Information Sciences 744 (2026) 123378

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Point-DPA: Unifying contrastive and generative learning for 3D point cloud understanding via dynamic prototypes

Xin Cao [a,b], Xinmeng Hu [a,b], Yinan Wang [a,b], Kang Li [a,b], Linzhi Su [a,b,*], Yangyang Liu [a,b], Fengjun Zhao [a,b,*]

[a] College of Computer Science, Northwest University, Xi'an, China
[b] School of Information Science & Technology, Northwest University, Xi'an, China

HIGHLIGHTS

- Unifies contrastive-generative learning for self-supervised 3D learning.
- First to introduce Masked Dynamic Prototype Alignment for point cloud SSL.
- Employs synergistic dual-alignment objectives for robust representations.
- Features Teacher-Student architecture with Evolving Prototype Memory.

ARTICLE INFO

Keywords:
Self-supervised learning
Masked dynamic prototype alignment
Point clouds
Representation learning
Teacher-student architecture

ABSTRACT

Self-supervised Learning (SSL) has emerged as a dominant paradigm for 3D point cloud understanding. However, existing methods still leave room for improvement in several key aspects: contrastive learning effectively captures global invariance but suffers from a 'granularity gap' by overlooking fine-grained local structures, while Masked Point Modeling (MPM) typically relies on semantic-poor coordinate reconstruction, forcing models to focus on high-frequency spatial noise rather than abstract semantics. Furthermore, the technical challenge of relying on complex offline tokenizers leads to significant pipeline inefficiency and potential domain gaps. To bridge these gaps and challenges, we propose Point-DPA, a unified single-stage framework via Masked Dynamic Prototype Alignment. Unlike coordinate-based approaches or those requiring complex offline tokenizers, we design an asymmetric Teacher-Student architecture where the Teacher maintains an Evolving Prototype Memory to dynamically update high-level semantic targets online. Specifically, the Student network is tasked with predicting spatially dense prototype distributions from masked inputs through a synergistic dual-objective mechanism: Cross-view Global Alignment to enforce instance-level invariance, and Local Patch Alignment to ensure robust structural reasoning. Extensive experiments demonstrate that Point-DPA achieves a competitive classification accuracy of 93.52% on ModelNet40 and establishes a new state-of-the-art of 87.03% on the real-world ScanObjectNN dataset. This significant improvement confirms that our method learns robust semantic representations resilient to noise, effectively overcoming the limitations of previous reconstruction-based paradigms. The source code and pre-trained models will be made publicly available upon publication.

* Corresponding authors at: College of Computer Science, Northwest University, Xi'an, China.
Email addresses: xin_cao@163.com (X. Cao), sulinzhi029@163.com (L. Su), fjzhao@nwu.edu.cn (F. Zhao).

https://doi.org/10.1016/j.ins.2026.123378
Received 26 January 2026; Received in revised form 15 March 2026; Accepted 16 March 2026
Available online 18 March 2026
0020-0255/© 2026 Elsevier Inc. All rights reserved, including those for text and data mining, AI training, and similar technologies.

1. Introduction

Point clouds have emerged as a ubiquitous and precise data representation for digitizing real-world objects and environments. While fully supervised methods have achieved remarkable success in 3D understanding, they fundamentally rely on large-scale annotated datasets. Collecting such annotations for 3D data is notoriously labor-intensive and time-consuming due to the sparse, unordered, and irregular nature of point clouds. Consequently, Self-Supervised Learning (SSL) has garnered significant attention, aiming to cultivate robust representations from unlabeled data by leveraging intrinsic data structures.

Drawing inspiration from the transformative successes of self-supervised learning in Natural Language Processing [1] and 2D computer vision [2], the research landscape of 3D point cloud understanding has primarily bifurcated into two dominant paradigms: contrastive learning and generative masked modeling. Contrastive learning approaches operate on the principle of instance discrimination. They aim to learn invariant representations by maximizing the feature similarity between two augmented views of the same object while pushing away features from different objects. Representative methods like PointContrast [3] exploit point-level consistency across views, whereas CrossPoint [4] enforces invariance to affine transformations at the object level. While these methods are effective for high-level classification tasks, they inherently suffer from a "granularity gap." By prioritizing global invariance, they tend to collapse fine-grained local details into holistic descriptors, rendering them suboptimal for dense prediction tasks like segmentation. Moreover, their performance heavily depends on complex, manually engineered data augmentations, limiting their scalability and cross-domain generalization.

Conversely, generative methods, particularly Masked Point Modeling (MPM), have emerged as a powerful alternative by mimicking the 'mask-and-predict' pretext task of BERT. These methods encourage the model to reason about the underlying shape structure by inferring masked regions from visible contexts. For instance, Point-MAE [5] employs K-Nearest Neighbor (KNN) grouping to mask random patches and tasks the decoder with reconstructing the raw geometric coordinates of the missing points. However, we argue that this reliance on low-level coordinate reconstruction is inherently flawed: it compels the model to fixate on fitting high-frequency spatial noise and local geometric interpolation, rather than capturing abstract semantic concepts. To address this, Point-BERT [6] attempts to introduce discrete semantic tokens via a pre-trained discrete variational autoencoder (dVAE). Yet, this introduces a complex, multi-stage pre-training pipeline where the tokenizer is trained independently, leading to potential domain gaps and significant computational inefficiency.

To bridge these gaps, we critically re-examine the reconstruction targets in masked modeling. We posit that an ideal pretext task should simultaneously enforce perturbation invariance (akin to contrastive learning) and dense contextual reasoning (akin to MPM) within a high-level semantic space. In this work, we propose Point-DPA, a unified single-stage self-supervised framework driven by Masked Dynamic Prototype Alignment. Adopting a paradigm that combines masking with prototype learning, Point-DPA departs from coordinate-based approaches by predicting dynamic prototype distributions rather than performing raw reconstruction. We employ an asymmetric Teacher-Student architecture where the Teacher maintains an Evolving Prototype Memory—a dynamic dictionary of visual patterns that updates online. For a given point cloud, the Teacher assigns local patches to these prototypes to generate "alignment vectors" as semantic targets. The Student, processing a masked input, is tasked with predicting these high-level targets. This design forces the network to capture the semantic essence of object components (e.g., recognizing a "table leg" pattern) rather than their exact spatial coordinates.

Our approach offers several distinct advantages. First, by predicting Prototype Alignment Vectors, we elevate the learning objective from geometric regression to semantic classification, effectively unifying the strengths of contrastive and generative learning. Second, we introduce a synergistic dual-objective mechanism: a Cross-view Global Alignment task to learn instance invariance, and a Local Patch Alignment task to reconstruct fine-grained semantic codes. Finally, our dynamic prototype mechanism eliminates the need for pre-training offline tokenizers, significantly streamlining the training process. In summary, the main contributions of our work are as follows:

- We propose a novel Self-Supervised Point Cloud Learning framework, Point-DPA, which integrates contrastive invariance and generative reasoning via Masked Dynamic Prototype Alignment.
- We design a synergistic dual-objective mechanism that unifies contrastive and generative learning. By enforcing Cross-view Global Alignment (utilizing dual augmented views for invariance) and Local Patch Alignment (for structural reasoning), our model captures both discriminative global features and fine-grained local details.
- Extensive experiments on ModelNet40 [7], ScanObjectNN [8], ShapeNetPart [9], and S3DIS [10] demonstrate that our method outperforms state-of-the-art methods on various downstream tasks while maintaining high training efficiency.

2. Related work

2.1. Deep learning on point clouds

The fundamental challenge in 3D deep learning lies in the data representation itself. Unlike 2D images, which possess a regular grid structure suitable for standard convolution operations, 3D point clouds are sparse, unordered, and irregularly distributed in continuous space. Developing effective backbones to extract robust features from such irregular data has been a central focus of the computer vision community.

