Point-LGMask: Local and Global Contexts Embedding for Point Cloud Pre-Training With Multi-Ratio Masking

Yuan Tang, Xianzhi Li, Jinfeng Xu, Qiao Yu, Long Hu, Yixue Hao, *Member, IEEE*,
and Min Chen, *Fellow, IEEE*

Abstract—**Self-supervised learning has achieved great success in both natural language processing and 2D vision, where masked modeling is a quite popular pre-training scheme. However, extending masking to 3D point cloud understanding that combines local and global features poses a new challenge. In our work, we present Point-LGMask, a novel method to embed both local and global contexts with multi-ratio masking, which is quite effective for self-supervised feature learning of point clouds but is unfortunately ignored by existing pre-training works. Specifically, to avoid fitting to a fixed masking ratio, we first propose multi-ratio masking, which prompts the encoder to fully explore representative features thanks to tasks of different difficulties. Next, to encourage the embedding of both local and global features, we formulate a compound loss, which consists of (i) a global representation contrastive loss to encourage the cluster assignments of the masked point clouds to be consistent to that of the completed input, and (ii) a local point cloud prediction loss to encourage accurate prediction of masked points. Equipped with our Point-LGMask, we show that our learned representations transfer well to various downstream tasks, including few-shot classification, shape classification, object part segmentation, as well as real-world scene-based 3D object detection and 3D semantic segmentation. Particularly, our model largely advances existing pre-training methods on the difficult few-shot classification task using the real-captured ScanObjectNN dataset by surpassing over 4% to the second-best method. Also, our Point-LGMask achieves 0.4% AP_{25} and 0.8% AP_{50} gains on 3D object detection task over the second-best method. For semantic segmentation, our Point-LGMask surpasses the second-best method by 0.4% mAcc and 0.5% mIoU.**

Index Terms—**Local and global contexts embedding, self-supervised learning, point cloud understanding, representation learning.**

I. Introduction

Self-supervised learning (SSL) enables AI systems to learn powerful representations from orders of magnitude data by leveraging the supervisory signals from the data itself, which has made great success in advancing the field of natural language processing (NLP) [1], 2D vision [2], [3], multimodal analysis [4], etc. Generally, models pre-trained this way yield considerably higher performance than when solely trained in a supervised manner.

Among existing SSL techniques, the masked modeling scheme shows a particularly profound impact on both NLP and 2D vision, which aims to predict any unobserved or hidden part (or property) of the input from any observed or unhidden part of the input. In our work, we consider *whether 3D point cloud can also continue the success of employing the masking scheme for point cloud pre-training*.

So far, only a few works try to apply masking to point cloud pre-training. Point-BERT [5] constructs a pretext task by predicting the tokens of the masked point clouds. Yet, the process requires an offline tokenizer. Later, Point-McBert [6] proposes multi-choice tokens to overcome the ambiguity in the self-supervised signal generated by Point-BERT's tokenizer. However, both Point-BERT and Point-McBert are limited by the ability of the tokenizer. In contrast, MaskPoint [7] and Point-MAE [8] do not require an offline tokenizer. MaskPoint is designed to distinguish whether local points are noise or masked points, while Point-MAE reconstructs the masked points. However, they both ignored the global representation, which is beneficial for downstream high-level tasks. Previous works like PointNet++ [9] show that considering local and global features together is crucial for point cloud understanding. However, existing attempts that use the masked modeling scheme in point cloud self-supervised learning, unfortunately, ignore the embedding of local and global features. In our work, we present Point-LGMask, a novel method to skillfully combine the embedding of local and global contexts and self-supervised learning with multi-ratio masking for point cloud pre-training. Unlike existing works [5], [6], [7], [8], we first propose to mask points using multiple ratios rather than a single ratio. Intuitively, compared to single-ratio masking, our multi-ratio masking prompts the encoder to fully explore representative features thanks to pretext tasks of different difficulties, thus producing more universal

Manuscript received 13 January 2023; revised 17 April 2023; accepted 21 May 2023. Date of publication 8 June 2023; date of current version 31 July 2024. This work was supported by the National Natural Science Foundation of China under Grants 62202182, 62276109, 62176101, and 61821003. The Guest Editor coordinating the review of this manuscript and approving it for publication was Dr. Jiajun Deng. *(Corresponding author: Xianzhi Li.)*

Yuan Tang, Xianzhi Li, Jinfeng Xu, Qiao Yu, Long Hu, and Yixue Hao are with the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: yuan_tang@hust.edu.cn; xzli@hust.edu.cn; jinfengxu.edu@gmail.com; qiaoyu_epic@hust.edu.cn; hulong@hust.edu.cn; yixuehao@hust.edu.cn).

