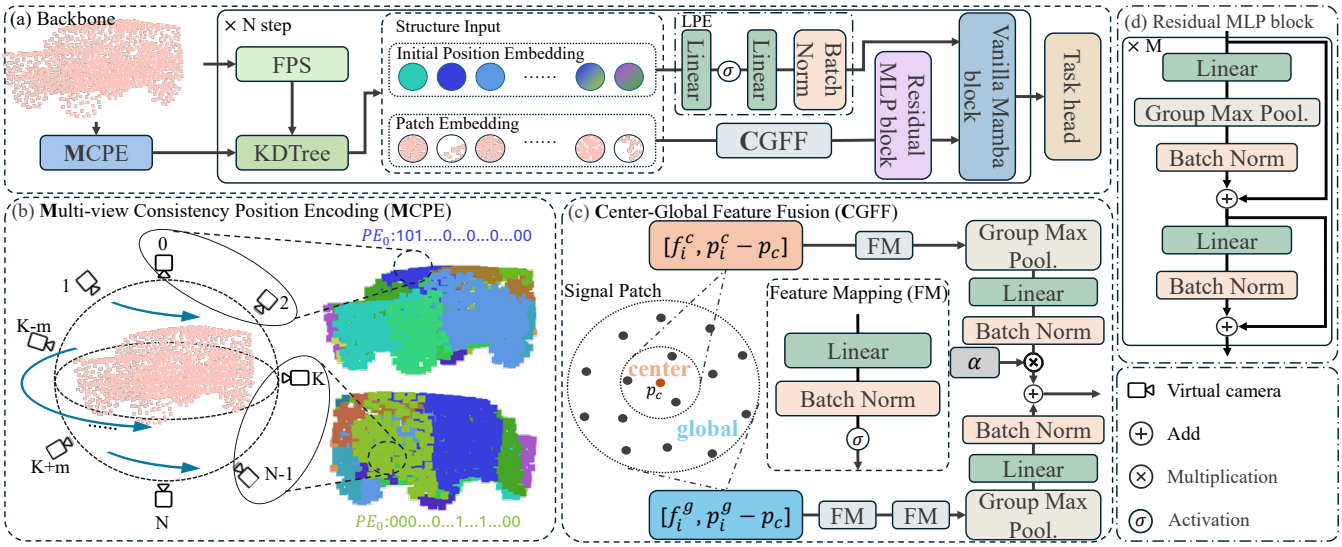

Figure 2: Overview of the proposed PointMC. (a) PointMC employs a hierarchical network architecture for point cloud processing. Downstream tasks are handled by respective task heads. (b) The MCPE is used to obtain the initial position encoding. (c) The CGFF is constructed to fuse features from different scales. (d) The Residual MLP block used in the backbone.

Local Geometry Feature Acquisition. Local geometric feature extraction is fundamental to point cloud analysis (Liu et al. 2020). PointNet++ (Qi et al. 2017) employs shared MLPs with max pooling and multi-resolution grouping for multiscale features. DGCNN (Wang et al. 2019) captures point relationships via edge convolutions. Methods like (Thomas et al. 2019; Xu et al. 2021) simulate point interactions using deformable convolution kernels. Other work (Qian et al. 2022; Lin et al. 2023; Deng et al. 2024; Zeng et al. 2025) leverage streamlined network designs to extract local features efficiently. In this study, we introduce a lightweight yet effective structure to enhance structural feature extraction from local patches.

Methods

Overall Architecture. The overall pipeline of the proposed PointMC is illustrated in Figure 2 (a). We employ a N-step hierarchical network to encode the point cloud and utilize task-specific heads for downstream tasks. Given the input point cloud $P \in \mathbb{R}^{N_\text{p}\times 3}$ and its associated features $F \in \mathbb{R}^{N_\text{p}\times C}$, we first initialize the position encoding $PE_0 \in \mathbb{R}^{N_\text{p}\times N_\text{vc}}$ using MCPE data preprocessing method. The Step $s \in [1, N]$ can be formulated as follows:

$$\begin{aligned} P_s,\ F_\text{idx} &= \text{FPS}(P_{s-1}),\\ F',\ PE_s &= F_{s-1}[F_\text{idx}],\ PE_{s-1}[F_\text{idx}],\\ K_\text{idx} &= \text{KDTree}(P_s, PE_s),\\ \mathcal{F}_s &= \text{Concat}(P_s[K_\text{idx}] - P_s, F'[K_\text{idx}]),\\ F_s &= \text{SSMs}(\text{RMLP}(\text{CGFF}(\mathcal{F}_s)), \text{LPE}(PE_s)), \end{aligned} \tag{1}$$

where the $F_s \in \mathbb{R}^{N_s\times C_s}$, N_s denotes number of points in step s, C_s is the feature dimension, N_vc denotes virtual cameras number. FPS gets the downsample index for P_{s-1}, KDTree get the groups points index, RMLP refers to the Residual MLP block shown in Figure 2 (d).

