PointChain: Learning Generalizable Point Cloud Representations via Structural Chain Modeling

Luyao Wang[1,2*], **Chuxin Wang**[1,2*], **Qiao Li**[1], **Tianzhu Zhang**[1,2†]

[1]University of Science and Technology of China
[2]National Key Laboratoray of Deep Space Exploration, Deep Space Exploration Laboratory

Abstract

Recent advances in point cloud analysis have increasingly leveraged large-scale unlabeled data through self-supervised representation learning. Autoregressive models based on next-token prediction have shown strong performance, but they usually model point clouds as linear sequences, ignoring their inherent spatial structure. To address this limitation, we propose PointChain, a novel autoregressive paradigm inspired by human perception mechanisms, designed to better align with the structural properties of point cloud. Specifically, we introduce structural chain encoding, which models the understanding process as a global-to-local structural chain inference, preserving spatial relationships throughout the prediction sequence. During pre-training, we design two auxiliary tasks: a next-scale prediction task that encourages cross-scale reasoning, and a scale-level contrastive learning task that promotes semantic consistency across scales. These components guide the model to learn more discriminative and generalizable point cloud representations. Experiments on multiple benchmarks, using both Transformer and Mamba backbones, validate the effectiveness of our approach. PointChain achieves state-of-the-art performance on several downstream tasks, including 93.75% accuracy on the hardest split of ScanObjectNN without voting strategy.

1 Introduction

Point cloud analysis, as a fundamental technology in computer vision and 3D perception, impacts the development of various cutting-edge fields, such as autonomous driving (Li et al. 2020), robotics (Pomerleau et al. 2015), and virtual reality (Guo et al. 2020). However, most existing methods rely heavily on manually annotated data for supervised learning (Qi et al. 2017a,b; Li et al. 2018; Thomas et al. 2019; Wang et al. 2023; Qian et al. 2022; Ma et al. 2022; Wang et al. 2025a; Zhang et al. 2025), limiting their generalizability and hindering deployment in complex real-world scenarios.

Inspired by advancements in NLP and CV, researchers have increasingly turned to self-supervised learning. This paradigm enables models to acquire general-purpose features and representations from point clouds without relying on extensive labeled datasets. Recent self-supervised learning methods are typically categorized into contrastive methods and generative methods. Contrastive methods (Xie et al. 2020; Zhang et al. 2021; Yang et al. 2018; Navaneet et al. 2020) (Fig. 1(a)) aim to learn discriminative representations. However, they depend on carefully designed data augmentations to generate diverse views, making their performance highly sensitive to the choice of augmentation strategies. Additionally, contrastive objectives focus on pulling similar samples together and pushing dissimilar ones apart, which may overlook fine-grained structural information. Generative methods include two main paradigms: masked point modeling (MPM) and autoregressive modeling (AR). MPM-based methods (Yu et al. 2022; Pang et al. 2022; Wang et al. 2024; Liang et al. 2024; Wang et al. 2025b; Zha et al. 2025) (Fig. 1(b)) aim to reconstruct masked regions of a point cloud, thus encouraging the model to capture both local geometric details and global structural information. However,

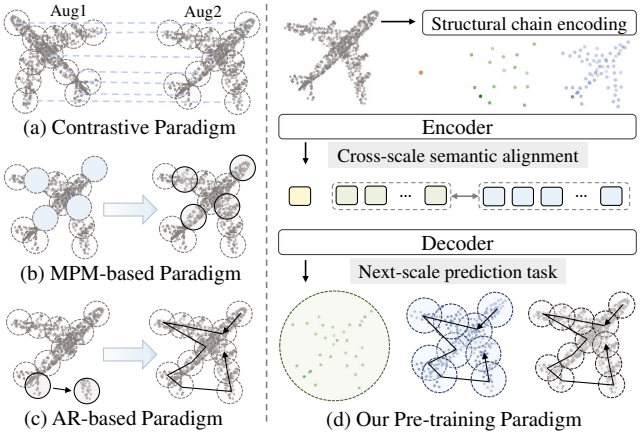


Figure 1: Pre-training paradigms. (a) Contrastive paradigm can learn discriminative features, but it requires complex matching pairs construction. (b) MPM-based paradigm rebuilds masked region, risking shape leakage. (c) AR-based paradigm predicts next-token by linearizing point clouds, disrupting spatial topology. (d) Our paradigm models point clouds as a structural chain, enhancing structural understanding via cross-scale inference and semantic alignment.

*These authors contributed equally.
†Corresponding author.

since the positional encodings of masked regions are exposed to the decoder, these methods suffer from global shape leakage, which lowers reconstruction difficulty and weakens generalization. To address this, AR-based methods (Chen et al. 2023) (Fig. 1(c)) mitigate shape leakage by autoregressively predicting the next-token in a sequence. However, modeling 3D point clouds as 1D sequences inherently disrupts their spatial topology, as it requires imposing a predefined order on inherently unordered data. This artificial order can obscure meaningful geometric structures and limit the model to capture complex spatial dependencies.

To overcome this limitation, we revisit the problem of how to construct effective token sequences for point cloud autoregressive modeling. To this end, we identify two key challenges that must be addressed: **(1) How to observe point clouds under the autoregressive paradigm?** As a direct representation of real-world 3D geometry, point clouds inherently encode rich structural information. When humans perceive 3D objects, they typically begin with a coarse understanding of global shapes and progressively focus on finer details. Inspired by this perceptual mechanism, we argue that the autoregressive process should use a strategy of processing the point cloud from coarse to fine. **(2) How to learn generalizable geometric and semantic knowledge?** For point clouds, this requires the model not only to understand local geometric patterns, but also to learn abstract semantic concepts that are consistent across scales. However, existing pretext tasks (Chen et al. 2023; Pang et al. 2022) often focus on low-level geometry, limiting the model to capture higher-level semantics and structural regularities. We believe that enabling the model to perform cross-scale reasoning and semantic alignment is essential for learning more generalizable and expressive point cloud representations.