Min Chen is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510640, China, and also with the Pazhou Laboratory, Guangzhou 510640, China (e-mail: minchen@ieee.org).

Codes have been released at https://github.com/TangYuan96/Point-LGMask.
Digital Object Identifier 10.1109/TMM.2023.3282568

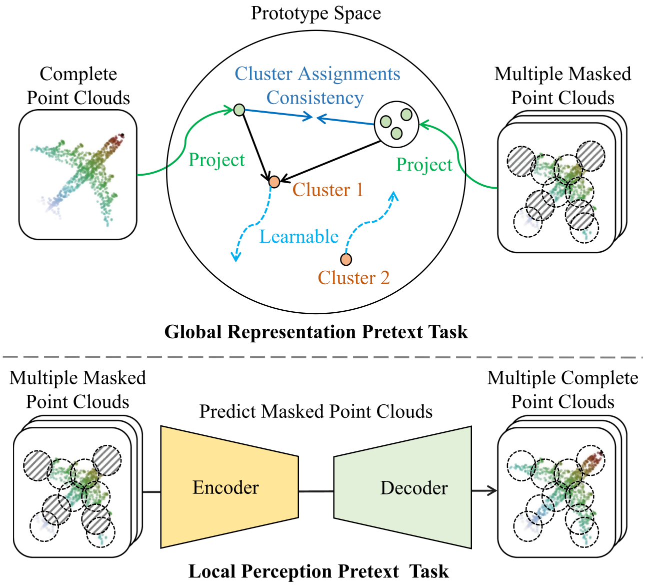
Prototype Space
Complete Point Clouds
Cluster Assignments Consistency
Multiple Masked Point Clouds
Project
Project
Cluster 1
Learnable
Cluster 2
Global Representation Pretext Task
Multiple Masked Point Clouds
Predict Masked Point Clouds
Multiple Complete Point Clouds
Encoder
Decoder
Local Perception Pretext Task

Fig. 1. Illustrating our proposed local and global contexts embedding for self-supervised learning. Given a 3D point cloud, our Point-LGMask first uses multi-ratio masking to generate multiple masked point clouds, then constructs a global representation pretext task (top) to encourage the cluster assignments of the masked point clouds to be consistent to that of the completed input, and a local perception pretext task (bottom) to predict the masked point clouds.

representations that work across a wider variety of downstream tasks. Also, the trained network model after multi-ratio masking scheme can be regarded as averaging the predictions of all possible settings of the parameters, thus avoiding the overfitting to a fixed ratio and increasing the generalization ability. Although the stochastic masking used in Point-BERT [5] and Point-McBert [6] tries to strengthen model generalization, it sacrifices training stability. In contrast, our method can balance training stability and model generalization by fixed multi-ratio masking. Further, to extract both local and global features, we construct two pretext tasks (see Fig. 1): one global representation pretext task to encourage cluster assignments of the masked point clouds to be consistent to that of the complete input, and one local perception pretext task to predict the masked point clouds. During end-to-end pre-training, the former pretext task drives model to capture global features, while the latter one drives to focus on local features.

Equipped with our Point-LGMask, we show that our learned representations transfer well to various downstream applications, including few-shot classification, shape classification, object part segmentation, 3D object detection and semantic segmentation on real-world large-scale scenes. Particularly, our Point-LGMask gets over 4% performance improvement compared to the second-best method on ScanObjectNN [10] dataset for the difficult few-shot classification. Our Point-LGMask also achieves 0.4% AP_{25} and 0.8% AP_{50} gains on 3D object detection task over the second-best method. For semantic segmentation, our Point-LGMask surpasses the second-best method by 0.4% mAcc and 0.5% mIoU. In general, we summarize our contributions as follows:

1) We present Point-LGMask, a novel method to effectively and skillfully integrate both local and global features into self-supervised learning by designing a compound loss to encourage our network model to embed both local and global contexts for point cloud pre-training.
2) We propose a multi-ratio masking scheme to enrich the difficulty of pretext tasks for representative feature learning and avoid overfitting.
3) We conduct extensive experiments to show the superior performance of our Point-LGMask for transferring the learned knowledge to various downstream tasks.

