CylinMamba3D: Geometry-Aware Cylindrical State Space Modeling for 3D Point Clouds

Anonymous ECCV submission

Paper ID 12135

Abstract. Modern state-space models, such as Mamba, achieve linear-time complexity and efficient parallelization for long-sequence modeling, making them attractive for large-scale 3D point-cloud understanding. However, directly applying Mamba to unordered point clouds yields suboptimal performance, as state-space models inherently assume structured sequential inputs and lack geometry-aware scan ordering. To address this limitation, we propose CylinMamba3D, a cylindrical state-space framework tailored for 3D point-cloud understanding. The core idea is to transform unordered local tokens into a structured cylindrical sequence by mapping token centers into cylindrical coordinates and defining a spatially coherent scan strategy. The ordered tokens are processed by a Cylindrical State Space Model that integrates stable-state dynamics, bidirectional propagation, and local mixing to preserve geometric continuity while enabling efficient global modeling. Extensive experiments demonstrate that CylinMamba3D achieves a strong trade-off between accuracy and efficiency across benchmark datasets. The proposed method achieves 94.0% overall accuracy on ModelNet40, 91.1% on ScanObjectNN, and 98.1% on ShapeNet. These results establish an effective geometry-aware state-space model for point-cloud learning with linear computational complexity.

Keywords: Point cloud, State Space Model, Cylindrical Scanning, Order-Robust Global Modeling

1 Introduction

Point cloud understanding serves as a cornerstone of modern computer vision, powering applications from autonomous driving and robotic manipulation to 3D vision-language-action paradigms [1–3]. In these contexts, perception and reasoning must occur in real-time, often processing large-scale LiDAR data that demands both high accuracy and computational efficiency [4]. However, the unordered and sparse nature of point clouds makes representation learning significantly more challenging than grid-based image processing. Recent breakthroughs in sequence modeling have introduced State Space Models (SSMs), such as Mamba, which offer linear-time complexity and strong parallelization [5]. While SSMs provide a scalable alternative to the quadratic complexity of Transformers, their direct application to 3D tasks remains suboptimal. This limitation stems

from a fundamental misalignment: SSMs assume structured sequential inputs, whereas point clouds lack a natural scan order. Conventional heuristic ordering strategies often fracture geometric locality, leading to unstable state propagation and degraded performance [6]. To bridge this gap, we propose CylinMamba3D, a geometry-aware sequential modeling framework that imposes a structured cylindrical ordering over local point tokens. Unlike arbitrary space-filling curves or XYZ-sorting, cylindrical scanning naturally aligns with the radial symmetry and height-dependent distribution common in 3D objects and sensor-centric scenes. By mapping token centers into cylindrical coordinates and scanning along vertical, angular, and radial axes, we preserve neighborhood dependencies more effectively; points that are spatially adjacent in 3D space remain proximal in the resulting sequence. This transformation converts the unordered point cloud into a geometrically coherent sequence without the structural disruption typical of Cartesian-based flattening.

We process these ordered tokens using a Cylindrical State Space Model (C-SSM), which integrates stable state-space dynamics with bidirectional propagation and local convolutional mixing. This architecture preserves geometric continuity while enabling efficient linear-time modeling of long-range dependencies. Extensive experiments on standard benchmarks demonstrate that CylinMamba3D consistently outperforms both Transformer-based architectures and recent Mamba adaptations in terms of accuracy and robustness.

- We introduce **CylinMamba3D**, a cylindrical state-space framework that harmonizes geometry-aware token ordering with linear-time modeling for efficient 3D point-cloud learning.
- We propose Cylindrical State Space Modeling (C-SSM) that integrates structured scan ordering, bidirectional state propagation, and local mixing to preserve spatial continuity and stabilize long-range modeling.
- We introduce an encoder built from stacked Cylindrical Mamba blocks, combining local geometric aggregation with sequential state modeling to efficiently encode cylindrical scan features.

We demonstrate state-of-the-art performance across multiple benchmarks, notably achieving 94.0% OA on ModelNet40 and 91.1% OA on ScanObjectNN, while exhibiting superior efficiency and strong few-shot generalization compared to contemporary SSM and Transformer baselines.