Based on the above analysis, we propose PointChain, a novel autoregressive paradigm that reconstructs point clouds into structural chains, progressing from a coarse global view to a fine local understanding, as shown in Fig. 1(d). The framework comprises three key components. **(1) Structural Chain Encoding.** We first construct multi-scale token representations from the input point cloud. Within each scale, tokens are ordered using the Morton curve (Morton 1966) to preserve spatial locality. Tokens from different scales are then organized into a unified 1D sequence following a coarse-to-fine order. To help the model distinguish hierarchical structures, we introduce a scale identifier that embeds scale information into token representations. This encoding scheme transforms the point cloud understanding process into structural chain inference, akin to a cognitive progression from global to local perception. **(2) Next-Scale Prediction.** Inspired by VAR (Tian et al. 2024), we design a pretext task in which the model autoregressively predicts the structure at the next-scale and position. This task encourages the model to learn progressive geometric abstraction, inferring local details conditioned on global context. Consequently, the model develops a hierarchical understanding of 3D structures, significantly improving its multi-scale reasoning capabilities. **(3) Cross-Scale Semantic Alignment.** To guide the model in learning semantic correspondences across scales, we introduce a scale-level contrastive learning task that aligns token features with the same semantics in different scales. By leveraging positive and negative pairs, this approach enhances the model to capture cross-scale semantic consistency and improves its generalization.

The main contributions of our work are: (1) We propose PointChain, a novel autoregressive paradigm that models point clouds as a structural chain progressing from coarse to fine. (2) We introduce two pre-training tasks: next-scale prediction for progressive cross-scale inference of geometric features, and cross-scale semantic alignment, a contrastive learning task enforcing semantic consistency. (3) Extensive experiments on Transformer and Mamba architectures demonstrate the superior performance of our method.

2 Related Work

2.1 Self-Supervised Learning for NLP and CV

Self-supervised learning can guide models to learn general features and knowledge from large amounts of unlabeled data and has been extensively studied in the fields of NLP and CV. Depending on pretext tasks, the methods can be classified into contrastive methods and generative methods. Contrastive methods (Gao, Yao, and Chen 2021; Chen et al. 2020b; Oord, Li, and Vinyals 2018; Qian et al. 2021; Yu et al. 2017) encourage model to learn discriminative features by constructing positive and negative pairs, with the optimization objective of pulling together the feature distances of positive pairs and pushing apart those of negative pairs. Generative methods include masked reconstruction and autoregressive modeling (AR). In terms of masked reconstruction (Bao et al. 2022; Devlin et al. 2019; He et al. 2022; Chang et al. 2022, 2023), BERT (Devlin et al. 2019) predicts the words that are masked in the input text, guiding the model to learn contextual semantics; MAE (He et al. 2022) utilizes visible regions to reconstruct masked regions in an image, helping the model learn both global and local details. AR-based methods (Radford et al. 2018; Chen et al. 2020a; Esser, Rombach, and Ommer 2021; Tian et al. 2024) achieve sequence modeling through the prediction of the next-token, demonstrating outstanding performance. Recently, VAR (Tian et al. 2024) innovatively apply autoregressive generation to the image domain, achieving progressive image generation through scale-by-scale predictions with remarkable results. Similarly to VAR, our PointChain draws inspiration from the hierarchical nature of visual data. However, unlike 2D image grids with regular spatial structure, 3D point clouds are inherently unordered and irregular, presenting unique challenges for autoregressive modeling. To address this, PointChain reformulates point clouds as structural chains by leveraging Morton order and multi-scale organization. Moreover, unlike VAR which focuses on generative modeling of 2D images, PointChain is designed for self-supervised learning in 3D, with pre-training tasks tailored to the geometric and semantic properties of point clouds.

2.2 Self-Supervised Learning for Point Cloud

Inspired by advancements in NLP and CV, increasing attention is being given to self-supervised frameworks for point cloud representation learning. Contrastive methods (Zhang

et al. 2021; Xie et al. 2020; Yang et al. 2018; Navaneet et al. 2020) learn discriminative features by constructing positive and negative pairs from different views of the same point cloud. Approaches like DepthContrast (Zhang et al. 2021) and PointContrast (Xie et al. 2020) perform instance-level and point-level discrimination tasks, respectively. MPM-besed methods (Yu et al. 2022; Pang et al. 2022; Zhang et al. 2022; Liu, Cai, and Lee 2022; Zhou et al. 2024; Wang et al. 2024, 2025b; Zha et al. 2025) such as PointBERT (Yu et al. 2022) and PointMAE (Pang et al. 2022), achieve feature learning by reconstructing the discrete tokens or original coordinates of the masked regions. To mitigate potential leakage of overall shape during mask reconstruction, existing methods adopt the strategy of introducing positional information of masked regions only in shallow decoders. However, this solution is suboptimal and fails to fundamentally resolve the conflict between information leakage and feature learning. In contrast, AR-based methods, represented by PointGPT (Chen et al. 2023), effectively circumvent the issue by modeling the 3D point cloud as a linear sequence and performing next-token prediction tasks, demonstrating excellent performance. However, this destroys the inherent spatial topology of point clouds, restricting the model to perceive complex spatial relationships. In addition, with the rise of state space models (Kalman 1960; Gu et al. 2021; Gu, Goel, and Ré 2021), Mamba (Gu and Dao 2023; Dao and Gu 2024) has emerged as an alternative framework to Transformer. Current Mamba-based methods (Liang et al. 2024; Han et al. 2024) mostly adopt the 1D serialization pre-training paradigm, inheriting the MPM strategy of Point-MAE (Pang et al. 2022), which still faces issues with spatial topology destruction as well as leakage of the overall shape. To more accurately capture spatial information in point clouds, we propose structural chain encoding and next-scale prediction tasks, guiding the model to perform coarse-to-fine modeling, which has not been systematically studied before. Our framework is architecture-agnostic and achieves performance improvements on both Transformer-based and Mamba-based architectures.