II. Related Work

Self-supervised Learning on Point Clouds: The core of SSL is to design a mechanism (or pretext tasks) to generate supervision signals from the input data itself. In the early stage, the generative model is designed to reconstruct or complete the given input point cloud via auto-encoders. For instance, Han et al. [11] presents MAP-VAE for unsupervised feature learning by jointly leveraging local and global self-supervision, where local self-supervision is enabled by multi-angle analysis and global geometry is learned by self-reconstruction. Cai et al. [12] proposes to learn a unified and structured latent space that encodes both partial and complete point clouds through self-reconstruction and completion. In recent years, pretext tasks that exploit the rich attributes of point clouds have been introduced, which drive the model to learn deeper semantic knowledge, such as contrastive learning [13], [14], [15], orientation estimation [16], category-level 6D object pose estimation [17], BERT-style pre-training for point cloud [5], [6], [7], [8], etc. Our work is greatly inspired by Point-BERT [5], which generalizes the concept of BERT [1] to 3D point clouds. Point-BERT constructs a pretext task by predicting the tokens of the masked point clouds. Yet, the process requires an offline tokenizer. Later, Point-McBert [6] proposes multi-choice tokens to overcome the ambiguity in the self-supervised signal generated by Point-BERT's tokenizer. However, Point-BERT and Point-McBert both require a tokenizer to be trained offline, resulting in the learned representations of point clouds being limited by the ability of the tokenizer. MaskPoint [7] and Point-MAE [8] do not require an offline tokenizer. MaskPoint is designed to distinguish whether local points are noise or masked points, while Point-MAE is designed to reconstruct the masked points. Furthermore, the self-supervised signal utilized by ConClu [15] is at a global level derived from contrastive and clustering losses, while disregarding local perception. In contrast, the self-supervisory signal of our Point-LGMask is derived from both global representation and local perception, which enables the model to learn richer knowledge.

Masked Image Modeling: Inspired by the masked language modeling in natural language processing, masked image modeling (MIM) motivates a flux of research [2], [3], [18], [19], [20] in terms of 2D image self-supervised pre-training. Following the "mask-and-reconstruct" pipeline, at the image token level, PeCo [18] and BEIT [20] utilize an offline trained tokenizer to

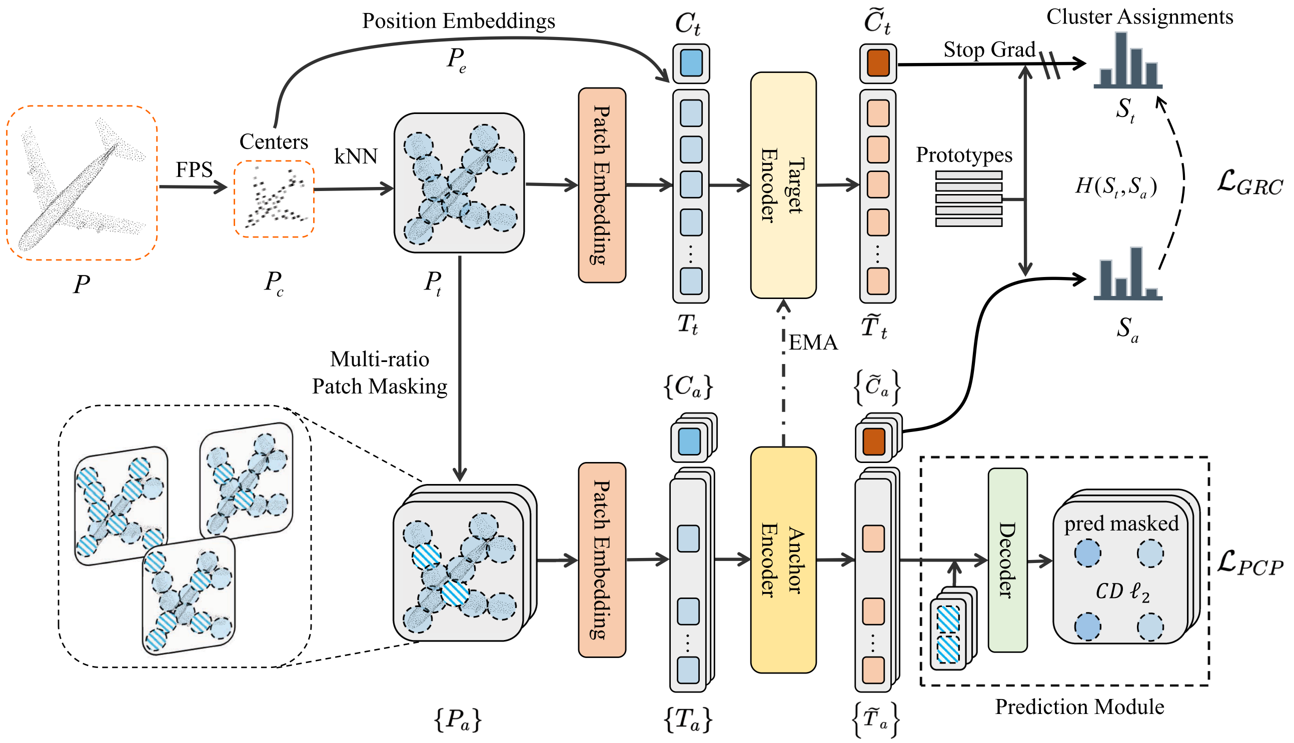

Fig. 2. Overview of our Point-LGMask. We first divide input point clouds P into N target point patches P_t. Next, we propose to employ multiple ratios to mask P_t to get multiple anchor patches $\{P_a\}$. Then, we embed both P_t and $\{P_a\}$, which are fed into encoders to obtain high-level representations. We formulate a compound objective function with two terms (i.e., $\mathcal{L}_{GRC}$ & $\mathcal{L}_{PCP}$) for both global and local contexts embedding.