2 Related Work

Point cloud understanding has traditionally navigated the tension between permutation-invariant local aggregation and the need for scalable global context modeling. Early point-based backbones prioritized local neighborhood grouping, but modern architectures increasingly operate on token sequences derived from point patches or sparse voxels to enable long-range interactions. A persistent challenge in this transition is that point clouds lack a canonical order; consequently,

sequence-oriented mixers benefit from structurally-principled serialization that induces stable token adjacency. Recent deterministic ordering schemes have sought to bridge this gap, such as Point Cloud Mamba, which utilizes Consistent Traverse Serialization (CTS) and axis-permuted variants to mitigate order ambiguity in global modeling [7]. Similarly, PointMamba adopts locality-preserving space-filling curves, such as Hilbert and Trans-Hilbert scans [8], to enable linear-time global mixing while enhancing robustness to input permutations [9]. Beyond SSMs, PTv3 demonstrates that such serialization strategies can effectively replace costly K-Nearest Neighbor (KNN) graph construction, supporting efficient attention with enlarged receptive fields and strong scalability on large-scale benchmarks [10].

This emphasis on serialization as an efficiency lever extends into voxel-based 3D perception, where the discretization of space naturally lends itself to structured long-range modeling. Sparse voxelization methods often struggle with preserving locality across window partitions, a problem addressed by Voxel Mamba through the serialization of sparse voxels into a single Hilbert-ordered sequence combined with dual-scale SSM mixing [11]. Other approaches, such as ScatterFormer, bypass explicit window sorting by flattening voxels into a global sequence and applying Scattered Linear Attention with Cross-Window Interaction to facilitate information exchange without expensive shifting operations [12]. These voxel-based designs underscore a critical theme: the method of token formation and the chosen scan order significantly dictate the effectiveness of global interaction modules, particularly under the sparse spatial layouts typical of LiDAR-based detection.

The search for more efficient alternatives to quadratic attention has recently positioned State Space Models (SSMs) as a primary candidate for spatial domain modeling via selective scanning. While originally formulated for 1D sequences, the expansion of SSMs into vision led by Mamba [5] has necessitated the design of scan patterns that respect multi-dimensional structures. In 2D vision, VMamba introduced 2D Selective Scan (SS2D) to bridge 1D scanning with 2D spatial aggregation [13], while VSSD advanced this through a non-causal State Space Duality formulation and multi-directional scanning to strengthen context propagation [14]. These innovations have paved the way for 3D-specific adaptations like E-Mamba, which combines density-aware Scalable Embedding with Holosync Reordering Pooling to improve feature extraction from irregular point samples [15]. Collectively, these advancements suggest that the success of SSMs in the 3D domain depends heavily on domain-appropriate tokenization and geometry-aware ordering a realization that directly motivates our proposed cylindrical scanning strategy for point cloud sequences.

3 Proposed Method

Fig. 1 illustrates the architecture of **CylinMamba3D**. Our framework addresses the fundamental mismatch between the unordered nature of 3D point clouds and the sequential requirements of State Space Models (SSMs). By projecting local