3 Method

3.1 Overview

Current AR-based methods typically model 3D point clouds as 1D sequences, destroying the inherent spatial topology of point clouds. We aim to design a new autoregressive paradigm that constructs point clouds as structural chains, performing global-to-local structural modeling. Fig. 2 shows an overview of PointChain. Multi-scale point clouds with different granularities are first constructed from the input point clouds P, and then embedded into C-dimensional feature space by token embedding. The tokens are ordered within the same scale, and different scales are organized into 1D structural chains in the order from coarse to fine. Meanwhile, we add scale identifiers to the structural chain, enabling the model to differentiate and model multi-scale structures. Then, an encoder composed of standard Mamba blocks or Transformer blocks is employed to model point cloud features from coarse to fine. To enhance high-level semantic understanding and achieve cross-scale semantic alignment, we perform scale-level contrastive learning on the output features M_e. In addition, to capture structural patterns across different point cloud resolutions, we introduce a shallow decoder that performs next-scale prediction, where each token is trained to predict the geometric structure at the location of the next token in the next finer scale.

3.2 Structural Chain Encoding

Point clouds inherently exhibit a multi-level structure. A modeling approach that starts with global geometric features and progressively focuses on local details aligns better with their characteristics. To achieve this, we propose structural chain encoding, which models point clouds as structural chains, guiding structural modeling from coarse to fine.

Generation of Multi-Scale Point Clouds. Given an input point cloud $P \in \mathbb{R}^{N\times 3}$, we first construct three scales of point cloud representations. Considering that farthest point sampling (FPS) has the property of preferentially covering the contours of the point cloud, we utilize FPS to sample the key points from P and take the first S_1 and S_2 points sequentially to form the coarse-grained point cloud P_{S_1} and the medium-grained point cloud P_{S_2}, respectively. Meanwhile, we use the original point cloud P as the finest-grained scale P_{S_3}. Due to the large number of point clouds at each scale, we further select a small number of representative points from P_{S_1}, P_{S_2}, and P_{S_3} following the aforementioned paradigm, to form the center point sets P_{G_0}, P_{G_1}, and P_{G_2}, which are used for the next-scale prediction and cross-scale semantic alignment tasks.

Token Embedding. After obtaining the multi-scale center points P_{G_0}, P_{G_1}, P_{G_2}, we first serialize the center points P_{G_n} at each scale using the Morton encoding (Morton 1966) to preserve geometric continuity and then mini-PointNet++ (Qi et al. 2017b) is used for local feature extraction. For each point in P_{G_n}, to enhance multi-scale locality, we select neighborhood points from $P_{S_{n+1}}$ within radius range r_k through ball query, resulting in the patch $P_{G_n}^{r_k}$:

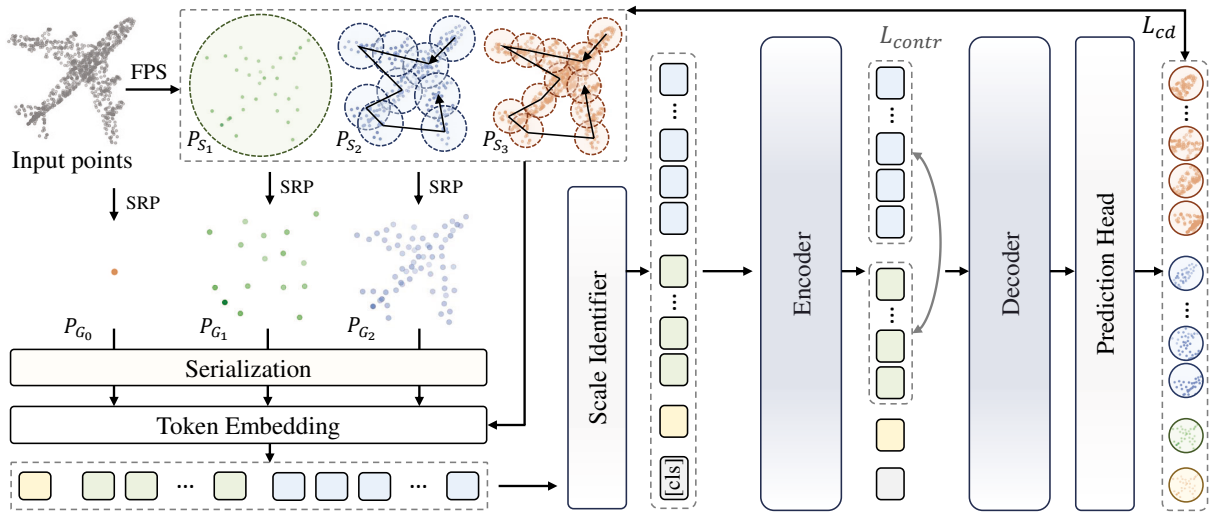
$$P_{G_n}^{r_k} = \text{BallQuery}\left(\left(P_{G_n}, P_{S_{n+1}}\right), r_k\right), \tag{1}$$

where $k \in \{1, 2, 3\}$. Subsequently, the multi-scale patches $P_{G_n}^{r_k}$ are processed by a lightweight token embedding module based on PointNet++, enabling the construction of hierarchical point cloud representations E_n across multiple scales. This process can be expressed as follows:

$$E_n = \text{Linear}\left(\phi_1\left(P_{G_n}^{r_1}\right) | \phi_2\left(P_{G_n}^{r_2}\right) | \phi_3\left(P_{G_n}^{r_3}\right)\right). \tag{2}$$

Due to varying information granularity across scales, the center point sets are fed into separate embedding layers with no shared weights for better feature learning.