Multi-view Consistency Learnable Position Encoding (MCLPE)

In nature language processing tasks, individual tokens possess inherent semantic meaning, allowing position encodings to function effectively at the token level. However, in point clouds, individual points lack such semantic significance, making per-point position encoding less effective. Instead, local geometric meaning emerges only through the aggregation of points into patches. In computer vision, methods such as Vision Transformer (Dosovitskiy et al. 2021) divide images into uniformly sized, non-overlapping patches and apply position encodings accordingly. By contrast, overlapping patches are common in point clouds, leading to regions influenced by multiple position encodings. This violates the uniqueness principle of position encoding and weakens the ability of model to capture local dependencies across patches. Additionally, using absolute position encoding based on patch information limits the capacity of model for relative position modeling. Therefore, effective position encoding design for point clouds should prioritize modeling spatial relationships inter-patch, reducing interference from overlapping encodings, and avoiding reliance on absolute point coordinates. Such strategies enhance the granularity of semantic representations across patches and help alleviate semantic ambiguity. MCLPE primarily comprises MCPE to generate initial position encodings and LPE to refine these encodings for better incorporation of prior knowledge.

MCPE Encoding Workflow. The structure of MCPE is depicted in Figure 2 (b), which illustrates the initial encoding generation process. This process initiates from the center position c of the input point cloud, where a bounding sphere is

generated. The radius l of sphere is calculated as:

$$l = \text{Max}(||c - \{p_i\}_{n=0}^{N_s-1}||_2) + \beta, \tag{2}$$

Where the β set as 0.5. The Fibonacci Spherical Sampling (FSS) algorithm is utilised to generate uniformly distributed virtual camera positions toward the center of the point cloud c on the bounding sphere. The number of virtual N_{vc} is controlled by a hyperparameter. The following formulae are employed to calculate the virtual camera position P_{vc}:

$$\begin{aligned} \theta_j &= 2\pi \cdot j/\phi, \\ \varphi_j &= l \cdot arcsin(1 - 2(j + 0.5)/N_{vc}), \\ P^j_{\text{vc}}(x) &= l \cdot cos\theta_j \cdot cos\varphi_j + C_x, \\ P^j_{\text{vc}}(y) &= l \cdot sin\theta_j \cdot cos\varphi_j + C_y, \\ P^j_{\text{vc}}(z) &= l \cdot sin\varphi_j + C_z, \end{aligned} \tag{3}$$

where ϕ is a constant known as the golden ratio, j is the index of the virtual camera. The methodology uses perspective projection to project point clouds onto virtual cameras. In the context of virtual cameras, the generation order of these entities is utilised as the sorting criterion for the generation of a binary initial position code, denoted by PE_0, for every points with length of N_{vc} and initial value of 0 for every point. This approach is employed due to the sequential nature of virtual camera generation, which occurs in a spiral pattern. In the event of point P_i being visible in the jth camera, the jth bit of PE_0 is to be set to 1.

Based on PE_0, the point cloud is adaptively divided into multiple non-overlapping regions, each comprising a distinct number of points. Adjacent patches may correspond to the same position encoding region, and although multiple position encodings can appear within a single patch, their associated regions remain distinct and non-overlapping. This design explicitly prevents interference from multiple position encodings in overlapping areas by ensuring region-wise uniqueness and consistency. The mechanism implicitly encodes relative spatial relationships in a structured manner, offering a principled alternative to traditional encoding strategies that rely on patch information, thereby providing a more robust foundation for modeling local dependencies in point clouds. Furthermore, since PE_0 does not incorporate explicit point cloud information, it potentially reduces the risk of overfitting in neural networks.

Adaptive Encoding via LPE. To facilitate the integration of the implicit prior brought by PE_0 into point cloud features, the position encoding is optimized during training to better accommodate complex spatial structures. To this end, the LPE module is introduced, which employs a lightweight MLP to adaptively model and refine inter-patch spatial relationships. This design aims to support the continuity of structural representation across patches during the position encoding process. The refined PE is then passed into the Vanilla Mamba block as position encoding. We provide the pseudo-code of the proposed MCLPE module in Algorithm 1 for step 1. In the implementation process, parallel computing is utilised to enhance the efficiency of computing.

Algorithm 1: Pseudo-code for MCLPE

Input: Input point cloud P with shape $(N_p, 3)$, the number of virtual cameras: N_{vc}, the idx get from FPS at step s.
Initialization: PE_0 := [[0] * N_p for _ in N_{vc}]. The G_{vcep} used to generate virtual camera external parameters. The F_{pp} used to involves the projection of point cloud onto virtual cameras. The LPE used to subsequent processing of PE_0.
Output: PE with shape (N, C)

```
c := Mean(P)
l := Max(||c − {p_i}_{n=0}^{N_s−1}||_2) + β
Pos := FSS(N_vc)
for j := 0 to N_vc do
    P^j_vc := G_vcep(Pos_j, c, l)
    for i := 0 to N_p do
        v := F_pp(P^j_vc, P_i)
        if v is visible then
            PE_0[i][j]:=1
        end if
    end for
end for
PE := LPE(PE_0[idx])
return PE
```