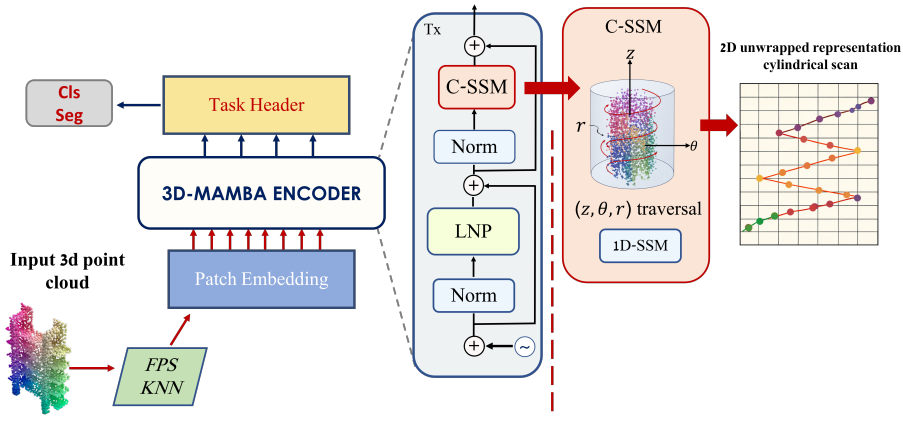


Fig. 1. Overview of the proposed Cylindrical State Space Model (C-SSM) within the 3D-Mamba encoder. After FPS and KNN-based patch embedding, tokens are processed by stacked Mamba blocks consisting of Norm, LNP, and Cylindrical State Space Model (C-SSM). The Cylindrical State Space Model (C-SSM) converts spatial centers to cylindrical coordinates (r, θ, z) and performs geometry-aligned helical traversal. The rightmost panel illustrates the *unwrapped cylindrical representation*, where θ is mapped to a linear axis while preserving z. This transformation converts the 3D helical path into a structured 2D grid, enabling sequential 1D state-space propagation along geometry-consistent paths.

tokens into a structured cylindrical coordinate system, we enable a *structurally-principled* serialization that preserves geometric continuity during global context modeling.

3.1 Point Cloud Tokenization

Given an input point cloud $\mathcal{P} = \{p_i \in \mathbb{R}^3\}_{i=1}^{N}$, we first perform **Local Point Grouping** to extract a manageable set of geometric primitives. We utilize Farthest Point Sampling (FPS) to select G representative centers $P = \{c_j\}_{j=1}^{G}$ and construct local neighborhoods $\mathcal{N}(c_j)$ via K-Nearest Neighbors (KNN). Each neighborhood is encoded into a high-dimensional token $x_j \in \mathbb{R}^C$ using a shared multilayer perceptron (MLP). The resulting token set $X = \{x_j\}_{j=1}^{G}$ maintains associated Cartesian coordinates $c_j = (x_j, y_j, z_j)$, which serve as the geometric anchors for the subsequent serialization process.

3.2 The 3D-Mamba Encoder

The core of our framework is the **3D-Mamba Encoder**, a hierarchical backbone consisting of stacked blocks designed to harmonize local geometric consistency with long-range dependency modeling. Unlike standard Transformers that suffer

from quadratic complexity, our encoder utilizes the linear-time complexity of SSMs. Each encoder block follows a residual structure:

$$X' = X + \text{LNP}(\text{Norm}(X)), \quad X'' = X' + \text{C-SSM}(\text{Norm}(X')), \tag{1}$$

where LNP denotes Local Neighborhood Processing and C-SSM represents the Cylindrical State Space Model. This dual-path design ensures that the model simultaneously attends to fine-grained local structures and global spatial organization.

Cylindrical Coordinate Transformation To enable sequence modeling, we must impose an order on X. Traditional heuristic scans (e.g., Morton or Z-order) often fracture 3D proximity when flattened. We propose a **Cylindrical Transformation** $\phi(c_j)$ to project Cartesian centers into (r_j, θ_j, z_j):

$$r_j = \sqrt{x_j^2 + y_j^2}, \quad \theta_j = \arctan 2(y_j, x_j), \quad z_j = z_j. \tag{2}$$

This mapping is specifically chosen because it aligns with the radial symmetry and vertical stratification common in 3D objects and LiDAR-centric scans. By decoupling height (z), angular position (θ), and radial depth (r), we establish a coordinate space where lexicographical sorting naturally preserves neighborhood dependencies.

3.3 Cylindrical State Space Model (C-SSM)

The proposed Cylindrical State Space Model (C-SSM) serves as the global interaction engine of our encoder. The objective is to facilitate stable state propagation along geometry-consistent trajectories.

Cylindrical Selective Scan: We define a scan mode m as a specific sorting priority of the cylindrical coordinates. The permutation π_m is induced by:

$$\pi_m = \text{argsort}(z, \theta, r). \tag{3}$$

By placing tokens with similar vertical and angular coordinates in adjacent sequence positions, we minimize the geometric distortion typically seen in 1D serialization. The reordered sequence X_m is then processed by the core SSM equations:

$$h_t = \mathbf{A}h_{t-1} + \mathbf{B}x_t, \quad y_t = \mathbf{C}h_t, \tag{4}$$

where $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are learnable parameters. The hidden state h_t thus acts as a global memory that accumulates information along a helical path through the 3D structure.