Point Cloud Structural Chain. We serialize multi-scale point cloud tokens in order from coarse to fine to form a structural chain. This structural chain guides global modeling with coarse-grained tokens, and then gradually incorporates more fine-grained local information, thus providing top-down structural prior knowledge and enhancing cross-scale inference. Since tokens at different scales may correspond to the same spatial location but encode features at



(a) Structural Chain Encoding (b) Pre-training Tasks

Figure 2: Illustration of PointChain. (a) Structural Chain Encoding. We first generate multi-scale point clouds from the input and select representative points (SRP). After mapping to high-dimensional space via token embedding, we add scale identifiers and organize the data into a 1D structural chain. (b) Pre-training Tasks. We add a [cls] token to the feature sequence and input it into the encoder-decoder architecture for scale-level contrastive learning and the next-scale prediction tasks.

different granularities, we design a scale identifier to map tokens at each scale E_n to different high-dimensional feature spaces E'_n, resulting in the final structure chain encoding M_0. This module helps the model distinguish the scale information, thus enhancing its ability to model multi-scale structures. The process can be represented as:

$$M_0 = \text{concat}\left(E'_0, E'_1, E'_2\right), \quad E'_n = \alpha_n * E_n + \beta_n. \quad (3)$$

3.3 Next-Scale Prediction

In contrast to conventional next-token prediction, we introduce a next-scale prediction task. This task encourages the model to understand the hierarchical structure within point clouds. In the following, we detail the autoregressive modeling and the next-scale prediction, respectively.

Autoregressive Modeling. We implement feature modeling using an encoder-decoder architecture. Following the previous method (Chen et al. 2023), we add absolute position encoding to the encoder to represent the basic geometric shape, and add relative position encoding (direction of the next patch to be predicted) to the decoder to guide point cloud reconstruction. After adding a [cls] token, the structural chain encoding M_0 and the absolute position encoding are input into the encoder, and the output feature M_e is used for cross-scale semantic alignment. The decoder takes M_e and relative position encoding as input and outputs feature M_d used to predict the next-scale structure information.

Encoder and decoder can be built from standard Mamba blocks or Transformer blocks, thus forming two architectures. Transformer-based architecture consists of ordinary Transformer decoder blocks, and we introduce causal temporal masking in both the encoder and decoder so that each token in the structure chain can only see the unmasked tokens before it. For the encoder, we additionally mask some of the tokens to improve feature learning (Chen et al. 2023) and maintain consistency across different scale masking regions. Mamba-based architecture consists of ordinary Mamba blocks, which aggregate structural chain information through unidirectional scanning, naturally satisfying the causal temporal modeling without the need for masking.

Next-Scale Prediction. In this task, the model autoregressively predicts the fine-grained spatial information at the next-scale with the help of the prior knowledge of coarse-grained tokens, i.e., it predicts the distribution of point of all patches at the next-scale with the current scale tokens. The model gradually learns the multi-level semantic correspondences of the point cloud during cross-scale inference, improving its understanding of point cloud structures.

From the multi-scale point clouds P_{S_1}, P_{S_2}, P_{S_3}, M neighboring points are selected by KNN for the center point sets $P_{G_0}, P_{G_1}, P_{G_2}$, constituting point patches with different information granularity as the ground truth for the next-scale prediction task. In order to avoid the training instability that may be caused by scale differences and to accelerate model convergence, we adopt a Dynamic Scaling Strategy (DSS) based on the finest granularity to spatially normalize multi-scale patches. Specifically, we perform centralizing on each scale patch, then calculate the radius ratio between the current scale and the finest-grained scale patches as the scale factor λ_n, and finally achieve the spatial normalization by inverse scaling. For each scale patch set $P_n \in \mathbb{R}^{G_n \times M \times 3}$, we obtain its normalized ground truth:

$$P_n^{gt} = \frac{P_n - c_n}{\lambda_n}, \quad \lambda_n = \frac{r_n}{r_b}, \quad (4)$$

where $c_n \in \mathbb{R}^{G_n \times 1 \times 3}$ represents the center coordinates of each patch in P_n, r_n represents the average radius of these

patches, and r_b serves as the radius benchmark, which is the radius of the finest-grained point cloud patches. Given the predicted patches P_{pre} and the normalized ground truth P_{gt}, we calculate the generation loss L^g using the l_1-form and l_2-form of the Chamfer Distance (Fan, Su, and Guibas 2017). Specifically, $L^g = L_1^g + L_2^g$. Here, l_n-form Chamfer Distance L_n^g, $n \in \{1, 2\}$, is defined as:

$$L_n^g = \frac{1}{|P_{pre}|} \sum_{a \in P_{pre}} \min_{b \in P_{gt}} \|a - b\|_n^n + \frac{1}{|P_{gt}|} \sum_{b \in P_{gt}} \min_{a \in P_{pre}} \|a - b\|_n^n. \quad (5)$$

Notably, the next-token prediction can be viewed as a specific instance of our proposed next-scale prediction in the single-scale setting. Compared to this baseline, our coarse-to-fine reconstruction strategy allows the model to start from simple contour reconstruction and progressively refine the structural details of the point cloud, leading to more comprehensive and robust feature representations.