Early research focused on point-based architectures that process raw points directly. The pioneering work, PointNet [11], revolutionized the field by utilizing shared Multi-Layer Perceptrons (MLPs) to extract point-wise features and employing a symmetric function to aggregate global information, thereby achieving permutation invariance. Building upon this, PointNet++ [12] addressed the lack of local context in PointNet by introducing a hierarchical grouping strategy. It applies PointNet recursively to nested partitions of the input point set, effectively capturing local geometric structures at different scales. Subsequent works have further refined these designs to better capture geometric topology. For instance, PointCNN [13] learns a χ - transformation to reorder points into a latent canonical order for convolution, while PointMLP [14] demonstrates that a pure residual MLP architecture, when equipped with geometric affine modules, can achieve competitive performance without complex local aggregators. Additionally, CurveNet [15] takes a distinct approach by aggregating features via guided walks to capture curve-like geometric features. Another prominent stream of research treats point clouds as graph structures. DGCNN [16] introduced the EdgeConv operation, which dynamically constructs graphs in the feature space rather than the coordinate space. This allows the network to learn semantic relationships between points that may be geometrically distant but semantically similar, updating the graph structure at every layer to capture evolving local geometric structures.

Like the network evolution of image processing, Transformer-based architectures have recently surged to the forefront of 3D vision, celebrated for their superior capacity to model long-range dependencies. Given that point clouds are inherently unordered sets equipped with positional attributes, they are naturally compatible with the permutation-invariant structure of self-attention mechanisms. Pioneering works such as PCT [17] have successfully adapted these mechanisms to the 3D domain, utilizing offset-attention and vector-based self-attention to capture complex geometric structures. Similarly, LCASAFormer [18] introduces a cross-attention mechanism to further enhance the seamless integration of local features with global contexts. Building upon these advancements, we adopt a standard Transformer encoder backbone in this work to fully capitalize on its robust sequence modeling capabilities for capturing semantic context.

2.2. Self-supervised point cloud learning

Self-Supervised Learning (SSL) has emerged as a predominant paradigm for acquiring robust feature representations from unlabeled data, effectively mitigating the reliance on expensive and time-consuming manual 3D annotations. By leveraging intrinsic data structures, SSL facilitates the learning of transferable features that generalize well to downstream tasks. Driven by the transformative success of SSL in natural language processing and 2D computer vision, the 3D vision community has primarily converged on two main paradigms: contrastive learning and masked point modeling.

Contrastive learning approaches focus on instance discrimination to learn representation invariance. This paradigm largely originates from 2D vision, where the method such as SimCLR [19] learn global invariance by maximizing the similarity between augmented views of the same image while pushing away negative samples. Furthermore, methods like Barlow Twins [20] demonstrated that negative pairs are not strictly necessary by utilizing asymmetric networks or redundancy reduction objectives. Adapting these principles to the 3D domain, PointContrast [3] extends this concept by utilizing point-level correspondences across different views to learn discriminative features. CrossPoint [4] further enhances invariance by enforcing consistency between 3D point clouds and their corresponding 2D rendered images, leveraging cross-modal signals. Similarly, STRL [21] explores spatio-temporal reasoning by interacting with an online network and a target network.

Parallel to contrastive methods, Masked Point Modeling (MPM), is motivated by the scalability of generative models. Following BERT [1], methods like BEiT [22] and SimMIM [23] mask a large portion of image patches and reconstruct missing content. These methods operate on the principle of “masking and reconstruction,” forcing the network to infer occluded regions from visible context. Point-BERT introduced this paradigm to 3D by using a pre-trained discrete Variational Autoencoder (dVAE) to tokenize point clouds, training a Transformer to predict missing tokens. To simplify this pipeline, Point-MAE proposed a fully end-to-end framework that masks random patches and reconstructs their raw coordinates directly. Variations such as Point-M2AE [24] introduce multi-scale masking to capture hierarchical geometries, while OcCo [25] focuses specifically on occlusion completion. Other works like Point-GAME [26] explore adaptive masking strategies or discrimination tasks, and ACT [27] utilizes autoencoders to learn latent features. Distinct from these existing paradigms, we propose a self-supervised learning scheme that integrates both contrastive learning and masked modeling for point clouds.

2.3. Prototype-based representation learning

To bridge the dichotomy between semantically impoverished low-level reconstruction and locally insensitive instance discrimination, Prototype-based Learning has emerged as a compelling alternative. Exemplified by pioneering works like MOCA [28] in 2D vision, this paradigm transitions the learning objective from pixel-wise reconstruction to predicting discrete codes or probability distributions over a dynamically updated prototype codebook.

Despite its success in 2D, dynamic prototype alignment remains underexplored in the 3D domain. Existing 3D methods face a dilemma: either rely on semantic-poor coordinate regression or complex static tokenizers. To address these limitations, we propose Point-DPA. Unlike Point-MAE, we do not reconstruct coordinates; unlike Point-BERT, we do not require a pre-trained static tokenizer. Instead, we introduce an Evolving Prototype Memory within a Teacher-Student architecture to dynamically learn high-level semantic concepts online. By masking input patches and enforcing the Student to predict the Teacher’s dynamic prototype assignments, we unify Global Contrastive Invariance and Local Generative Reasoning in a single-stage framework. This allows the model to capture both fine-grained structure and high-level semantics efficiently.

3. Method

As illustrated in Fig. 1, Point-DPA operates as an asymmetric Teacher-Student framework, unifying contrastive invariance and generative reasoning via Masked Dynamic Prototype Alignment. The pipeline comprises four integral stages: Dual-View Point Tokenization, a Teacher branch with Evolving Prototype Memory, a Student branch with dynamic prediction heads and the synergistic Dual-Alignment Objectives.

3.1. Dual-view point tokenization and asymmetric masking

We construct a data pipeline that transforms raw point clouds into masked semantic tokens, ensuring robustness to geometric transformations while preserving fine-grained structure. This process involves three sequential stages: dual-view generation, patch tokenization, and asymmetric masking.

Dual-View Input and Stochastic Augmentation: Given a batch of input point clouds $P \in \mathbb{R}^{B\times N\times 3}$, where B is the batch size and N is the number of points (typically 1024), we aim to generate two correlated views, P_{v1} and P_{v2}, via independent stochastic transformations. This is formulated as:

$$P_{v1} = \mathcal{T}_1(P), \quad P_{v2} = \mathcal{T}_2(P) \tag{1}$$

where $\mathcal{T}_1(\cdot)$ and $\mathcal{T}_2(\cdot)$ are independent augmentations sampled from a composite family $\mathcal{T}$. To maximize data diversity and invariance, $\mathcal{T}$ applies three operations sequentially:

- *Random Scaling:* Object coordinates are scaled by a factor $s \sim \mathcal{U}[0.8, 1.25]$ to enforce scale invariance.
- *Random Rotation:* A rotation matrix $R_z(\theta)$ is applied around the gravity axis (Z-axis), with $\theta \sim [0, 2\pi]$, enabling orientation-agnostic recognition.
- *Random Jittering:* Gaussian noise $\mathcal{N}(0, 0.01)$ is injected into each point (clamped to $[-0.05, 0.05]$) to simulate sensor noise and enhance robustness against local surface perturbations.

Patch Generation via Grouping: Unlike regular 2D image grids, 3D point clouds are spatially sparse and unordered. As illustrated in Fig. 2, we first employ Farthest Point Sampling (FPS) to select a subset of G center points $C = \{c_i\}_{i=1}^{G}$, ensuring uniform coverage of the underlying shape. For each center c_i, we query its K-nearest neighbors via the KNN algorithm to form a local patch $x_i = \{p_{i,j}\}_{j=1}^{K} \in \mathbb{R}^{K\times 3}$. To achieve local translation invariance, coordinates within each patch are normalized relative to their center: $p'_{i,j} = p_{i,j} - c_i$. These normalized patches are then mapped to high-dimensional embeddings via a lightweight MLP (mini-PointNet) and summed element-wise with positional embeddings $E_{\text{pos}}(c_i)$ to preserve global spatial context.