Multi-mode Aggregation: To ensure the model is robust to varied geometries and reduces scanning bias, we utilize multiple traversal modes $\mathcal{M}$ (e.g., reversing the z-axis or θ-direction). After processing each mode, we apply the

Algorithm 1: 3D Cylindrical Selective Scan

Input : Local tokens $X \in \mathbb{R}^{G\times D}$, spatial centers $P \in \mathbb{R}^{G\times 3}$, scan modes $\mathcal{M}$
Output: Geometry-aware aggregated features $Y \in \mathbb{R}^{G\times D}$
Initialize $Y \leftarrow 0$;
$\Phi \leftarrow \text{Transform}(P) \rightarrow (r, \theta, z)$; `// Cylindrical Projection`
for *each scan mode* $m \in \mathcal{M}$ **do**
  $\pi_m \leftarrow \text{argsort}(\Phi, \text{mode} = m)$; `// Sort by` $z \rightarrow \theta \rightarrow r$
  $X_m \leftarrow \text{Permute}(X, \pi_m)$; `// Serialize tokens`
  `// State Space Modeling`
  $H_m \leftarrow \text{SSM_Forward}(X_m; A, B)$; // $h_t = Ah_{t-1} + Bx_t$
  $Y_m \leftarrow \text{SSM_Project}(H_m; C)$; // $y_t = Ch_t$
  $\tilde{Y}_m \leftarrow \text{InversePermute}(Y_m, \pi_m)$; `// Restore spatial order`
  $Y \leftarrow Y + \tilde{Y}_m$; `// Directional aggregation`
return LayerNorm(Y);

inverse permutation π_m^{-1} to restore the original spatial indexing, followed by a summation of the parallel paths:

$$Y = \sum_{m\in\mathcal{M}} \pi_m^{-1}(\text{SSM}(\pi_m(X))). \tag{5}$$

This procedure, detailed in Algorithm 1, allows the model to perceive the point cloud from multiple orientation-aware perspectives while maintaining linear efficiency.

3.4 Local Neighborhood Processing and Global Head

Parallel to the global C-SSM, the **Local Neighborhood Processing (LNP)** module refines features by aggregating information within the KNN-graph. We apply a weighted pooling mechanism to enhance discriminative local geometric features:

$$x_j^{\text{LNP}} = \frac{\sum_k \hat{x}_{j,k} e^{\hat{x}_{j,k}}}{\sum_k e^{\hat{x}_{j,k}}}, \tag{6}$$

where $\hat{x}_{j,k}$ represents normalized neighbor features. Finally, for tasks like object classification, we employ a classification head that aggregates the global sequence through combined max and mean pooling, concatenated with a class token x_{CLS}. This feature vector f is passed through a multi-layer perceptron to produce the final prediction $\hat{y} = \text{MLP}(f)$.

4 Evaluation

This section details our experimental setup and benchmarks the proposed **Cylin-Mamba3D** framework against contemporary state-of-the-art architectures across multiple downstream tasks, including object classification, part segmentation,

and few-shot learning. Furthermore, we conduct comprehensive ablation studies to quantify the contributions of specific architectural components. Unless otherwise specified, all performance metrics are reported for models trained from scratch without the use of voting strategies.

4.1 Implementation Details

Our architecture employs $L = 12$ encoder layers with a constant feature dimension of $C = 384$. Each input point cloud is sampled to $N = 1,024$ points and partitioned into $G = 128$ patches, with each patch containing 32 points. The local neighborhood size within the Local Neighborhood Processing (LNP) block is fixed at $k = 4$. We utilize a drop path rate of 0.02 and a classification dimension of 40. For the state-space modeling, we adopt the bidirectional Mamba `v4` variant, where the cylindrical scan ordering is integrated directly into the encoder to ensure seamless spatial-to-sequential mapping without external reordering overhead.

The model is optimized using the AdamW optimizer paired with a cosine learning rate decay schedule over 300 epochs. We set the initial learning rate to 5×10^{-4} with a weight decay of 0.05. To ensure training stability, gradient norm clipping is applied with a maximum threshold of 1.0. Optimization is conducted with a batch size of 32, and label smoothing is disabled. The learning rate ratios for both the classification head and the local feature aggregation module are maintained at 1.0. All experiments were executed on a single NVIDIA RTX 4070 Ti SUPER GPU with 16GB of memory.