3.4 Cross-Scale Semantic Alignment

Semantic structures in point clouds exhibit inherent multi-scale hierarchies, where semantic cues across different scales are interrelated and progressively refined. To enhance the model to capture semantic consistency across scales, we propose a scale-level contrastive task. This task introduces an explicit inductive bias for multi-scale semantic alignment through structured supervision.

Specifically, we exploit the property that patches at different scales may share the same center point but encode different levels of detail to define a point-by-point comparison strategy: cross-scale patches with the same center point are treated as positive pairs while the remaining combinations are treated as negative pairs. This formulation encourages the model to enhance consistency between representations that share a common semantics but differ in granularity, improving the model to distinguish between semantically unrelated structures. We adopt the PointInfoNCE loss (Xie et al. 2020) as the optimization objective, maximizing the mutual information of positive pairs while minimizing the similarity of negative pairs. The loss L^c is defined as follows:

$$L^c = -\frac{1}{|M_p|} \sum_{(i,j) \in M_p} \log \frac{\exp\left(E_i^{P_1} \times E_j^{P_2} / \tau\right)}{\sum_{(\cdot,k) \in M_p} \exp\left(E_i^{P_1} \times E_k^{P_2} / \tau\right)}, \quad (6)$$

where M_p represents the index set of one-to-one matching pairs, E^{P_1} and E^{P_2} are the features of different scale point clouds, and τ is a hyperparameter that controls the sharpness of the distribution. For point i in point cloud P_1, $(i, j) \in M_p$ is a positive pair, and its features $(E_i^{P_1}, E_j^{P_2})$ are encouraged to be similar. Meanwhile, $\{(i, k) | (\cdot, k) \in M_p, k \neq j\}$ is the negative sample set, whose features are pushed apart. In pre-training phase, the training objective is L^p:

$$L^p = L^g + L^c, \quad (7)$$

where L^g is the next-scale prediction loss and L^c is the contrastive learning loss. In fine-tuning phase, we integrate L^p as a regular term to prevent the model from forgetting the knowledge gained during pre-training, which can accelerate training convergence and enhance generalization ability. The loss functions L^f are defined as:

$$L^f = L^d + \lambda \times L^p, \quad (8)$$

where L^d is the downstream task loss and λ is a hyperparameter that balances the contributions of L^p and L^d.

4 Experiments

4.1 Implementation Details

Our structural chain consists of three scale point clouds P_{S_1}, P_{S_2}, P_{S_3}, from which representative points are selected to form the center point sets P_{G_0}, P_{G_1}, P_{G_2} for pre-training. For each center point, KNN is used to select $M = 32$ neighborhood points to form patches, which are then spatially normalized and used as the ground truth for the next-scale prediction task. The encoder contains 12 Mamba blocks or Transformer blocks and the decoder contains 4 identical blocks with feature dimension $C = 384$. Following previous work (Pang et al. 2022; Yu et al. 2022), we pre-train our model on the ShapeNet dataset (Chang et al. 2015), which contains 55 common object categories with 52,472 independent 3D models. Each input point cloud contains $N = 1024$ points, and we construct multi-scale point clouds with the number of points $[S_1, S_2, S_3] = [32, 256, 1024]$. The first two scales are downsampled and combined with the input point cloud centroid to obtain the center point sets with the number of points $[G_0, G_1, G_2] = [1, 16, 64]$, which are used to perform the next-scale prediction and cross-scale semantic alignment. The hyperparameter τ in the contrastive learning task is set to 0.1 to control the distribution sharpness. All experiments are conducted with an NVIDIA RTX 3090 GPU. More details can be found in supplementary material.

4.2 Comparison on Downstream Tasks

We conduct experiments on multiple downstream tasks using Transformer (Vaswani et al. 2017) and Mamba (Gu and Dao 2023) as basic architectures. By default, the reported results do not employ any voting strategy.

Real-World Classification on ScanObjectNN. ScanObjectNN (Uy et al. 2019) is a challenging real-world 3D dataset with 15 categories and around 15,000 objects, captured from cluttered indoor scenes. As shown in Tab. 1, we conduct experiments on three increasingly difficult variants of the dataset: OBJ-BG, OBJ-ONLY, and PB-T50-RS. Our models based on both Transformer and Mamba achieve state-of-the-art performance within their respective architecture families. Notably, the Mamba-based model achieves both the best FLOPs efficiency (enabling global modeling with fewer tokens and without bidirectional SSM) and the highest classification accuracy, reaching 93.75% on the challenging PB-T50-RS split. These results highlight that structural chain modeling combined with cross-scale pre-training effectively captures the intrinsic features of point clouds, enhancing both representation and generalization.

Synthetic Classification on ModelNet40. ModelNet40 is a widely used synthetic dataset (Wu et al. 2015) that contains

Method	Backbone	Param. (M)	FLOPs (G)	ScanObjectNN			ModelNet40
				OBJ-BG	OBJ-ONLY	PB-T50-RS	1k P
Supervised learning							
PointNet	-	3.5	0.5	73.3	79.2	68.0	89.2
PointNet++	-	1.5	1.7	82.3	84.3	77.9	90.7
PointCNN	-	0.6	-	86.1	85.5	78.5	92.2
DGCNN	-	1.8	2.4	82.8	86.2	78.1	92.9
PointMLP	-	12.6	31.4	-	-	85.4	94.5
PCM	-	34.2	45.0	-	-	88.1	93.4
Self-supervised learning							
PointBERT	Transformer	22.1	4.8	87.43	88.12	83.07	92.7
MaskPoint	Transformer	22.1	4.8	89.30	88.10	84.30	-
PointM2AE	Transformer	12.7	7.9	91.22	88.81	86.43	92.9
PointMAE†	Transformer	22.1	4.8	92.77	91.22	89.04	92.7
PointGPT-S†	Transformer	22.1	4.9	93.39	92.43	89.17	-
Ours†	Transformer	24.3	3.8	**94.15**	**94.15**	**91.50**	**94.7**
PointMamba†	Mamba	12.3	3.1	94.32	92.60	89.31	93.6
Mamba3D†	Mamba	16.9	3.9	93.12	92.08	92.05	94.7
Ours†	Mamba	18.2	2.9	**95.70**	**94.15**	**93.75**	**95.2**

Table 1: Shape classification on ScanObjectNN and ModelNet40 datasets. We report the classification accuracy (%) without voting strategy. † indicates that using simple rotational augmentation (Dong et al. 2023) for training.