Asymmetric Dual-View Masking Strategy: We employ a random masking strategy $\mathcal{M}$ to partition the G patches into a visible subset $\mathcal{V}$ and a masked subset $\mathcal{M}$. We adopt a high masking ratio ($\rho = 0.75$) to reduce spatial redundancy, compelling the model to infer complete shapes from sparse cues. Crucially, to prevent the network from learning trivial solutions by merely "copying" features between views, we enforce an *asymmetric masking* pattern. Specifically, the masking indices for view1 (M_{v1}) are generated to be distinct from those of view2 (M_{v2}). This non-overlapping design eliminates short-cut learning and forces the Student network to perform holistic structural reasoning to recover missing semantics from partial, non-corresponding observations.

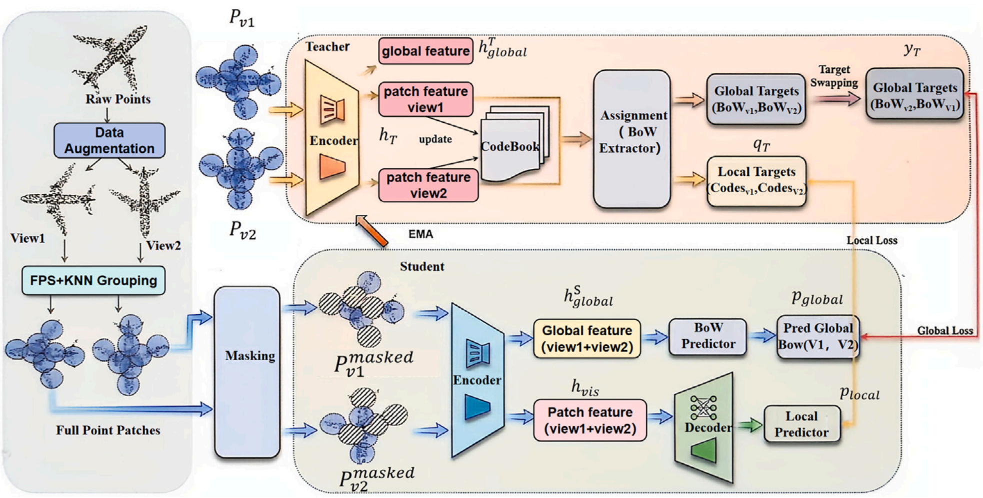

Fig. 1. The overall architecture of Point-DPA. We employ a Teacher-Student design to perform masked dynamic Prototype Alignment. Teacher branch (Top): utilizes an Evolving Prototype Memory to compute the global targets and local targets. Student branch (Bottom): processes masked patches and is trained with two complementary objectives: (1) Cross-view global Alignment, which uses target swapping to learn global invariance; and (2) local patch Alignment, which enforces the reconstruction of local semantic prototypes to capture fine-grained structure.

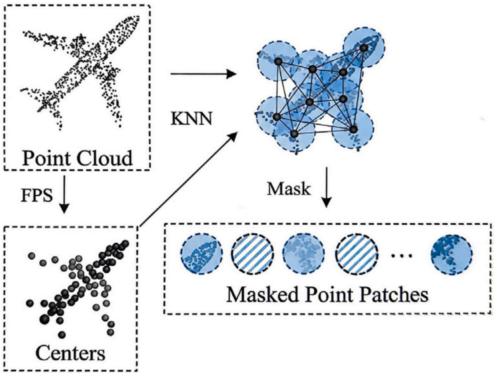

Fig. 2. Patch Tokenization and Masking. The input point cloud is discretized into local patches via FPS sampling and KNN grouping to capture local geometry. Subsequently, an asymmetric masking strategy is applied to generate sparse inputs for the Student network.

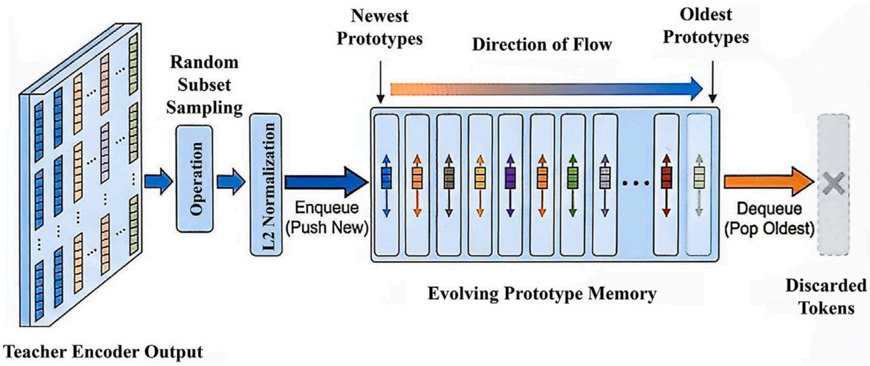

Fig. 3. The Evolving Prototype Memory. Designed as a FIFO queue, this dynamic codebook updates online to track the current feature distribution. New patch tokens from the Teacher branch are enqueued to refresh the memory, while outdated prototypes are dequeued, ensuring the codebook remains a "living" dictionary of high-level semantic patterns without requiring static pre-computation.

3.2. Teacher branch: evolving prototype memory

The Teacher branch provides stable, high-level semantic guidance via an Evolving Prototype Memory, eliminating the need for static, pre-defined tokenizers. This branch features a momentum teacher and latent representation anchor, an online evolving codebook, and a hierarchical target generation mechanism.

Momentum Teacher and Latent Representation Anchor: The Teacher network, parameterized by θ_T, mirrors the Student's architecture but functions as a stable target generator. To ensure semantic consistency and prevent solution collapse, θ_T is decoupled from gradient descent; instead, it evolves via an Exponential Moving Average (EMA) of the Student parameters θ_S:

$$\theta_T \leftarrow \alpha\theta_T + (1-\alpha)\theta_S \tag{2}$$

where $\alpha \in [0, 1)$ is the momentum coefficient. Based on these stable parameters, the Teacher encoder processes the input to generate a sequence of patch features h_T. The sequence of patch features h_T is aggregated using both Max-Pooling and Mean-Pooling, which generates the Teacher branch's internal global representation h^T_{global}:

$$h^T_{\text{global}} = \phi\left([\text{Max}(h_T) \parallel \text{Mean}(h_T)]\right) \tag{3}$$

Within our asymmetric framework, h^T_{global} serves as a latent semantic anchor that represents a holistic semantic descriptor of the complete object. It is important to distinguish this internal feature anchor from the eventual alignment targets. While h^T_{global} captures the high-level geometric essence of the full shape, the actual supervision for the global alignment task is provided by the semantic distribution (Bag-of-Words) derived from prototype assignments. The h^T_{global} acts solely as an intermediate descriptor and does not directly participate in the final loss computation.

Online Codebook Construction (FIFO Queue): To circumvent the complexity and domain constraints of offline tokenizers (e.g., dVAE), we introduce an Evolving Prototype Memory. As depicted in Fig. 3, this is implemented as a dynamic codebook $C = \{c_k\}_{k=1}^K \in \mathbb{R}^{K\times D}$, operating as a First-In-First-Out (FIFO) queue. At each iteration, a subset of patch tokens is stochastically sampled from the Teacher's output, L_2-normalized, and enqueued into C, displacing the oldest entries. This mechanism ensures that the prototypes continuously adapt to the current data distribution, maintaining a diverse and up-to-date dictionary of semantic patterns (e.g., planar surfaces, curvilinear edges).