Center-Global Feature Fusion (CGFF)

In contrast to the hierarchical global feature aggregation of PointNet++, we propose a center-global information aggregation framework operating at the peer level, termed CGFF, as illustrated in Figure 2(c). This method extracts center-global features both globally and locally from a single point patch, integrating them using a gated fusion mechanism. As adjacent point patches overlap, we focus the extraction of local characteristics on the center region of each patch $\mathcal{G} = \{[p_i - p_{\text{c}}, f_i] | i = 0, ..., k-1\}$, p_{c} is the center point of patch, the center $\mathcal{G}_{\text{core}} \in \mathbb{R}^{\lfloor \rho \times k \rfloor \times (3+C)}$ can be formulated:

$$\mathcal{G}_{\text{core}} = \{[p_i - p_{\text{c}}, f_i] \in \mathcal{G} | \text{Top}_{\lfloor \rho \times k \rfloor}(\frac{1}{\|p_i - p_{\text{c}}\|_2})\}, \tag{4}$$

where ρ is an empirically set hyperparameter.

The CGFF module adopts a dual-guidance mechanism that leverages center and global structural cues to enhance the consistency and coordination of local spatial representations, mitigating the loss of intra-patch structural information. A dual-branch architecture is designed to concurrently process information at different spatial scales: the global branch extracts coarse-grained semantics from broader contexts, while the center branch focuses on fine-grained refinement within local neighborhoods.

Global Spatial Perception. For global spatial perception that require modeling long-range dependencies and capturing structural relationships, we employ deep structure to perform hierarchical abstraction of high-level semantics, enabling the extraction of comprehensive context information. The global feature extraction process can be expressed as:

$$F_{\text{global}} = \text{BN}(\text{Linear}(\text{Max}(\text{FM}(\text{FM}(\mathcal{G}))))). \tag{5}$$

Center Spatial Aggregation. A shallow neural network can adaptively integrate neighborhood features and perform cen-

Method	Ref.	Area5	6-fold
with Self-supersived Pre-training			
PCP-MAE	NeurIPS'24	61.3	/
Sonata	CVPR'25	76.0	82.3
Supervised Learning Only			
PDNet-XXL	AAAI'24	72.3	78.3
PCM	AAAI'25	74.1	/
DeepLA-120	CVPR'25	75.7	79.8
CamPoint	CVPR'25	<u>83.3</u>	/
PointMC	/	**83.6**	**87.4**

Table 1: Semantic segmentation results (%) on the S3DIS dataset are evaluated on Area5 and 6-fold cross-validation.

Method	Ref.	Ins. mIoU	Cls. mIoU
with Self-supersived Pre-training			
PointMamba	NeurIPS'24	86.2	84.4
SI-Mamba	CVPR'25	86.1	/
Supervised Learning Only			
PCM	AAAI'25	**86.9**	85.0
SAMBLE	CVPR'25	86.7	84.5
DAFNet	CVPR'25	<u>86.8</u>	<u>85.2</u>
SI-Mamba	CVPR'25	85.9	/
PointMC	/	**86.9**	**85.4**

Table 2: Part segmentation results (%) on ShapeNetPart. The mIoU for all classes (Cls.) and instances (Ins.) are reported.

tral spatial feature aggregation, thereby minimizing computational redundancy and mitigating the inherent risk of overparameterization in deep architectures. The center spatial feature aggregation process can be formulated as:

$$F_{\text{core}} = \text{BN}(\text{Linear}(\text{Max}(\text{FM}(\mathcal{G}_{\text{core}})))). \tag{6}$$

Integrate Center and Global Information. The features obtained from the global perception and center aggregation processes are fused through a lightweight gating mechanism, enabling efficient feature integration. This design facilitates more comprehensive extraction and preservation of structural information within intra-patch regions, while avoiding redundant information and interference that may arise from simple feature concatenation. The integrate center and global information process can be expressed as:

$$F = \alpha \times F_{\text{core}} + F_{\text{global}}, \tag{7}$$

where $F_{\text{core}} \in \mathbb{R}^{N_p \times C}$, the $F_{\text{global}} \in \mathbb{R}^{N_p \times C}$, and FM denotes the Feature Mapping block, as shown in Figure 2(c).

Experiments

Implementation Details. The AdamW (Loshchilov and Hutter 2017) optimizer is used in conjunction with a cosine learning rate scheduler. During training, a warm-up phase is employed to enhance initial training stability. The KDTree (Friedman, Bentley, and Finkel 1977) groups patches based on input point clouds and PE_0. In Vanilla Mamba block, the original ordering of point cloud data is preserved without any spatial sorting or reordering. To ensure fairness, no voting mechanism is used during testing. All experiments are conducted on a GeForce RTX 4090 with 24GB of VRAM.