Method	Part segmentation		Semantic segmentation	
	mIoU_c	mIoU_i	mACC	mIoU
PointNet	80.4	83.7	49.0	41.1
PointNet++	81.9	85.1	67.1	53.5
PointBERT	84.1	85.6	69.7	60.5
PointMAE	84.2	86.1	69.9	60.8
PointGPT-S	84.1	86.2	-	-
Ours (T)	**84.4**	**86.4**	**70.6**	**61.2**
PointMamba	84.4	86.2	-	-
Mamba3D	83.6	85.6	-	-
Ours (M)	**84.6**	**86.4**	**70.2**	**60.6**

Table 2: Segmentation results. For ShapeNetPart, we report the mean category IoU (mIoU_c) and the mean instance IoU (mIoU_i). For S3DIS, we report the mean instance IoU (mIoU) and the mean class accuracy (mAcc).

Method	5-way		10-way	
	10-shot	20-shot	10-shot	20-shot
PointBERT	94.6±3.6	93.9±3.1	86.4±5.4	91.3±4.6
MaskPoint	95.0±3.7	97.2±1.7	91.4±4.0	92.7±5.1
PointMAE	96.3±2.5	97.8±1.8	92.6±4.1	93.4±3.5
PointM2AE	96.8±1.8	98.3±1.4	92.3±4.5	95.0±3.0
PointGPT-S	96.8±2.0	**98.6**±1.1	92.6±4.6	95.2±3.4
Ours (T)	**97.1**±2.2	**98.6**±1.3	**92.8**±4.4	**95.3**±3.3
PointMamba	96.9±2.0	**99.0**±1.1	93.0±4.4	95.6±3.2
Mamba3D	96.4±2.2	98.2±1.2	92.4±4.1	95.2±2.9
Ours (M)	**97.3**±2.1	98.8±1.2	**93.1**±4.2	**95.8**±3.2

Table 3: Few-shot classification on ModelNet40. We report overall accuracy and standard deviation.

12,311 3D CAD models across 40 categories. As shown in Tab. 1, our proposed PointChain achieves accuracies of 94.7% and 95.2% with Transformer and Mamba backbones, respectively, outperforming existing self-supervised methods (Chen et al. 2023; Han et al. 2024). In line with previous works, we also report the model size and FLOPs on the classification task. Thanks to the unidirectional structural chain design, our method achieves competitive performance with significantly lower computational overhead. This balance between efficiency and accuracy highlights the strong representation learning capability of our method.

Part Segmentation. We evaluate part segmentation on the ShapeNetPart (Yi et al. 2016) dataset, which requires assigning a part-level label to each point. The dataset consists of 16 object categories with 16,880 models and a total of 50 part-level labels. As shown in Tab. 2, although ShapeNetPart is a saturated benchmark with some annotation errors, our method surpasses previous methods, achieving a significant performance improvement.

Semantic Segmentation. To further evaluate the semantic representation capability, we conduct semantic segmentation experiments on the S3DIS dataset (Armeni et al. 2016). S3DIS contains 271 indoor scenes from 6 areas, annotated with 13 semantic categories. We follow the standard protocol by using Area 5 for testing and the remaining areas for training. The processing pipeline mirrors that of the part segmentation experiments. As shown in Tab. 2, our models based on Transformer and Mamba outperform previous methods. The result show the effectiveness of our paradigm in capturing contextual information and semantic features.

Few-Shot Learning. We further evaluate our method on few-shot learning using ModelNet40 to assess its effectiveness under limited data conditions. Following standard procedures (Pang et al. 2022), we perform 10 independent runs for each setting and report the average accuracy with standard deviation. As shown in Tab. 3, even under limited

Method	PB-T50-RS
Baseline	90.11
w/ PointNet++-like Embedding	90.53
w/ Structural Chain & Next-Scale Prediction	92.96
w/ Cross-Scale Semantic Alignment	**93.75**

Table 4: Effect of the designed modules.

MSTE	SI	DSS	PB-T50-RS
✗	✗	✗	92.26
✓	✗	✗	92.75
✓	✓	✗	93.34
✓	✓	✓	**93.75**

Table 5: Effect of structural chain reasoning. 'MSTE' denotes multi-scale token embedding, 'SI' indicates the scale identifier, and 'DSS' denotes dynamic scaling strategy.

data conditions, our PointChain achieves competitive performance, demonstrating excellent generalization ability.

4.3 Ablation Study

Below, we verify the key designs of PointChain. All experiments are conducted using a Mamba-based architecture on splits of ScanObjectNN without voting strategy.

Effect of the Designed Modules. As shown in Tab. 4, we incrementally add each designed module to a Mamba-based baseline to evaluate their contributions. The baseline is built on the standard Mamba (Gu and Dao 2023) block and follows the same architecture as PointGPT-S (Chen et al. 2023). Replacing the PointNet-like embedding with a PointNet++-like design yields limited improvement, indicating that enhancing structural representation alone is insufficient for downstream gains. In contrast, significant gains are achieved when the structural chain and the next-scale prediction pretraining task are introduced, confirming that the structural chain enables effective coarse-to-fine reasoning and improves the ability to capture the hierarchical structure of point clouds. Finally, introducing the scale-level contrastive learning task explicitly aligns semantic features across scales, further enhancing the ability to capture complex spatial structures and semantic relationships, and leading to the best overall performance.