Hierarchical Target Generation: The Teacher generates alignment targets at both the local and global levels to guide the Student's learning. First, for *Local Patch Alignment*, we compute the cosine similarity between each teacher patch feature $h_{T,i}$ and the prototypes in C. This yields a soft assignment distribution, denoted as the Local Target $q_{T,i}$:

$$q_{T,i} = \text{Softmax}\left(\frac{h_{T,i} \cdot C^\top}{\tau_t}\right) \tag{4}$$

where τ_t is a dynamic temperature parameter stabilized by the running average of the distance distribution. Subsequently, for *Global Alignment*, we aggregate these local semantics to form a holistic descriptor. Specifically, we apply Mean Pooling over the sequence of local distributions $\{q_{T,i}\}_{i=1}^{G}$ to compute the Global Target y_T (a Bag-of-Words vector):

$$y_T = \frac{\sum_{i=1}^{G} q_{T,i}}{\left\|\sum_{i=1}^{G} q_{T,i}\right\|_1} \tag{5}$$

This explicit L_1 normalization ensures numerical stability, resulting in a valid probability distribution that represents the global semantic composition of the object.

3.3. Student branch: efficient encoding and dynamic prediction

The Student branch learns robust representations by reconstructing the Teacher's semantic targets from partial observations. Key components include an efficient encoder with an Early Masking strategy, a lightweight decoder, and dynamic prediction heads.

Efficient Encoder via Early Masking Strategy: To optimize computational efficiency, the Student encoder E_S adopts an Early Masking strategy. Unlike standard BERT-style Transformers that replace masked tokens with learnable embeddings and process the full sequence length G, our Student physically discards the masked patches at the input stage and processes only the visible subset $\mathcal{V}$. The encoder consists of L blocks of Multi-head Self-Attention (MSA) and Feed-Forward Networks (FFN). The complexity of MSA is quadratic to the sequence length. By discarding masked patches, we reduce the sequence length by approximately 75% (given a mask ratio of 0.75). Through the forward propagation of the Student encoder, the patch features h_{vis} are generated. The Student network adopts the identical aggregation logic from Eq. (3) to generate its global representation from sparse observations:

$$h^S_{\text{global}} = \phi\left([\text{Max}(h_{\text{vis}}) \parallel \text{Mean}(h_{\text{vis}})]\right) \tag{6}$$

Given the extreme masking ratio ($\rho = 0.75$), the primary challenge for the Student is to infer the complete object's structure from highly sparse and fragmented cues. By applying the Dual-Pooling strategy—combining Max-Pooling to capture salient geometric primitives and Mean-Pooling to capture holistic statistical features—the Student produces a robust h^S_{global} that serves as the functional basis for its global reasoning. Only by relying on this enriched contextual information can the student-side's prediction head accurately infer the corresponding global semantic distribution from the Teacher branch.

Lightweight Decoder for Structure Reconstruction: To support the local alignment task, we employ a lightweight Transformer Decoder. The input to the decoder is constructed by concatenating the encoded visible features h_{vis} with learnable Mask Tokens $m \in \mathbb{R}^{|\mathcal{M}|\times D}$. Crucially, full positional embeddings are added to this combined sequence to inform the decoder about the absolute spatial locations of the missing patches. The decoder outputs latent vectors h_{dec} corresponding to the masked regions.

Dynamic Prediction Heads via Residual Weight Generation: A critical innovation in our architecture is the use of Dynamic Prediction Heads. Conventional linear classifiers rely on static weights W, which inevitably misalign with the Evolving Prototype Memory C as it updates online. To address this, we generate predictor weights dynamically from the codebook itself using a hypernetwork-inspired module termed the Residual Weight Generator (ResWGEN).

Instead of learning fixed parameters, ResWGEN transforms the current prototypes into a dynamic weight matrix W_{pred}. Specifically, we first L_2-normalize the input prototypes C to obtain $\tilde{C}$. These are then mapped through a two-layer MLP augmented with a residual connection, followed by a final normalization step to ensure numerical stability. Formally, this process is defined as:

$$W_{pred} = \frac{\mathbf{H}}{\|\mathbf{H}\|_2}, \mathbf{H} = \Phi(\tilde{C}) + \mathcal{S}(\tilde{C}) \tag{7}$$

where $\tilde{C} = \frac{C}{\|C\|_2}$ denotes the normalized prototypes. $\Phi(\cdot)$ represents the non-linear MLP projection, and $\mathcal{S}(\cdot)$ denotes the residual mapping (identity or linear projection for dimension matching). We enforce strict boundary conditions for numerical validity: $H \in \mathbb{R}^{D\times D}$ with $\|H\|_2 \in (0, +\infty)$ (to avoid division by zero) and $\|W_{pred}\|_2 = 1$ (unit norm constraint for consistent semantic scaling, ensuring each element of W_{pred} lies in $[-1, 1]$). The student's features are then multiplied by these dynamic weights W_{pred} to produce prediction logits. This mechanism ensures that the student's optimization objective remains synchronized with the semantic drift of the prototype memory in real-time.

Subsequently, the prediction logits Z are computed via a matrix multiplication between the student's encoded features h_S and the transposed dynamic weights:

$$Z = h_S \cdot \left(W_{pred}\right)^\top \tag{8}$$

Mathematically, this operation calculates the cosine similarity between the student's representation and each dynamic prototype in W_{pred}. These logits are then scaled by a temperature factor and passed through a Softmax function to produce the probability distribution. The BoW Predictor utilizes this mechanism to align the student's global feature (p_{global}) with global targets, while the Local Predictor applies the same mechanism to the decoder's output features to reconstruct local semantic assignments (p_{local}).

3.4. Dual-alignment objectives and optimization

Our synergistic objective function unifies instance discrimination and structural reasoning by minimizing two complementary losses: Cross-view Global Alignment ($\mathcal{L}_{\text{global}}$) and Local Patch Alignment ($\mathcal{L}_{\text{local}}$).

Cross-view Global Alignment with Target Swapping ($\mathcal{L}_{\text{global}}$): This objective aims to learn global geometric invariance by aligning the holistic representations of two correlated views. We employ a Target Swapping strategy: the Student's global feature prediction from view 1, p^{v1}_{global}, is forced to predict the Teacher's global semantic target from view 2, y^{v2}_{T}, and vice versa. This prevents the network from relying on low-level local cues and compels it to extract shape-invariant semantics.

Mathematically, the loss is defined as the Kullback-Leibler (KL) divergence between the predicted distribution and the target distribution. To prevent the predicted distribution from collapsing into a Dirac delta (i.e., over-confidence in a single prototype) or becoming overly uniform, we incorporate an Entropy Penalty term. If the entropy of the prediction $H(p)$ falls below a threshold ϵ (set to 1.0 based on uniform distribution entropy logic), a penalty is added:

$$\mathcal{L}_{\text{global}} = \sum_{v\in\{1,2\}} D_{\text{KL}}\left(y^{\tilde{v}}_{T} \parallel p^{v}_{\text{global}}\right) + \mathbb{I}(H(p) < \epsilon)\cdot\beta(\epsilon - H(p)) \tag{9}$$

where $\tilde{v}$ denotes the opposite view, and $\beta \in (0,1]$ is the penalty coefficient. This loss function adheres to strict probability simplex constraints: $\sum_{k=1}^{K} y^{\tilde{v}}_{T,k} = 1$, $\sum_{k=1}^{K} p^{v}_{\text{global},k} = 1$ and $y^{\tilde{v}}_{T,k}$, $p^{v}_{\text{global},k} \in [0,1]$ for all $k = 1,2,\dots,K$ (where K represents the number of prototypes in the Evolving Prototype Memory). To ensure numerical stability and balanced optimization, explicit boundary conditions are enforced for each loss component: $D_{\text{KL}}\left(y^{\tilde{v}}_{T} \parallel p^{v}_{\text{global}}\right) \in [0,+\infty)$, $\mathbb{I}(H(p) < \epsilon) \in \{0,1\}$, and the entropy penalty term $\mathbb{I}(H(p) < \epsilon)\cdot\beta(\epsilon - H(p)) \in [0,1]$ (constrained to prevent overwhelming the KL divergence term). This comprehensive regularization mechanism ensures that the global features fully exploit the semantic capacity of the prototype memory space.