4.2 Results Object Classification

We evaluate CylinMamba3D on two standard object classification benchmarks: ModelNet40 [16] and ScanObjectNN [17]. **ModelNet40** consists of approximately 12,000 synthetic 3D CAD models spanning 40 categories. Following established protocols, point clouds are subsampled to $N = 1,024$ points, and data augmentation is performed via random scaling and translation. **ScanObjectNN** contains roughly 15,000 real-world scanned objects and is evaluated across three established variants: *OBJ_BG*, *OBJ_ONLY*, and the rigorous *PB_T50_RS* split [17]. For ScanObjectNN, we utilize $N = 2,048$ points and apply random rotations as the primary data augmentation.

Results. Table 1 provides a detailed comparison of object classification performance. When trained from scratch, CylinMamba3D achieves a state-of-the-art Overall Accuracy (OA) of **91.1%** on the challenging *PB_T50_RS* split, which further scales with the application of a voting strategy. On the *OBJ_BG* and *OBJ_ONLY* variants, CylinMamba3D reaches **94.5%** and **93.5%** OA, respectively surpassing the baseline Mamba3D by **+1.56%** and **+1.42%**. On ModelNet40, the framework achieves **94.0%** OA without voting and **94.6%** with voting, demonstrating the efficacy of our geometry-aware cylindrical scanning.

Method	Backbone	Ordering	ScanObjectNN			ModelNet40
			OBJ_BG↑	OBJ_ONLY↑	PB_T50_RS↑	1k P↑
Supervised Learning Only: Architectures						
PointNet [18]	MLP	Unordered	73.3	79.2	68.0	89.2
PointNet++ [19]	Hierarchical PointNet(SA)	FPS + local Neighborhood	82.3	84.3	77.9	90.7
DGCNN [20]	EdgeConv	Dynamic KNN graph	82.8	86.4	78.1	92.9
Transformer-based Architectures						
Point-BERT [21]	Transformer	Group tokens (FPS + kNN)	87.1	90.8	83.1	92.7
PointMAE [22]	Transformer (MAE)	Group tokens + masking	88.2	91.4	85.2	93.2
Mamba-based Architectures						
PointMamba [9]	Mamba (SSM)	Hilbert / Trans-Hilbert scan	88.30	87.78	82.48	93.6
PCM [7]	PointMLP + Mamba	CTS (6 xyz permutations)	88.0	90.7	84.9	93.4
PoinTramba [23]	Transformer + Mamba	BIO (bidirectional ordering)	87.6	90.3	84.0	92.7
Mamba3D [24]	Mamba (SSM) + LNP	Patch tokens + bi-SSM	92.94	92.08	91.81	93.4
⋆ **CylinMamba3D (w/o vot.)**	Mamba (C-SSM)	Cylindrical Scanning	**94.5**	**93.5**	**91.1**	**94.0**
⋆ **CylinMamba3D (w/ vot.)**	Mamba (C-SSM)	Cylindrical Scanning	**93.9**	**93.0**		**94.6**

Table 1. Classification results on ScanObjectNN and ModelNet40. The proposed CylinMamba3D replaces conventional token traversal strategies with cylindrical scanning within the state-space models framework.

Method	5-way		10-way	
	10-shot↑	20-shot↑	10-shot↑	20-shot↑
Supervised Learning Only				
DGCNN [20]	31.6 ± 2.8	40.8 ± 4.6	19.9 ± 2.1	16.9 ± 1.5
Transformer [25]	87.8 ± 5.2	93.3 ± 4.3	84.6 ± 5.5	89.4 ± 6.3
Mamba3D [24]	92.6 ± 3.7	96.9 ± 2.4	88.1 ± 5.3	93.1 ± 3.6
⋆ **CylinMamba3D**	**93.6 ± 4.46**	**97.7 ± 2.2**	**89.3 ± 6.5**	**94.0 ± 3.2**

Table 2. Few-shot classification on ModelNet40. Overall accuracy (%) without voting is reported.