Experimental Results on Multiple Benchmarks

Semantic Segmentation on S3DIS Dataset. We evaluate the indoor scene understanding of PointMC on the S3DIS (Armeni et al. 2016) dataset with random rotation augmentation. S3DIS is a widely used benchmark for indoor scene parsing, featuring dense point cloud scans with rich geometric and semantic complexity, containing annotated point clouds of 271 rooms from six areas with 13 semantic categories. Performance is reported using mean Intersection over Union (mIoU) on Area 5 and 6-fold cross-validation. As illustrated in Table 1, PointMC consistently outperforms existing methods, achieving 83.6% mIoU on Area5 and delivering notable gains in 6-fold cross-validation. These comprehensive results further validate the effectiveness and generalizability of PointMC, as the spatial consistency enforced by MCLPE and the enhanced local feature aggregation enabled by CGFF together contribute to more robust, fine-grained, and discriminative feature representations for accurate point cloud semantic segmentation. The visualization of the results on S3DIS Area5 as shown in Figure 3.

Part Segmentation on ShapeNetPart Dataset. To assess the fine-grained shape understanding capability of PointMC, experiments on the ShapeNetPart (Yi et al. 2016) dataset are conducted with random scaling augmentation. ShapeNetPart is a benchmark for part-level segmentation of 3D objects, consisting of 16880 shapes across 16 categories with 50 annotated part labels, suitable for evaluating detailed geometric representations. Performance is reported using mIoU for all classes (Cls.) and instances (Ins.). Table 2 presents the performance of PointMC, which achieves optimal results on the mIoU evaluation metric. This demonstrates its superior and robust fine-grained shape understanding capabilities compared to other SOTA methods across various categories, highlighting its effectiveness in accurate and reliable detailed part segmentation tasks on complex 3D structures.

Shape Classification on ScanObjectNN Dataset. We evaluate PointMC on the ScanObjectNN (Uy et al. 2019) benchmark under the PB_T50_RS setting, which contains 2902 real-world 3D object instances across 15 indoor categories. The augmentation of the data is achieved through the implementation of rotation, scale and shuffle, with the point cloud size of 1024. Performance is reported using Overall Accuracy (OA) and mean Accuracy (mAcc). The experimental results, as shown in the Table 3, PointMC achieves significant performance improvements over existing methods while using notably fewer parameters. It consistently outperforms previous state-of-the-art approaches in both OA and mAcc, further confirming the effectiveness of our proposed design. Specifically, MCLPE enforces spatial consistency across patches for clearer and more robust semantic

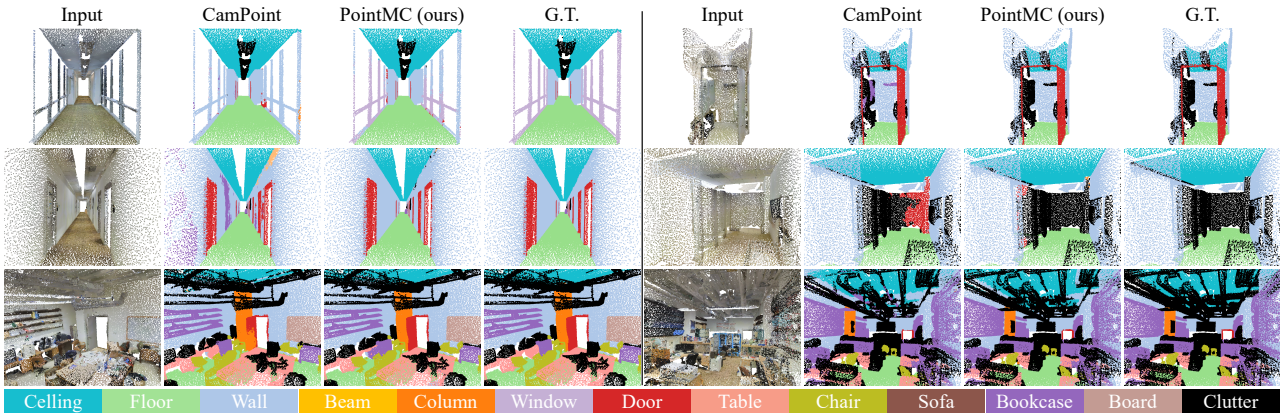

Figure 3: Visualization of large-scale object-level semantic segmentation results on S3DIS Area5 dataset

Method	Ref.	Input	OA	mAcc	Params
with Self-supersived Pre-training					
Point-FEMAE	AAAI'24	2k	90.2	/	27.4M
PCP-MAE	NeurIPS'24	2k	90.4	/	22.1M
Supervised Learning Only					
PCM	AAAI'25	1k	88.1	86.6	34.2M
DeepLA-24	CVPR'25	/	90.6	89.5	/
SI-Mamba	CVPR'25	2k	89.1	/	12.3M
CamPoint	CVPR'25	1k	92.1	91.1	**11.7**M
PointMC	/	1k	**93.4**	**92.7**	**11.7**M

Table 3: Classification results (%) on the ScanObjectNN are evaluated on the most challenging PB_T50_RS variant.