Local Patch Alignment ($\mathcal{L}_{\text{local}}$): While the global loss handles invariance, the local objective fosters dense contextual reasoning. For each masked patch $i \in \mathcal{M}$ (where $\mathcal{M}$ is the set of masked indices), the Student's decoder output is used to predict the Teacher's local prototype assignment $q_{T,i}$.

$$\mathcal{L}_{\text{local}} = \frac{1}{|\mathcal{M}|}\sum_{i\in\mathcal{M}} D_{\text{KL}}\left(q_{T,i} \parallel p_{\text{local},i}\right) \tag{10}$$

Unlike coordinate regression losses (e.g., Chamfer Distance) used in prior generative methods like Point-MAE, this cross-entropy-based objective forces the model to infer high-level geometric semantics (e.g., identifying that a missing region corresponds to a "table leg" prototype) based on the visible context. Similar to the global loss, entropy regularization is applied here to prevent local collapse.

Total Optimization Objective: The final training objective is a weighted sum of the global and local losses:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{global}}\mathcal{L}_{\text{global}} + \lambda_{\text{local}}\mathcal{L}_{\text{local}} \tag{11}$$

where λ_{global} and λ_{local} are hyperparameters balancing the contributions of instance discrimination and structural reconstruction (empirically set to 1.0). The Student network parameters θ_S are optimized end-to-end using the AdamW optimizer with a cosine learning rate schedule. We strictly enforce that gradients are only back-propagated through the Student network; the Teacher network parameters θ_T and the Prototype Memory C are updated solely via the momentum mechanism and queue operations, respectively. This detached update strategy creates a stable "moving target," preventing the model from learning degenerate solutions.

4. Experiments

We pre-train Point-DPA on ShapeNet to extract robust, generic representations, followed by validation on downstream tasks involving both synthetic and real-world point clouds. Additionally, comprehensive ablation studies verify the contributions of each design component.

4.1. Datasets

Our experiments utilized five widely adopted standard point cloud datasets covering various scenarios, including synthetic objects for pre-training and diverse benchmarks for downstream tasks: ShapeNet55 [29], ModelNet40 [7], ScanObjectNN [8], ShapeNetPart [9], and S3DIS [10].

Pre-training: We employ ShapeNet55 [29] for self-supervised pre-training, which comprises 57,448 synthetic models across 55 categories. We utilize the masked completion task on the full dataset to enable the encoder to learn robust latent features from complex 3D structures.

Object classification: We utilize ModelNet40 [7] for 3D object classification. Following standard protocols, the dataset is split into 9843 training and 2468 testing models, covering 40 categories. We also evaluate our method on the real-world ScanObjectNN dataset [8] to validate robustness against sensor noise and background clutter. It contains 2902 unique object instances from 15 categories. We follow the official data split strategy for evaluation.

Shape part segmentation: For fine-grained 3D understanding, we evaluate on ShapeNetPart [9]. The dataset consists of 16,881 shapes from 16 categories, annotated with 50 specific part labels. We adhere to the standard protocol, using 14,007 shapes for training and 2874 for testing.

Indoor scene segmentation: We conduct experiments on S3DIS [10] for scene-level semantic segmentation. The dataset includes 6 large indoor areas with 271 rooms, labeled into 13 semantic categories. Following standard pre-processing, rooms are divided into $1\mathrm{m} \times 1\mathrm{m}$ blocks, where each block contains 4096 points represented by a 9-dimensional feature vector (XYZ, RGB, and normalized coordinates).

4.2. Implementation details

Pre-training Settings: We utilize ShapeNet55 as the pre-training dataset. Following standard protocols, we sample 1024 points from each model and partition them into 32 patches ($G = 32$) using the FPS and KNN strategy, with each patch containing 64 points ($K = 64$). A random masking strategy with a masking ratio of 75% is employed to construct the pretext task. The Point-DPA framework is implemented in PyTorch. We employ a standard Transformer encoder (similar to ViT-Base) as the backbone. The model is pre-trained for 300 epochs with a batch size of 128. We use the AdamW optimizer with a weight decay of 0.05. The learning rate is initialized to 1×10^{-3} and decays following a Cosine Annealing schedule with a 10-epoch warm-up period. The momentum coefficient α for the Teacher network update is set to 0.996. All experiments are conducted on a single GeForce RTX 4090 graphics card.

Fine-tuning Settings: To adapt Point-DPA for downstream tasks, we discard the pre-training decoder and prediction heads, retaining only the pre-trained encoder as the backbone to which task-specific heads are appended for end-to-end fine-tuning. For classification, we implement a dual-path MLP that integrates the [CLS] token with global max-pooled geometric features, using 50% dropout for robustness, while for segmentation, we employ a feature propagation decoder that leverages multi-scale hierarchical tokens from depths [3, 7, 11] and integrates global category priors to generate predictions via a three-layer point-wise MLP. The task heads are initialized via a truncated normal distribution ($std = 0.02$), and we employ CrossEntropy loss alongside a linear DropPath strategy (0 to 0.1) to ensure optimal alignment between pre-trained geometric features and downstream semantic labels. Furthermore, to ensure a fair comparison with existing state-of-the-art methods, all task-specific hyperparameters, such as learning rate and epochs, are strictly tuned following the established protocols of Point-MAE and Point-BERT.

4.3. 3D object classification

4.3.1. Evaluation on ModelNet40

We fine-tune the pre-trained Transformer encoder on the ModelNet40 dataset for shape classification. Following standard protocols, input point clouds are uniformly sampled to 1024 points. The model is fine-tuned for 300 epochs with a batch size of 128, utilizing the AdamW optimizer with an initial learning rate of 5×10^{-4} and a weight decay of 0.05. A Cosine Annealing Learning Rate (CosLR) scheduler with a 10-epoch linear warm-up is employed to regulate training, alongside gradient clipping (max norm of 10) to ensure optimization stability. The encoder architecture remains identical to the pre-training phase to preserve feature integrity.

Table 1 presents a comparative analysis with state-of-the-art methods. Point-DPA achieves a top-1 accuracy of 93.52%. It significantly outperforms supervised baselines such as PointNet++ (90.7%) and DGCNN (92.9%), and surpasses standard tokenizer-based self-supervised methods like Point-BERT (93.2%). Furthermore, our method achieves performance on par with the strong generative baseline Point-MAE (93.8%), demonstrating that explicit coordinate reconstruction is not strictly necessary for learning high-quality object representations. We attribute this competitive performance to the proposed Masked Dynamic Prototype Alignment strategy, which compels the encoder to capture abstract high-level semantic structures (e.g., geometric primitives stored in the prototype memory).

Table 1
Object classification results (%) on ModelNet40.

Training category	Methods	Accuracy
Supervised methods	PointNet [11]	89.2
	PointNet++ [12]	90.7
	DGCNN [16]	92.9
	PointCNN [13]	92.5
	PRANet [30]	93.1
	PCT [17]	93.2
	OctFormer [31]	92.7
Self-supervised methods	OcCo [25]	93.0
	STRL [21]	93.1
	Point-BERT [6]	93.2
	Point-MAE [5]	93.8
	Point-MAE (Rep)	93.45
	Point-GAME [26]	93.35
	Ours	**93.52**

Table 2
Object classification results (%) on ScanObjectNN.

Methods	OBJ-BG	OBJ-ONLY	PB-T50-RS
PointNet [11]	73.3	79.2	68.0
PointNet++ [12]	82.3	84.3	77.9
SpiderCNN [32]	77.1	79.5	73.7
PointCNN [13]	86.1	85.5	78.5
DGCNN [16]	82.8	86.2	78.1
BGA-DGCNN [8]	–	–	79.7
PointGL [33]	–	–	86.9
Transformer [6]	79.86	80.55	77.24
Transformer + OcCo [25]	84.85	85.54	78.79
Point-BERT [6]	87.43	88.12	83.07
MaskPoint [34]	88.1	89.3	84.3
Point-MAE [5]	90.02	88.29	85.18
Point-MAE(Rep)	88.98	88.29	84.31
Point-M2AE [24]	91.22	88.81	86.43
Joint-MAE [35]	90.94	88.86	86.07
Ours	**92.03**	**89.45**	**87.03**

Table 3
Few-shot classification results (%) on ModelNet40.