Effect of Structural Chain Reasoning. Tab. 5 presents the ablation study on structural chain reasoning. Introducing multi-scale token embedding yields moderate improvements, highlighting the benefits of hierarchical feature modeling and providing a foundation for coarse-to-fine reasoning. Adding the scale identifier further enhances performance by enabling the model to distinguish features across different scales. Finally, the dynamic scaling strategy, which normalizes point clouds to a consistent radius range for scale-invariant prediction, ensures stable training of the next-scale prediction task. The combination of all components delivers the best performance, highlighting their complementary contributions to structural reasoning.

Effect of Pre-training Tasks. To verify the effect of the proposed pre-training tasks, we conduct ablation studies as shown in Tab. 6. Compared with the MPM task, our next-scale prediction task better captures the hierarchical structure of point clouds, leading to improved performance. Further incorporating the cross-scale semantic alignment task yields additional gains by promoting semantic consistency across different scales. We also investigate the impact of the regularization term in Eq. (8) by varying its loss weight λ during fine-tuning. Since setting $\lambda = 0$ (i.e., removing the regularization) causes premature forgetting and degrades generalization, we ultimately adopt $\lambda = 2$ to balance preserving pre-trained knowledge and downstream adaptation.

MPM	NSP	CSSA	λ	PB-T50-RS
✓	✗	✗	2	92.09
✗	✓	✗	2	92.96
✗	✓	✓	2	**93.75**
✗	✓	✓	0	91.74
✗	✓	✓	4	93.44

Table 6: Effect of pre-training paradigm. 'NSP' denotes next-scale prediction, 'CSSA' indicates cross-scale semantic alignment, and λ is weight of regularization term in Eq. (8).

Numbers	OBJ-BG	OBJ-ONLY	PB-T50-RS
[1 16 64]	94.84	93.80	**93.75**
[1 32 96]	95.01	**94.49**	93.37
[1 32 128]	**95.70**	94.15	93.72
[1 64 256]	94.66	93.80	93.03

Table 7: Ablation on the number of representative points $[G_0, G_1, G_2]$ used in the structural chain.

Ablation on Structural Chain Token Number. To enable progressive structural modeling, we construct a structural chain using multi-scale representative points. Tab. 7 presents the ablation results for different combinations of representative points $[G_0, G_1, G_2]$ used in the chain. All four configurations achieve consistently strong performance, demonstrating the robustness of our method across different scales. Considering the trade-off between computational efficiency and accuracy, and to ensure fair comparison with existing methods, we adopt $[1, 32, 128]$ for OBJ-BG and OBJ-ONLY, and $[1, 16, 64]$ for PB-T50-RS. Additional implementation details are provided in the supplementary material.

5 Conclusion

In this paper, we propose PointChain, a novel autoregressive paradigm that models 3D point clouds as structural chains. Specifically, we draw inspiration from human perception mechanisms to design a structural chain encoding for coarse-to-fine point cloud understanding. Furthermore, we introduce next-scale prediction and cross-scale semantic alignment tasks, which significantly enhance the discriminative and generalization capabilities of the pre-trained models. Extensive experiments on both Transformer and Mamba validate the effectiveness of our method, achieving state-of-the-art performance on multiple benchmarks.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (Grant No. 2024YFB3909902) and the Youth Innovation Promotion Association of the Chinese Academy of Sciences. We gratefully acknowledge their financial support, which made this research possible.

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 *Proceedings of the IEEE conference on CVPR*, 1534–1543.

Bao, H.; Dong, L.; Piao, S.; and Wei, F. 2022. BEiT: BERT Pre-Training of Image Transformers. In *International Conference on Learning Representations*.

Chang, A. X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*.

Chang, H.; Zhang, H.; Barber, J.; Maschinot, A.; Lezama, J.; Jiang, L.; Yang, M.-H.; Murphy, K.; Freeman, W. T.; Rubinstein, M.; et al. 2023. Muse: Text-to-image generation via masked generative transformers. In *Proceedings of the 40th International Conference on Machine Learning*, 4055–4075.

Chang, H.; Zhang, H.; Jiang, L.; Liu, C.; and Freeman, W. T. 2022. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on CVPR*, 11315–11325.

Chen, G.; Wang, M.; Yang, Y.; Yu, K.; Yuan, L.; and Yue, Y. 2023. Pointgpt: Auto-regressively generative pre-training from point clouds. *Advances in Neural Information Processing Systems*, 36: 29667–29679.

Chen, M.; Radford, A.; Child, R.; Wu, J.; Jun, H.; Luan, D.; and Sutskever, I. 2020a. Generative pretraining from pixels. In *International conference on machine learning*, 1691–1703. PMLR.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020b. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Dao, T.; and Gu, A. 2024. Transformers are SSMs: Generalized Models and Efficient Algorithms Through Structured State Space Duality. *Proceedings of Machine Learning Research*, 235: 10041–10071.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 4171–4186.

Dong, R.; Qi, Z.; Zhang, L.; Zhang, J.; Sun, J.; Ge, Z.; Yi, L.; and Ma, K. 2023. Autoencoders as Cross-Modal Teachers: Can Pretrained 2D Image Transformers Help 3D Representation Learning? In *The Eleventh International Conference on Learning Representations*.

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on CVPR*, 12873–12883.

Fan, H.; Su, H.; and Guibas, L. J. 2017. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on CVPR*, 605–613.