4.3 Few-shot Classification

We assess the generalization capability of CylinMamba3D under limited supervision on ModelNet40 using the standard n-way, m-shot protocol. In this setup, n classes are randomly sampled, with m labeled instances per class provided for training ($n \times m$ total samples), while 20 unseen instances per class are utilized for testing. We evaluate $n \in \{5, 10\}$ and $m \in \{10, 20\}$ and report the mean OA (%) ± standard deviation over 10 independent runs without voting (Table 2).

CylinMamba3D demonstrates robust few-shot performance, achieving **93.6% ± 4.46** and **97.7% ± 2.2** in the 5-way 10-shot and 20-shot settings, respectively. Under the more complex 10-way configurations, the model maintains high accuracy with **89.3% ± 6.5** (10-shot) and **94.0% ± 3.2** (20-shot). These results highlight the ability of our cylindrical state-space formulation to extract discriminative geometric features even when labeled data is extremely sparse.

Method	mIoU$_C$ (%)↑	mIoU$_I$ (%)↑	#P (M)↓	#F (G)↓
Supervised Learning Only				
PointNet [18]	80.4	83.7	3.6	4.9
PointNet++ [19]	81.9	85.1	1.0	4.9
DGCNN [20]	82.3	85.2	1.3	12.4
Transformer [25]	83.4	85.1	27.1	15.5
Mamba3D [24]	83.7	85.7	23.0	11.8
⋆ **CylinMamba3D**	**85.3**	**87.4**	**20.8**	**8.2**

Table 3. Part segmentation on ShapeNetPart dataset. We report class mIoU (mIoU$_C$) and instance mIoU (mIoU$_I$), along with model parameters #P and FLOPs #F.

Scan Strategy	OA (%) ↑	#F (G) ↓
Sequential (Baseline)	91.40	4.1
Cylindrical	93.50	**3.2**
Spherical	**94.66**	3.8

Table 4. Ablation study of scan strategies on ScanObjectNN (OBJ_ONLY). Spherical traversal achieves the highest accuracy, while cylindrical scanning offers the peak computational efficiency.

5 Ablation Studies

Comprehensive ablation experiments are performed to examine the architectural components and to assess the influence of scan ordering strategies. All models are trained from scratch and evaluated on the ScanObjectNN (OBJ_ONLY) benchmark unless otherwise specified.

5.1 Impact of Scan Design

This section investigates the influence of geometric scan strategies on representation learning within our Cylindrical State Space Model (C-SSM) framework. We evaluate three distinct traversal mechanisms: **Sequential Scanning** (baseline), **Cylindrical Scanning**, and **Spherical Scanning**. To ensure a controlled comparison, all configurations utilize an identical architecture, training schedule, and parameter count, isolating the token traversal strategy as the sole variable in the state-space mixer as shown in table 4, .

Sequential Scanning (Baseline). Sequential scanning processes tokens in their default grouped order without imposing explicit geometric constraints. While this approach minimizes implementation complexity, the resulting sequence fails to reflect the intrinsic 3D spatial proximity of the points. Consequently, spatially

distant tokens may become adjacent in the 1D sequence, introducing significant geometric distortion during state propagation. This misalignment disrupts locality preservation and degrades modeling efficiency.

Cylindrical Scanning. Cylindrical scanning restructures the token sequence by sorting centers lexicographically based on (z, θ, r) coordinates. This strategy aligns the traversal path with vertical and angular continuity, reflecting the structural priors of common 3D geometries. By aggregating complementary traversal paths through multiple scan modes, this geometry-aware permutation reduces the discrepancy between spatial and sequential distances. This leads to more stable state transitions and enhanced feature consistency.

Spherical Scanning. Spherical scanning organizes tokens according to polar coordinates (r, θ, ϕ) using bin-based lexicographic ordering. By discretizing angular dimensions and ranking radial distances, this implementation constructs a deterministic, geometry-consistent traversal that captures global angular structure and radial distribution. Compared to the cylindrical approach, spherical scanning provides a more isotropic alignment between sequential propagation and the 3D spatial layout. Spherical scanning achieves a superior accuracy of 94.66% with 3.8 GFLOPs, establishing the optimal trade-off between representation power and computational expenditure.