Method	5-way		10-way	
	10-shot	20-shot	10-shot	20-shot
DGCNN-OcCo [25]	90.6±2.8	92.5±1.9	82.9±1.3	86.5±2.2
Transformer [6]	87.8±5.2	93.3±4.3	84.6±5.5	89.4±6.3
Transformer-OcCo [6]	94.0±3.6	95.9±2.3	89.4±5.1	92.4±4.6
Point-BERT [6]	94.6±3.1	96.3±2.7	91.0±5.4	92.7±5.1
Point-MAE [5]	96.3±2.5	97.8±1.8	92.6±4.1	95.0±3.0
Point-GAME [26]	96.1±3.1	97.9±1.4	90.3±5.1	93.8±2.5
Point-M2AE [24]	96.8±1.8	98.3±1.4	92.3±4.5	**95.0±3.0**
Ours	**96.8±2.6**	**98.4±1.3**	**92.5±4.4**	94.7±2.8

4.3.2. Evaluation on ScanObjectNN

We further assess the robustness and transferability of our representations in complex real-world environments using the ScanObjectNN benchmark. This dataset introduces significant challenges arising from background clutter, partial occlusions, and sensor noise, presenting a stark contrast to clean synthetic data. We report performance on three standard splits: *OBJ-BG* (with background), *OBJ-ONLY*, and the rigorous *PB-T50-RS* (with perturbations).

As detailed in Table 2, our model demonstrates exceptional generalization, achieving accuracies of 92.03%, 89.45%, and 87.03% on OBJ-BG, OBJ-ONLY, and PB-T50-RS, respectively. Notably, on the most challenging PB-T50-RS split, Point-DPA establishes a new state-of-the-art, outperforming the strong generative baseline Point-MAE by a substantial margin of $+1.85\%$ and surpassing the hierarchical Point-M2AE by $+0.60\%$. These results provide compelling evidence that our Dual-Alignment strategy effectively mitigates the domain gap between synthetic pre-training data (ShapeNet) and noisy real-world scans, ensuring structural robustness even under severe corruptions.

4.3.3. Evaluation on few-shot learning

We further assess the data efficiency of the learned representations using the "K-way N-shot" protocol on ModelNet40, with $K \in \{5, 10\}$ and $N \in \{10, 20\}$. Table 3 reports the mean accuracy and standard deviation over 10 independent runs. Point-DPA consistently surpasses existing methods across most settings. specifically achieving 96.8% in the 5-way 10-shot scenario and 92.5% in the 10-way 10-shot setting. These results indicate that the representations acquired via Masked Dynamic Prototype Alignment possess high transferability and robustness, enabling the Transformer encoder to rapidly adapt to novel categories with limited annotated data.

4.4. Shape part segmentation

We investigate the model's proficiency in dense prediction tasks and fine-grained local geometry capture through part segmentation experiments on the ShapeNetPart dataset. This dataset comprises 16,881 objects from 16 categories, densely annotated with 50 part labels. We report both the mean Intersection-over-Union across instances ($mIoU_I$) and categories ($mIoU_C$).

The quantitative results are summarized in Table 4. Point-DPA achieves a competitive instance mIoU of 85.9% and a category mIoU of 84.39%. Notably, our method outperforms the tokenizer-based Point-BERT (85.6%) and achieves performance parity with the reconstruction-specialist Point-MAE (86.1%). This result is particularly significant as it confirms that our Local Patch Alignment objective successfully mitigates the "global bias" typically associated with contrastive learning. By forcing the encoder to reconstruct local semantic prototypes, Point-DPA preserves the intricate spatial topology necessary for dense point-wise classification. Qualitative

Table 4
Part segmentation results (%) on ShapeNetPart dataset.

Methods	mIoU$_C$	mIoU$_I$	aero	bag	cap	car	chair	earph.	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	skateb.	table
PointNet [11]	80.39	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
PointNet++ [12]	81.85	85.1	82.4	79.0	86.7	77.3	90.8	71.8	91.0	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6
DGCNN [16]	82.33	85.2	84.0	83.4	86.7	77.8	90.6	74.7	91.2	87.5	82.8	95.7	66.3	94.9	81.1	63.5	74.5	82.6
Transformer [6]	83.42	85.1	82.9	85.4	87.7	78.8	90.5	80.8	91.1	87.7	85.3	95.6	73.9	94.9	83.5	61.2	74.9	80.6
Point-BERT [6]	84.11	85.6	84.3	84.8	88.0	79.8	91.0	**81.7**	91.6	87.9	85.2	95.6	75.6	**94.7**	84.3	63.4	76.3	81.5
Point-MAE [5]	84.19	**86.1**	84.3	**85.0**	88.3	80.5	91.3	78.5	92.1	87.4	86.1	**96.1**	75.2	94.6	84.7	63.5	77.1	**82.4**
Ours	**84.39**	85.9	**84.9**	83.9	**88.8**	**81.2**	**91.4**	78.8	**92.4**	**88.0**	**86.1**	95.9	**77.0**	94.6	**84.8**	**63.9**	**77.5**	80.9

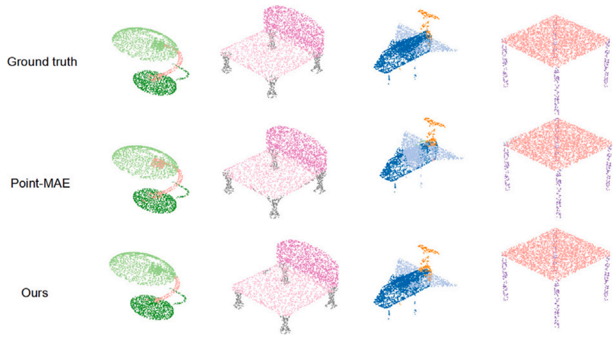

Fig. 4. Qualitative visualization of Part segmentation results on ShapeNetPart. We compare the segmentation predictions of Point-DPA against the Ground Truth and Point-MAE. Our method generates more coherent predictions with fewer artifacts, particularly in thin structural regions, demonstrating superior preservation of fine-grained geometric topology.

Table 5
Semantic segmentation results (%) on S3DIS Area 5.

Methods	Input	OA	mAcc	mIoU	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
PointNet [11]	xyz+rgb	–	49.0	41.1	88.8	97.3	69.8	0.1	3.9	46.3	10.8	58.9	52.6	5.9	40.3	26.4	33.2
PointNet++ [12]	xyz+rgb	83.0	62.0	53.5	89.4	97.7	75.4	0.0	1.8	58.3	19.5	69.2	79.0	46.2	59.1	58.7	41.6
DGCNN [16]	xyz+rgb	84.1	–	56.1	–	–	–	–	–	–	–	–	–	–	–	–	–
PointCNN [13]	xyz+rgb	85.9	63.9	57.3	92.3	98.2	79.4	0.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	56.7
Transformer [6]	xyz	87.3	69.3	60.2	94.6	98.5	79.9	0.5	30.7	57.0	73.2	73.6	81.6	29.7	65.5	46.4	51.5
Point-MAE [5]	xyz	**87.8**	68.6	60.8	**94.1**	98.3	81.0	0.0	23.7	**60.7**	72.4	75.0	**85.4**	27.8	**67.2**	**51.1**	53.6
Ours	xyz	87.6	**69.3**	**61.4**	92.7	**98.3**	**81.7**	0.0	**27.4**	60.4	**73.2**	**76.4**	82.7	**38.0**	66.3	48.4	**53.7**

comparisons are visualized in Fig. 4. As observed, the segmentation maps predicted by Point-DPA are visually closer to the Ground Truth compared to baselines. Specifically, while Point-MAE tends to generate noisy artifacts or disconnected components in thin, fragile structures (e.g., the neck of the lamp), our method produces significantly more spatially coherent and smooth segmentation boundaries. This evidence suggests that predicting high-level semantic prototypes yields more robust boundary supervision than reconstructing raw low-level coordinates.