Gao, T.; Yao, X.; and Chen, D. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Gu, A.; and Dao, T. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Gu, A.; Goel, K.; and Ré, C. 2021. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*.

Gu, A.; Johnson, I.; Goel, K.; Saab, K.; Dao, T.; Rudra, A.; and Ré, C. 2021. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34: 572–585.

Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; and Bennamoun, M. 2020. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12): 4338–4364.

Han, X.; Tang, Y.; Wang, Z.; and Li, X. 2024. Mamba3d: Enhancing local features for 3d point cloud analysis via state space model. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 4995–5004.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on CVPR*, 16000–16009.

Kalman, R. E. 1960. A new approach to linear filtering and prediction problems. *Trans. ASME, D*, 82: 35–44.

Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; and Chen, B. 2018. Pointcnn: Convolution on x-transformed points. *Advances in neural information processing systems*, 31.

Li, Y.; Ma, L.; Zhong, Z.; Liu, F.; Chapman, M. A.; Cao, D.; and Li, J. 2020. Deep learning for lidar point clouds in autonomous driving: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8): 3412–3432.

Liang, D.; Zhou, X.; Xu, W.; Zhu, X.; Zou, Z.; Ye, X.; Tan, X.; and Bai, X. 2024. Pointmamba: A simple state space model for point cloud analysis. *Advances in neural information processing systems*, 37: 32653–32677.

Liu, H.; Cai, M.; and Lee, Y. J. 2022. Masked discrimination for self-supervised learning on point clouds. In *European Conference on Computer Vision*, 657–675. Springer.

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 *International Conference on Learning Representations*.

Morton, G. M. 1966. *A computer oriented geodetic data base and a new technique in file sequencing*. International Business Machines Company.

Navaneet, K.; Mathew, A.; Kashyap, S.; Hung, W.-C.; Jampani, V.; and Babu, R. V. 2020. From image collections to point clouds with self-supervised shape and pose networks. In *Proceedings of the IEEE/CVF Conference on CVPR*, 1132–1140.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Pang, Y.; Wang, W.; Tay, F. E.; Liu, W.; Tian, Y.; and Yuan, L. 2022. Masked Autoencoders for Point Cloud Self-supervised Learning. In *European Conference on Computer Vision*, 604–621. Springer.

Pomerleau, F.; Colas, F.; Siegwart, R.; et al. 2015. A review of point cloud registration algorithms for mobile robotics. *Foundations and Trends® in Robotics*, 4(1): 1–104.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on CVPR*, 652–660.

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

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

Qian, R.; Meng, T.; Gong, B.; Yang, M.-H.; Wang, H.; Belongie, S.; and Cui, Y. 2021. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF conference on CVPR*, 6964–6974.

Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I.; et al. 2018. Improving language understanding by generative pre-training. OpenAI technical report.

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 *Proceedings of the IEEE/CVF international conference on computer vision*, 6411–6420.

Tian, K.; Jiang, Y.; Yuan, Z.; Peng, B.; and Wang, L. 2024. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37: 84839–84865.

Uy, M. A.; Pham, Q.-H.; Hua, B.-S.; Nguyen, T.; and Yeung, S.-K. 2019. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1588–1597.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30: 5998–6008.

Wang, C.; Deng, J.; He, J.; Zhang, T.; Zhang, Z.; and Zhang, Y. 2023. Long-short range adaptive transformer with dynamic sampling for 3d object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(12): 7616–7629.

Wang, C.; Yang, W.; Liu, X.; and Zhang, T. 2025a. State Space Model Meets Transformer: A New Paradigm for 3D Object Detection. In *The Thirteenth International Conference on Learning Representations*.

Wang, C.; Zha, Y.; He, J.; Yang, W.; and Zhang, T. 2024. Rethinking Masked Representation Learning for 3D Point Cloud Understanding. *IEEE Transactions on Image Processing*.

Wang, C.; Zha, Y.; Yang, W.; and Zhang, T. 2025b. StruMamba3D: Exploring Structural Mamba for Self-supervised Point Cloud Representation Learning. *arXiv preprint arXiv:2506.21541*.

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 *Proceedings of the IEEE conference on CVPR*, 1912–1920.

Xie, S.; Gu, J.; Guo, D.; Qi, C. R.; Guibas, L.; and Litany, O. 2020. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *European conference on computer vision*, 574–591. Springer.

Yang, Y.; Feng, C.; Shen, Y.; and Tian, D. 2018. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE conference on CVPR*, 206–215.

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. *ACM Transactions on Graphics (ToG)*, 35(6): 1–12.

Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. SeqGAN: sequence generative adversarial nets with policy gradient. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2852–2858.

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 *Proceedings of the IEEE/CVF conference on CVPR*, 19313–19322.

Zha, Y.; Wang, C.; Yang, W.; and Zhang, T. 2025. Exploring Semantic Masked Autoencoder for Self-supervised Point Cloud Understanding. *arXiv preprint arXiv:2506.21957*.

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. *Advances in neural information processing systems*, 35: 27061–27074.

Zhang, T.; Yuan, H.; Qi, L.; Zhang, J.; Zhou, Q.; Ji, S.; Yan, S.; and Li, X. 2025. Point cloud mamba: Point cloud learning via state space model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 10121–10130.

Zhang, Z.; Girdhar, R.; Joulin, A.; and Misra, I. 2021. Self-supervised pretraining of 3d features on any point-cloud. In *Proceedings of the IEEE/CVF international conference on computer vision*, 10252–10263.

Zhou, X.; Liang, D.; Xu, W.; Zhu, X.; Xu, Y.; Zou, Z.; and Bai, X. 2024. Dynamic adapter meets prompt tuning: Parameter-efficient transfer learning for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on CVPR*, 14707–14717.