5.2 Sensitivity Analysis of Cylindrical Binning

The sensitivity of the Cylindrical State Space Model (C-SSM) to the discretization granularity used for constructing the cylindrical scan order is analyzed. Given token centers, cylindrical coordinates (r, θ, z) are quantized into b discrete bins to define a lexicographic traversal. The bin count b controls the spatial resolution of the scan: smaller values yield coarser partitions, whereas larger values induce a finer discretization of the underlying geometry.

To evaluate robustness, the bin density is varied as $b \in \{8, 16, 32\}$ while keeping the architecture, training schedule, and random seed fixed. Table **??** reports the overall accuracy (OA) on ScanObjectNN *OBJ_ONLY*. The best performance is obtained at $b = 16$, while both smaller and larger bin counts reduce accuracy. This behavior indicates that excessively coarse binning can merge distinct local structures, whereas overly fine binning can fragment neighborhoods and weaken the continuity of state propagation along the scan. Overall, the results suggest that moderate discretization provides the most effective trade-off between geometric fidelity and traversal stability, and $b = 16$ is used as the default setting in the remaining experiments.

6 Conclusions

In this paper, we presented CylinMamba3D, a geometry-aware state-space framework designed for efficient and effective 3D point-cloud understanding. By transforming unordered point tokens into structurally-principled cylindrical sequences,

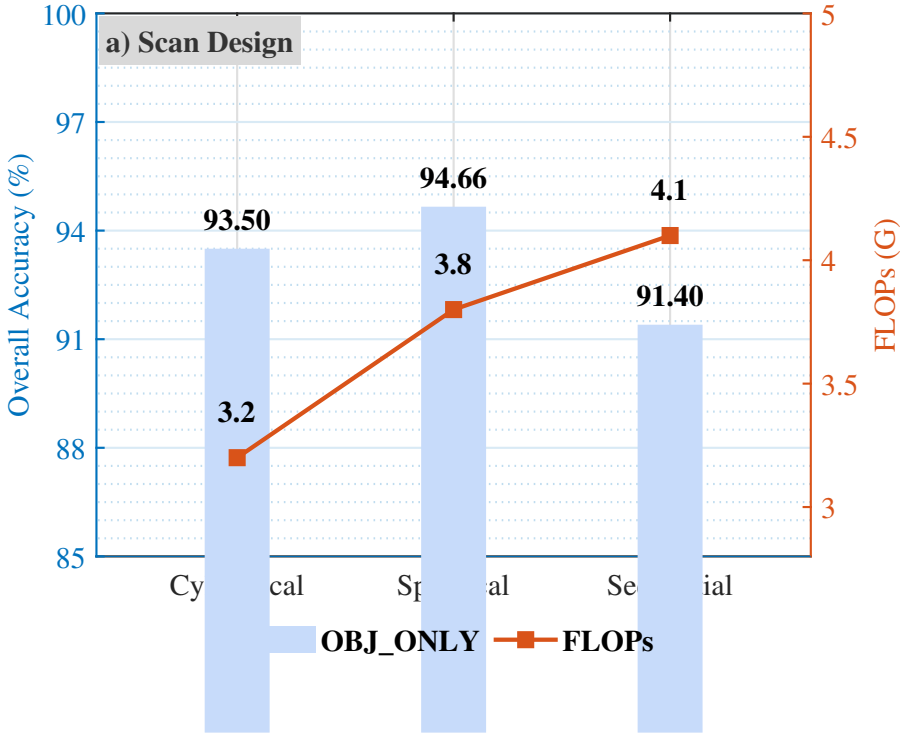


Fig. 2. Ablation studies on scan strategy and cylindrical binning.

we bridge the gap between unstructured 3D geometry and sequential state-space modeling. Our proposed Cylindrical State Space Module (C-SSM) leverages the radial and vertical symmetries inherent in 3D data, ensuring that the serialization preserves spatial continuity. Through the integration of bidirectional propagation and local convolutional mixing, the framework maintains stable long-range dependencies while operating with linear computational complexity. Extensive evaluations on benchmark datasets demonstrate the efficacy of our approach, achieving 94.0% OA on ModelNet40, 91.1% OA on ScanObjectNN. These results indicate that CylinMamba3D provides a superior trade-off between accuracy and efficiency compared to both Transformer-based and contemporary Mamba-based architectures. Furthermore, our ablation studies confirm that the cylindrical scan strategy and stable-state dynamics are critical for capturing complex 3D structures. Future work will explore extending the CylinMamba3D backbone to multi-modal 3D vision-language tasks.