Method	5-way		10-way	
	10-shot	20-shot	10-shot	20-shot
with Self-supersived Pre-training				
Point-BERT	94.6±3.1	96.3±2.7	91.0±5.4	92.7±5.1
MaskPoint	95.0±3.7	97.2±1.7	91.4±4.0	93.4±3.5
Supervised Learning				
DGCNN	31.6±2.8	40.8±4.6	19.9±2.1	16.9±1.5
Transformer	87.8±5.2	93.3±4.3	84.6±5.5	89.4±6.3
OcCo	90.6±2.8	92.5±1.9	82.9±1.3	86.5±2.2
PointMC	**93.0**±3.2	**97.0**±2.4	**88.8**±5.5	**93.2**±3.6

Table 4: Few-shot classification results (%) on ModelNet40. The results are reported as mAcc ± standard deviation.

discrimination, while CGFF facilitates enhanced local geometric feature aggregation to improve intra-patch structural representation. Their complementary integration leads to notable gains in both classification performance and overall feature representation quality.

Few-shot Learning on ModelNet40 Dataset. Following previous work (Yu et al. 2022), we conduct few-shot classification experiments to verify the representation ability of PointMC. The ModelNet40 (Wu et al. 2015) dataset is used with "K-way N-shot" configuration. ModelNet40 is a common benchmark for 3D object classification, containing 12311 CAD generated models across 40 categories. For each configuration, mean accuracy and standard deviation are reported from 10 independent experiments without voting. Table 4 shows that PointMC outperforms other supervised learning methods, highlighting its superior ability to efficiently learn from limited samples without relying on large-scale pre-training. This underscores the strength of PointMC as a purely supervised framework in few-shot scenarios. Nevertheless, when evaluated against other self-supervised pre-training techniques, PointMC revealed a performance disparity. This discrepancy can be attributed to the utilisation of large-scale pre-training and transfer learning mechanisms by the pre-trained model. These mechanisms facilitate the conversion of small sample tasks into rapid awakening and adaptation of extant knowledge. This results in a level of performance that exceeds that of supervised learning methods.

Ablation Study

In this section, we conduct ablation studies on the proposed components in PointMC. The experiments are performed on the ScanObjectNN and S3DIS Area5 Datasets:

Analysis of Each Component. The initial investigation focused on the efficacy of the components proposed in PointMC, as illustrated in Table 5. The first row signifies the baseline model, devoid of any supplementary components. The second row incorporates MCLPE, which significantly improve performance in both classification and segmentation tasks. This demonstrates that MCLPE effectively models spatial relationships between points in point clouds. The third row introduces CGFF, which also enhances performance in comparison to the baseline. This finding suggests that CGFF is capable of extracting and fusing global and core information within patches in an adequate manner. The final row combines MCLPE and CGFF, achieving optimal performance. Collectively, these results confirm that the synergistic integration of MCLPE and CGFF substan-

Baseline	MCLPE	CGFF	ScanObjectNN OA(%)	S3DIS mIoU(%)
✓			91.1	82.4
✓	✓		93.1	82.5
✓		✓	92.4	82.8
✓	✓	✓	**93.4**	**83.6**

Table 5: The efficacy of each component in PointMC.

Method	ScanObjectNN OA(%)	ScanObjectNN mAcc(%)	S3DIS mIoU(%)
V_1	92.3	91.4	82.6
V_2	92.2	91.2	80.6
MCPE	**93.4**	**92.7**	**83.6**

Table 6: The performance of PointMC using MCPE and its variants on scanobjectnn and S3DIS Area5.

tially contributes to improving the robustness and accuracy of PointMC in diverse 3D point cloud understanding tasks.

Analysis of Model Hyperparameters. We conduct ablation studies on the hyperparameters of PointMC, including the α and ρ in the CGFF, and the number of virtual cameras. The results are presented in Figure 4.

It is evident that in order to achieve optimal performance, both ScanObjectNN and S3DIS Area5 necessitate the configuration of distinct values α, ρ and the number of virtual cameras. The findings, when considered holistically, suggest that the selection of the parameter α, ρ and the number of virtual cameras have a considerable effect on the performance of PointMC. Therefore, careful tuning of these hyperparameters is essential for optimal results.

More Analysis of MCLPE

The Variants of MCPE. As outlined in the Method section, two variant encoding methods are devised to establish PE_0. The first variant, termed "cutting cake" (V_1), starts from the c, generates multiple normals, calculates angles betwee vectors from c to any point p_i and each normal, assigns binary values based on which side of each normal the point lies. The second method, termed "building blocks" (V_2), divides the point cloud into equally-sized grids and converts the 3D coordinates within each grid into binary position encoding. The visualization are illustrated as Figure 5.

Table 6 presents the comparison results. It is evident that the proposed MCPE outperforms both variants. This indicates that the proposed method effectively captures the spatial relationships among points in the point cloud, thereby enhancing performance in various downstream tasks.