4.5. Semantic segmentation

Unlike part segmentation with synthetic point clouds, semantic segmentation on real-world scenes is substantially more challenging as real-world point clouds contain abundant outliers and noisy points. We utilize the S3DIS dataset to test our Point-DPA on indoor scenes. We strictly follow previous works to prepare the training data and test the performance of our model on Area 5 of the S3DIS dataset. Different from most previous fully supervised methods that use both xyz coordinates and rgb colors as inputs, we only take xyz coordinates as inputs to align with our pre-training regime, which relies solely on spatial geometry. Table 5 shows the quantitative results in terms of the overall pointwise accuracy (OA), the mean classwise accuracy (mAcc), and the mean classwise IoU (mIoU). The proposed Point-DPA achieves 87.6% OA, 69.3% mAcc, and 61.4% mIoU, demonstrating strong generalization capability in complex indoor environments. Specifically, compared with the previous best masked autoencoding-based counterpart Point-MAE, our Point-DPA achieves 0.6% and 0.7% performance improvement on mIoU and mAcc, respectively, while maintaining a comparable Overall Accuracy (87.6% vs. 87.8%). This improvement in class-wise metrics (mIoU and mAcc) evidences that our prototype-based pre-training effectively captures discriminative semantic features for tail classes, rather than being dominated by background points. To visually prove the superiority of our method, we visualize the segmentation results of our Point-DPA in Fig. 5. It can be seen that our method can produce a higher quality segmentation prediction than other competitors, particularly in distinguishing adjacent objects with similar geometric structures.

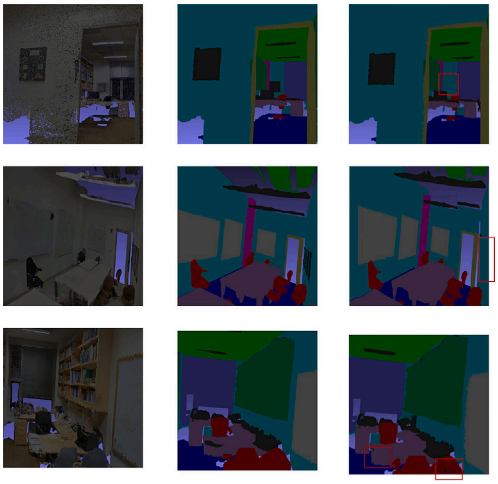

Fig. 5. Visualization of semantic segmentation results on S3DIS Area 5 the columns display, from left to right: the raw data, the Ground Truth (GT) annotations, and the output predicted semantic segmentation results from the same viewpoint. The red boxes mark areas where our predictions have minor flaws for a fair presentation.

Table 6
Ablation study of dual-alignment objectives on ModelNet40.

Model	Loss functions		Results	
	$\mathcal{L}_{\text{local}}$	$\mathcal{L}_{\text{global}}$	Accuracy (%)	Δ
A	✓		91.85	−1.67
B		✓	92.43	−1.09
C (Ours)	✓	✓	**93.52**	–

Table 7
Ablation study of dual-alignment objectives on ScanObjectNN.

Model	Loss functions		Results	
	$\mathcal{L}_{\text{local}}$	$\mathcal{L}_{\text{global}}$	Accuracy (%)	Δ
A	✓		85.15	−1.88
B		✓	85.62	−1.41
C (Ours)	✓	✓	87.03	–

4.6. Ablation study

(1) Effectiveness of Dual-Alignment Objectives: We first investigate the individual contributions of the Global Alignment Loss ($\mathcal{L}_{\text{global}}$) and the Local Patch Alignment Loss ($\mathcal{L}_{\text{local}}$). The quantitative comparisons are summarized in Tables 6 and 7. On the clean synthetic ModelNet40 dataset, the full model achieves an accuracy of 93.52%, consistently outperforming single-objective variants. Specifically, removing global alignment leads to a performance penalty of 1.67%, while omitting local alignment results in a 1.09% degradation. Crucially, this performance disparity is significantly more pronounced on the real-world ScanObjectNN dataset. As shown in Table 7, the model trained solely with $\mathcal{L}_{\text{local}}$ suffers a substantial drop of 1.88%, while the variant using only $\mathcal{L}_{\text{global}}$ declines by 1.41% compared to the full model (87.03%). This evidence underscores a critical insight: in noisy real-world scenarios, neither local reconstruction nor global invariance is sufficient in isolation. The synergy of both objectives is essential; $\mathcal{L}_{\text{local}}$ compels the encoder to disentangle fine-grained object geometry from background clutter, while $\mathcal{L}_{\text{global}}$ stabilizes the semantic representation against partial occlusions and perturbations.

(2) Reconstruction Targets(Raw Coordinates vs. Prototype Codes): We substantiate our central hypothesis that predicting semantic prototypes yields superior representations compared to direct coordinate regression, particularly within the domain of noisy real-world data.

Quantitative results in Table 8 reveal that the coordinate regression baseline (analogous to Point-MAE) attains an accuracy of 85.84% on the ScanObjectNN dataset. In contrast, our prototype-based approach achieves 87.03%, marking a significant improvement of +1.19%.

We attribute this performance gain to the inherent nature of the reconstruction targets. Real-world point clouds are frequently plagued by high-frequency sensor noise. Forcing the network to regress exact geometric coordinates compels it to overfit this stochastic noise. Conversely, our Masked Dynamic Prototype Alignment functions as a *semantic denoising* mechanism. By abstracting noisy local patches into clean, discrete semantic codes (e.g., mapping a perturbed "table leg" fragment to a

Table 8
Comparison of different reconstruction targets on ScanObjectNN.

Target type	Objective	Accuracy (%)
Raw Coordinates	Chamfer Distance	85.84
Prototype Codes (Ours)	Cross Entropy (KL)	**87.03**

Table 9
Comparison of different feature aggregation methods on ScanObjectNN.

Aggregation method	Definition	Accuracy (%)
Mean-Pooling	Mean (H_T)	86.12
Max-Pooling	Max (H_T)	86.55
Dual-Pooling (Ours)	Concat (Max, Mean)	**87.03**

canonical “cylinder” prototype), the model effectively filters out local geometric jitter and learns more robust, shape-invariant features.

(3) Impact of Feature Aggregation Strategies: As described in the Method section, our Teacher branch employs a Dual-Pooling strategy (concatenating Max-Pooling and Mean-Pooling features) to extract global semantic targets. To validate the necessity of this design on complex real-world data, we compare it against using standard Max-Pooling or Mean-Pooling alone on the ScanObjectNN dataset in Table 9. The results indicate that using only Mean-Pooling yields the lowest accuracy (86.12%), as it tends to smooth out distinct geometric features which are crucial in distinguishing objects from clutter. Max-Pooling performs slightly better (86.55%) by effectively capturing prominent features like corners and edges. However, the proposed Dual-Pooling strategy achieves the highest accuracy of 87.03%. This improvement demonstrates that Max-Pooling and Mean-Pooling are complementary even in noisy environments: Max-Pooling preserves sharp local structures, while Mean-Pooling captures the holistic statistical distribution of the shape, and combining them provides a more comprehensive semantic guidance for the Student network.

5. Conclusion

This work presents Point-DPA, a unified self-supervised framework bridging the dichotomy between contrastive invariance and generative reasoning. By eschewing the reliance on low-level coordinate reconstruction and complex offline tokenizers, we introduced an Evolving Prototype Memory within an asymmetric Teacher-Student architecture, enabling the model to capture high-level semantic concepts dynamically. Extensive evaluations on synthetic and real-world benchmarks, including ModelNet40 (93.52%) and ScanObjectNN (87.03%), demonstrate that our approach effectively learns representations that are both globally discriminative and locally sensitive. Despite the encouraging performance of Point-DPA across various benchmarks, the framework still exhibits a few limitations that call for future improvement. Specifically, the inherently irregular and sparse nature of 3D point clouds continues to pose challenges for semantic abstraction in extremely sparse regions. Furthermore, the reliance on a fixed random masking strategy may not optimally capture the varied geometric complexities present in diverse real-world scenes. Addressing these limitations remains a priority for our future research. We intend to explore more adaptive, density-aware masking strategies and extend our dynamic prototype mechanism to large-scale outdoor LiDAR environments to verify its scalability. Additionally, integrating multi-modal signals to further enrich the semantic embedding space will be a key direction to enhance the model’s robustness and generalization in complex scenarios.