References

1. Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M.: Deep learning for 3d point clouds: A survey. IEEE transactions on pattern analysis and machine intelligence **43**(12) (2020) 4338–4364
2. Yang, Z., Chen, L., Sun, Y., Li, H.: Visual point cloud forecasting enables scalable autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2024) 14673–14684
3. Chen, S., Garcia, R., Schmid, C., Laptev, I.: Polarnet: 3d point clouds for language-guided robotic manipulation. arXiv preprint arXiv:2309.15596 (2023)
4. Krispel, G., Schinagl, D., Fruhwirth-Reisinger, C., Possegger, H., Bischof, H.: Maeli: Masked autoencoder for large-scale lidar point clouds. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. (2024) 3383–3392
5. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. In: First conference on language modeling. (2024)
6. He, Y., Tu, B., Liu, B., Li, J., Plaza, A.: 3dss-mamba: 3d-spectral-spatial mamba for hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing **62** (2024) 1–16
7. Zhang, T., Yuan, H., Qi, L., Zhang, J., Zhou, Q., Ji, S., Yan, S., Li, X.: Point cloud mamba: Point cloud learning via state space model. In: Proceedings of the AAAI Conference on Artificial Intelligence. Volume 39. (2025) 10121–10130
8. Nguyen, K., Hassan, G.M., Mian, A.: Occlusion-aware text-image-point cloud pretraining for open-world 3d object recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2025) 16965–16975
9. Liang, D., Zhou, X., Xu, W., Zhu, X., Zou, Z., Ye, X., Tan, X., Bai, X.: Pointmamba: A simple state space model for point cloud analysis. Advances in neural information processing systems **37** (2024) 32653–32677
10. Wu, X., Jiang, L., Wang, P.S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H.: Point transformer v3: Simpler faster stronger. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. (2024) 4840–4851
11. Zhang, G., Fan, L., He, C., Lei, Z., Zhang, Z., Zhang, L.: Voxel mamba: Group-free state space models for point cloud based 3d object detection. Advances in Neural Information Processing Systems **37** (2024) 81489–81509
12. He, C., Li, R., Zhang, G., Zhang, L.: Scatterformer: Efficient voxel transformer with scattered linear attention. In: European Conference on Computer Vision, Springer (2024) 74–92
13. Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., Ye, Q., Jiao, J., Liu, Y.: Vmamba: Visual state space model. Advances in neural information processing systems **37** (2024) 103031–103063
14. Shi, Y., Li, M., Dong, M., Xu, C.: Vssd: Vision mamba with non-causal state space duality. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. (2025) 10819–10829
15. Li, D., Gao, Z., Hao, S., Xun, Z., Song, J., Cheng, J., Zhao, J.: E-mamba: An efficient mamba point cloud analysis method with enhanced feature representation. Neurocomputing (2025) 130201
16. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 1912–1920

17. Uy, M.A., Pham, Q.H., Hua, B.S., Nguyen, D.T., Yeung, S.K.: Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In: International Conference on Computer Vision (ICCV). (2019)
18. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 652–660
19. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
20. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (tog) **38**(5) (2019) 1–12
21. Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., Lu, J.: Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. (2022) 19313–19322
22. Pang, Y., Tay, E.H.F., Yuan, L., Chen, Z.: Masked autoencoders for 3d point cloud self-supervised learning. World Scientific Annual Review of Artificial Intelligence **1** (2023) 2440001
23. Wang, Z., Chen, Z., Wu, Y., Zhao, Z., Zhou, L., Xu, D.: Pointramba: A hybrid transformer-mamba framework for point cloud analysis. arXiv preprint arXiv:2405.15463 (2024)
24. Han, X., Tang, Y., Wang, Z., Li, X.: Mamba3d: Enhancing local features for 3d point cloud analysis via state space model. In: Proceedings of the 32nd ACM International Conference on Multimedia. (2024) 4995–5004
25. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)