The MCLPE used in Other Methods. To evaluate the compatibility of MCLPE, we integrated it into existing approaches and conducted experiments on the ScanObjectNN dataset. The results reveal a significant performance improvement, as detailed in Table 7, demonstrating that MCLPE exhibits strong compatibility and that the prior information constructed during the position encoding phase

Method	OA(%)	mAcc(%)
PointMLP	85.4	83.9
PointMLP+MCLPE	86.3(**+0.9**)	84.7(**+0.8**)
CamPoint	92.1	91.1
CamPoint+MCLPE	92.8(**+0.7**)	91.9(**+0.8**)

Table 7: The performance of MCLPE used in other methods.

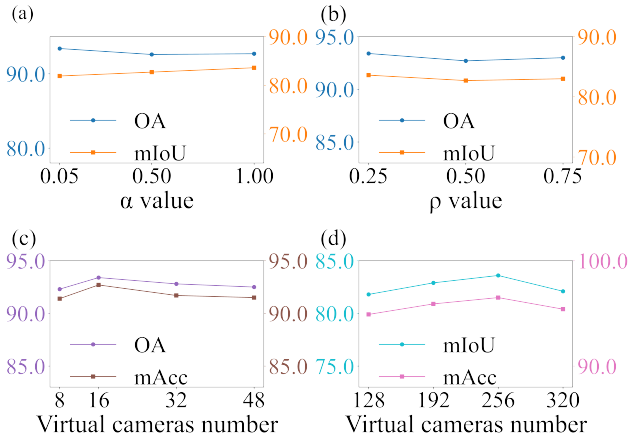

Figure 4: (a)-(b) present ablation studies on α and ρ hyperparameter values, showing the OA on the ScanObjectNN dataset and the mIoU on the S3DIS dataset. (c)-(d) illustrate the impact of varying the number of virtual cameras. Specifically, (c) shows results on the ScanObjectNN dataset, and (d) shows results on the S3DIS dataset.

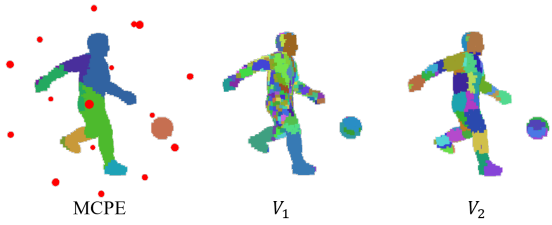

Figure 5: The visualization of MCPE and its variants. The red dots shown in MCPE are the position of virtual cameras.

effectively enhances model performance. These results also indicate the proposed MCLPE method demonstrates strong generalization ability across different approaches, validating its robust plug-and-play potential in diverse scenarios.

Conclusion

This study presents PointMC, addressing position encoding and patch information extraction in point cloud analysis. We propose MCLPE to effectively reduce semantic ambiguity from overlapping patches and develop CGFF to efficiently extract and fuse fine-grained features from local geometric structures. Extensive evaluations show PointMC achieves SOTA results in scene segmentation, object classification, part segmentation, and few-shot classification.

Acknowledgments

This work was supported by the Science and Technology Development Fund of Macau Project 0096/2023/RIA2, and the National Natural Science Foundation of China under grants 62406320.

References

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

Bahri, A.; Yazdanpanah, M.; Noori, M.; Dastani, S.; Cheraghalikhani, M.; Hakim, G. A. V.; Osowiechi, D.; Beizaee, F.; Ben Ayed, I.; and Desrosiers, C. 2025. Spectral informed mamba for robust point cloud processing. In *CVPR*, 11799–11809.

Chen, G.; Wang, M.; Yang, Y.; Yu, K.; Yuan, L.; and Yue, Y. 2023. Pointgpt: Auto-regressively generative pre-training from point clouds. In *NeurIPS*, volume 36, 29667–29679.

Deng, H.; Jing, K.; Cheng, S.; Liu, C.; Ru, J.; Bo, J.; and Wang, L. 2024. Linnet: Linear network for efficient point cloud representation learning. In *NeurIPS*, volume 37, 43189–43209.

Deng, X.; Zhang, W.; Ding, Q.; and Zhang, X. 2023. Pointvector: A vector representation in point cloud analysis. In *CVPR*, 9455–9465.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

Friedman, J. H.; Bentley, J. L.; and Finkel, R. A. 1977. An algorithm for finding best matches in logarithmic expected time. *ACM TOMS*, 3(3): 209–226.

Gu, A.; and Dao, T. 2024. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. In *COLM*.

Hamdi, A.; AlZahrani, F.; Giancola, S.; and Ghanem, B. 2025. MVTN: Learning Multi-view Transformations for 3D Understanding. *International Journal of Computer Vision*, 133(4): 2197–2226.

Hamdi, A.; Giancola, S.; and Ghanem, B. 2023. Voint Cloud: Multi-View Point Cloud Representation for 3D Understanding. In *ICLR*.

Han, X.; Tang, Y.; Wang, Z.; and Li, X. 2024. Mamba3D: Enhancing Local Features for 3D Point Cloud Analysis via State Space Model. In *ACM MM*, 4995–5004. Melbourne VIC Australia.