CRediT authorship contribution statement

Xin Cao: Methodology, Investigation, Formal analysis, Conceptualization. **Xinmeng Hu:** Writing – review & editing, Writing – original draft, Visualization. **Yinan Wang:** Visualization, Software, Data curation. **Kang Li:** Supervision, Project administration, Funding acquisition. **Linzhi Su:** Formal analysis, Visualization, Writing – review & editing. **Yangyang Liu:** Supervision, Formal analysis, Data curation. **Fengjun Zhao:** Validation, Supervision, Resources.

Funding

This work was supported in part by the Key Research and Development Program of Shaanxi Province (2024SF-YBXM-681), and in part by the National Natural Science Foundation of China (62572394, 61806164, 62476218).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, 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), 2019, pp. 4171–4186.
[2] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, R. Girshick, Masked autoencoders are scalable vision learners, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 15979–15988.
[3] S. Xie, J. Gu, D. Guo, C.R. Qi, L. Guibas, O. Litany, PointContrast: unsupervised pre-training for 3D point cloud understanding, in: European Conference on Computer Vision, Springer, 2020, pp. 574–591.
[4] M. Afham, I. Dissanayake, D. Dissanayake, A. Dharmasiri, K. Thilakarathna, R. Rodrigo, CrossPoint: self-supervised cross-modal contrastive learning for 3D point cloud understanding, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9902–9912.
[5] Y. Pang, E.H.F. Tay, L. Yuan, Z. Chen, Masked autoencoders for 3D point cloud self-supervised learning, World Sci. Annu. Rev. Artif. Intell. 1 (2023) 2440001.
[6] X. Yu, L. Tang, Y. Rao, T. Huang, J. Zhou, J. Lu, Point-BERT: pre-training 3D point cloud transformers with masked point modeling, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 19291–19300.
[7] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, J. Xiao, 3D ShapeNets: a deep representation for volumetric shapes, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1912–1920.
[8] M.A. Uy, Q.-H. Pham, B.-S. Hua, T. Nguyen, S.-K. Yeung, Revisiting point cloud classification: a new benchmark dataset and classification model on real-world data, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 1588–1597.
[9] L. Yi, V.G. Kim, D. Ceylan, I.-C. Shen, M. Yan, H. Su, C. Lu, Q. Huang, A. Sheffer, L. Guibas, A scalable active framework for region annotation in 3D shape collections, ACM Trans. Graph. 35 (6) (2016) 1–12.
[10] I. Armeni, O. Sener, A.R. Zamir, H. Jiang, I. Brilakis, M. Fischer, S. Savarese, 3D semantic parsing of large-scale indoor spaces, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1534–1543.
[11] C.R. Qi, H. Su, K. Mo, L.J. Guibas, PointNet: deep learning on point sets for 3D classification and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.
[12] C.R. Qi, L. Yi, H. Su, L.J. Guibas, PointNet++: deep hierarchical feature learning on point sets in a metric space, Adv. Neural Inf. Process. Syst. 30 (2017).
[13] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, B. Chen, PointCNN: convolution on X-transformed points, Adv. Neural Inf. Process. Syst. 31 (2018).
[14] X. Ma, C. Qin, H. You, H. Ran, Y. Fu, Rethinking network design and local geometry in point cloud: a simple residual mlp framework, arXiv preprint arXiv:2202.07123, 2022.
[15] A.A.M. Muzahid, W. Wan, F. Sohel, L. Wu, L. Hou, CurveNet: curvature-based multitask learning deep networks for 3D object recognition, IEEE/CAA J. Autom. Sin. 8 (6) (2020) 1177–1187.
[16] Y. Wang, Y. Sun, Z. Liu, S.E. Sarma, M.M. Bronstein, J.M. Solomon, Dynamic graph CNN for learning on point clouds, ACM Trans. Graph. 38 (5) (2019) 1–12.
[17] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R.R. Martin, S.-M. Hu, PCT: point cloud transformer, Comput. Vis. Media 7 (2) (2021) 187–199.
[18] S. Guo, J. Cai, Y. Hu, Q. Liu, M. Xu, LCASAFormer: cross-attention enhanced backbone network for 3D point cloud tasks, Pattern Recognit. 162 (2025) 111361.
[19] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: International Conference on Machine Learning, PmLR, 2020, pp. 1597–1607.
[20] J. Zbontar, L. Jing, I. Misra, Y. LeCun, S. Deny, Barlow twins: self-supervised learning via redundancy reduction, in: International Conference on Machine Learning, PMLR, 2021, pp. 12310–12320.
[21] S. Huang, Y. Xie, S.-C. Zhu, Y. Zhu, Spatio-temporal self-supervised representation learning for 3D point clouds, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 6535–6545.
[22] H. Bao, L. Dong, S. Piao, F. Wei, BEiT: BERT pre-training of image transformers, arXiv preprint arXiv:2106.08254, 2021.
[23] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, H. Hu, SimMIM: a simple framework for masked image modeling, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9653–9663.
[24] R. Zhang, Z. Guo, P. Gao, R. Fang, B. Zhao, D. Wang, Y. Qiao, H. Li, Point-M2AE: multi-scale masked autoencoders for hierarchical point cloud pre-training, Adv. Neural Inf. Process. Syst. 35 (2022) 27061–27074.
[25] H. Wang, Q. Liu, X. Yue, J. Lasenby, M.J. Kusner, Unsupervised point cloud pre-training via occlusion completion, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9782–9792.
[26] Y. Liu, X. Yan, Z. Li, Z. Chen, Z. Wei, M. Wei, PointGame: geometrically and adaptively masked autoencoder on point clouds, IEEE Trans. Geosci. Remote Sens. 61 (2023) 1–12.
[27] R. Dong, Z. Qi, L. Zhang, J. Zhang, J. Sun, Z. Ge, L. Yi, K. Ma, Autoencoders as cross-modal teachers: can pretrained 2D image transformers help 3D representation learning? arXiv preprint arXiv:2212.08320, 2022.
[28] S. Gidaris, A. Bursuc, O. Simeoni, A. Vobecky, N. Komodakis, M. Cord, P. Pérez, MOCA: self-supervised representation learning by predicting masked online codebook assignments, arXiv preprint arXiv:2307.09361, 2023.
[29] A.X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al., ShapeNet: an information-rich 3D model repository, arXiv preprint arXiv:1512.03012, 2015.
[30] S. Cheng, X. Chen, X. He, Z. Liu, X. Bai, PRA-Net: point relation-aware network for 3D point cloud analysis, IEEE Trans. Image Process. 30 (2021) 4436–4448.
[31] P.-S. Wang, OctFormer: octree-based transformers for 3D point clouds, ACM Trans. Graph. 42 (4) (2023) 1–11.
[32] Y. Xu, T. Fan, M. Xu, L. Zeng, Y. Qiao, SpiderCNN: deep learning on point sets with parameterized convolutional filters, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 87–102.
[33] J. Li, J. Wang, T. Xu, PointGL: a simple global-local framework for efficient point cloud analysis, IEEE Trans. Multimedia 26 (2024) 6931–6942.
[34] H. Liu, M. Cai, Y.J. Lee, Masked discrimination for self-supervised learning on point clouds, in: European Conference on Computer Vision, Springer, 2022, pp. 657–675.
[35] Z. Guo, R. Zhang, L. Qiu, X. Li, P.-A. Heng, Joint-MAE: 2D–3D joint masked autoencoders for 3D point cloud pre-training, arXiv preprint arXiv:2302.14007, 2023.