Li, J.; Wang, J.; and Xu, T. 2024. Pointgl: A simple global-local framework for efficient point cloud analysis. *IEEE Transactions on Multimedia*, 26: 6931–6942.

Li, Z.; Gao, Z.; Tan, C.; Ren, B.; Yang, L. T.; and Li, S. Z. 2024. General Point Model Pretraining with Autoencoding and Autoregressive. In *CVPR*, 20954–20964.

Liang, D.; Zhou, X.; Xu, W.; Zhu, X.; Zou, Z.; Ye, X.; Tan, X.; and Bai, X. 2025. PointMamba: a simple state space model for point cloud analysis. In *NeurIPS*.

Lin, H.; Zheng, X.; Li, L.; Chao, F.; Wang, S.; Wang, Y.; Tian, Y.; and Ji, R. 2023. Meta architecture for point cloud analysis. In *CVPR*, 17682–17691.

Liu, H.; Liu, J.; Jiang, G.; and Jin, X. 2025a. Mssf: A 4d radar and camera fusion framework with multi-stage sampling for 3d object detection in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*.

Liu, Y.; Zhang, C.; Dong, X.; and Ning, J. 2025b. Point Cloud-Based Deep Learning in Industrial Production: A Survey. *ACM Computing Surveys*, 57(7): 1–36.

Liu, Z.; Hu, H.; Cao, Y.; Zhang, Z.; and Tong, X. 2020. A Closer Look at Local Aggregation Operators in Point Cloud Analysis. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *ECCV*, 326–342. Cham.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Ma, X.; Qin, C.; You, H.; Ran, H.; and Fu, Y. 2022. Rethinking Network Design and Local Geometry in Point Cloud: A Simple Residual MLP Framework. In *ICLR*.

Park, C.; Jeong, Y.; Cho, M.; and Park, J. 2022. Fast point transformer. In *CVPR*, 16949–16958.

Park, J.; Lee, S.; Kim, S.; Xiong, Y.; and Kim, H. J. 2023. Self-positioning point-based transformer for point cloud understanding. In *CVPR*, 21814–21823.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017. PointNet++: deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 5105–5114.

Qian, G.; Li, Y.; Peng, H.; Mai, J.; Hammoud, H.; Elhoseiny, M.; and Ghanem, B. 2022. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In *NeurIPS*, volume 35, 23192–23204.

Qu, W.; Wang, J.; Gong, Y.; Huang, X.; and Xiao, L. 2025. An end-to-end robust point cloud semantic segmentation network with single-step conditional diffusion models. In *CVPR*, 27325–27335.

Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; and Liu, Y. 2024. RoFormer: Enhanced transformer with Rotary Position Embedding. *Neurocomputing*, 568: 127063.

Su, K.; Wu, Q.; Cai, P.; Zhu, X.; Lu, X.; Wang, Z.; and Hu, K. 2025. RI-MAE: Rotation-Invariant Masked AutoEncoders for Self-Supervised Point Cloud Representation Learning. In *AAAI*, volume 39, 7015–7023. Issue: 7.

Sun, H.; Wang, Y.; Wang, P.; Deng, H.; Cai, X.; and Li, D. 2024. VSFormer: Mining Correlations in Flexible View Set for Multi-view 3D Shape Understanding. *IEEE Transactions on Visualization and Computer Graphics*.

Thomas, H.; Qi, C. R.; Deschaud, J.-E.; Marcotegui, B.; Goulette, F.; and Guibas, L. J. 2019. Kpconv: Flexible and deformable convolution for point clouds. In *ICCV*, 6411–6420.

Uy, M. A.; Pham, Q.-H.; Hua, B.-S.; Nguyen, D. T.; and Yeung, S.-K. 2019. Revisiting Point Cloud Classification: A New Benchmark Dataset and Classification Model on Real-World Data. In *ICCV*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, .; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*, volume 30.

Wang, C.; He, S.; Fang, X.; Han, J.; Liu, Z.; Ning, X.; Li, W.; and Tiwari, P. 2025. Point Clouds Meets Physics: Dynamic Acoustic Field Fitting Network for Point Cloud Understanding. In *CVPR*, 22182–22192.

Wang, W. 2024. Real-Time Fast 3D Reconstruction of Heritage Buildings Based on 3D Gaussian Splashing. In *ICSECE*, 1014–1018.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic Graph CNN for Learning on Point Clouds. *ACM Transactions on Graphics*, 38(5): 1–12.

Wang, Z.; Chen, Z.; Wu, Y.; Zhao, Z.; Zhou, L.; and Xu, D. 2024. PoinTramba: A Hybrid Transformer-Mamba Framework for Point Cloud Analysis. ArXiv:2405.15463 [cs].

Wei, X.; Yu, R.; and Sun, J. 2022. Learning view-based graph convolutional network for multi-view 3D shape analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6): 7525–7541.

Wu, C.; Wan, Y.; Fu, H.; Pfrommer, J.; Zhong, Z.; Zheng, J.; Zhang, J.; and Beyerer, J. 2025a. SAMBLE: Shape-Specific Point Cloud Sampling for an Optimal Trade-Off Between Local Detail and Global Uniformity. In *CVPR*, 1342–1352.

Wu, X.; DeTone, D.; Frost, D.; Shen, T.; Xie, C.; Yang, N.; Engel, J.; Newcombe, R.; Zhao, H.; and Straub, J. 2025b. Sonata: Self-supervised learning of reliable point representations. In *CVPR*, 22193–22204.

Wu, X.; Jiang, L.; Wang, P.-S.; Liu, Z.; Liu, X.; Qiao, Y.; Ouyang, W.; He, T.; and Zhao, H. 2024. Point transformer v3: Simpler faster stronger. In *CVPR*, 4840–4851.

Wu, X.; Lao, Y.; Jiang, L.; Liu, X.; and Zhao, H. 2022. Point transformer v2: Grouped vector attention and partition-based pooling. In *NeurIPS*, volume 35, 33330–33342.

Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; and Xiao, J. 2015. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, 1912–1920.

Xu, L.; Cui, Q.; Hong, R.; Xu, W.; Chen, E.; Yuan, X.; Li, C.; and Tang, Y. 2024. Group multi-view transformer for 3d shape analysis with spatial encoding. *IEEE Transactions on Multimedia*.

Xu, M.; Ding, R.; Zhao, H.; and Qi, X. 2021. Paconv: Position adaptive convolution with dynamic kernel assembling on point clouds. In *CVPR*, 3173–3182.

Yan, T.; Yin, J.; Lang, X.; Yang, R.; Xu, C.-Z.; and Shen, J. 2025. OLiDM: Object-aware LiDAR Diffusion Models for Autonomous Driving. In *AAAI*, volume 39, 9121–9129. Issue: 9.

Yi, L.; Kim, V. G.; Ceylan, D.; Shen, I.-C.; Yan, M.; Su, H.; Lu, C.; Huang, Q.; Sheffer, A.; and Guibas, L. 2016. A Scalable Active Framework for Region Annotation in 3D Shape Collections. *SIGGRAPH Asia*.

Yin, X.; Yang, X.; Liu, L.; Wang, N.; and Gao, X. 2024. Point deformable network with enhanced normal embedding for point cloud analysis. In *AAAI*, volume 38, 6738–6746. Issue: 7.

Yu, H.-T.; and Song, M. 2024. Mm-point: Multi-view information-enhanced multi-modal self-supervised 3d point cloud understanding. In *AAAI*, volume 38, 6773–6781. Issue: 7.

Yu, X.; Li, J.; Wong, C.-C.; Vong, C.-M.; and Liang, Y. 2025a. FACNet: Feature alignment fast point cloud completion network. *Computational Visual Media*, 11(1): 141–157.

Yu, X.; Tang, L.; Rao, Y.; Huang, T.; Zhou, J.; and Lu, J. 2022. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *CVPR*, 19313–19322.

Yu, Y.; Verbree, E.; van Oosterom, P.; and Pottgiesser, U. 2025b. 3D Gaussian Splatting for Modern Architectural Heritage: Integrating UAV-Based Data Acquisition and Advanced Photorealistic 3D Techniques. *AGILE: GIScience Series*, 6: 51.

Zeng, Z.; Dong, M.; Zhou, J.; Qiu, H.; Dong, Z.; Luo, M.; and Li, B. 2025. DeepLA-Net: Very Deep Local Aggregation Networks for Point Cloud Analysis. In *CVPR*, 1330–1341.

Zha, Y.; Ji, H.; Li, J.; Li, R.; Dai, T.; Chen, B.; Wang, Z.; and Xia, S.-T. 2024. Towards compact 3d representations via point feature enhancement masked autoencoders. In *AAAI*, volume 38, 6962–6970.

Zhang, Q.; Peng, J.; Huang, Z.; Feng, W.; and Lin, D. 2025a. Generative Hard Example Augmentation for Semantic Point Cloud Segmentation. In *CVPR*, 22205–22214.

Zhang, R.; Guo, Z.; Gao, P.; Fang, R.; Zhao, B.; Wang, D.; Qiao, Y.; and Li, H. 2022. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. In *NeurIPS*, volume 35, 27061–27074.

Zhang, T.; Yuan, H.; Qi, L.; Zhang, J.; Zhou, Q.; Ji, S.; Yan, S.; and Li, X. 2025b. Point Cloud Mamba: Point Cloud Learning via State Space Model. In *AAAI*, 10121–10130.

Zhang, X.; Zhang, S.; and Yan, J. 2024. Pcp-mae: Learning to predict centers for point masked autoencoders. In *NeurIPS*, volume 37, 80303–80327.

Zhao, H.; Jiang, L.; Jia, J.; Torr, P. H.; and Koltun, V. 2021. Point transformer. In *ICCV*, 16259–16268.

Zou, Y.; Yu, H.; Yang, Z.; Li, Z.; and Akhtar, N. 2024. Improved mlp point cloud processing with high-dimensional positional encoding. In *AAAI*, volume 38, 7891–7899. Issue: 7.