predict masked tokens, enabling the model to learn local representations higher than pixels. In contrast, IBOT [19] proposes to rely on the teacher-student architecture for self-supervision without tokenizer. At the pixel level, MAE [2] and SimMIM [3] mask parts of image pixels, prompting the model to predict masked regions to encourage the model to focus on image local semantics. Inspired by MAE [2], one of our objectives is to predict the masked parts of point clouds in the pre-training stage.

Contrastive Learning: As a key technique in SSL, contrastive learning constructs a pretext task to learn generalizable representations using contrastive pairs without relying on labeled data. The development of contrastive learning so far can be roughly divided into the following four stages. In the first stage, early contrastive learning methods [21], [22] are still in exploration, where the model structure, objective function, and pretext task style are not unified. In the second stage, the objective functions are roughly unified into infoNCE and its variants. In addition, diverse works [23], [24] have realized the importance of data augmentation and projection head. In the third stage, some methods [25], [26] begin to discard negative samples and focus on positive sample augmentation. In the fourth stage, with the huge impact of Transformers on visual, several works [27], [28] organically combine Transformers with contrastive learning. At the same time, the emerging MIM also provides an augmentation idea for contrastive learning. MSN [29] proposes to match the representation of an image view containing randomly masked patches to the representation of the original unmasked image. In this work, we seek to continue the success of contrastive learning and extend it to point cloud feature learning.

III. Method

A. Overview

The goal of this work is to develop a pre-training method for feature learning of 3D point clouds in a self-supervised manner. Inspired by masked point modeling [5], we design our Point-LGMask as shown in Fig. 2. Generally, given an input point cloud P, we first divide it into N point patches (denoted as target point patches P_t) by using the farthest point sampling (FPS) and k-nearest neighbor algorithm (kNN). To capture good representations, we propose to apply multiple different masking ratios to the N patches, and the resulting incomplete point patches after masking are regarded as multiple anchor point patches $\{P_a\}$. Next, both the target point patch P_t and the multiple anchor point patches $\{P_a\}$ are embedded to target tokens T_t and multiple anchor tokens $\{T_a\}$ via a patch embedding module, which is implemented using mini-PointNet [30]. Subsequently, we design a target encoder to lift the target tokens T_t and the target class token C_t that is a learnable parameter to higher embeddings $\widetilde{T}_t$ and $\widetilde{C}_t$, respectively. Similarly, we also design an anchor encoder to lift multiple anchor tokens $\{T_a\}$ and multiple anchor class tokens $\{C_a\}$ that are learnable parameters to higher embeddings $\{\widetilde{T}_a\}$ and $\{\widetilde{C}_a\}$, respectively. Note that, both the target encoder and the anchor encoder are built upon standard Transformer, which shall be detailed later. Finally, as shown in the right part of Fig. 2, to learn both global and local contexts, we design a global representation contrastive loss $\mathcal{L}_{GRC}$ to enforce consistency between cluster assignments produced by target and anchor representations, and design a point cloud prediction loss $\mathcal{L}_{PCP}$ to measure the accuracy of predictions. For end-to-end training, the anchor encoder is trained by using the two loss terms, and the parameters of the target encoder are updated using exponential moving average (EMA) combined with the parameters of the anchor encoder.

Below, we will present the details of our key components in Point-LGMask, including the process of patch generation, masking and tokenization, the (target and anchor) encoder design based on Transformer, as well as two objectives to learn local and global contexts.

B. Patch Generation, Masking and Tokenization

In this subsection, we describe the detailed process of patch generation, multi-ratio masking, and tokenization.

Patch generation: To adopt the masked point modeling scheme, we follow Point-BERT [5] to treat a local region around a reference point as one token. Specifically, as shown in the left-side of Fig. 2, given an input point cloud P, we first employ FPS to pick N points as patch centers $P_c \in \mathbb{R}^{N\times 3}$. Then we use kNN to group a local patch around each center point, where k is set to be 32 empirically. In this way, we totally obtain N target point patches $P_t \in \mathbb{R}^{N\times k\times 3}$. Note that, for fast convergence, we further normalize P_t by taking P_c as the coordinate origin.

Multi-ratio patch masking: Once we obtain the N target patches P_t, the next step is to mask a certain percentage of them. Not surprisingly, when a larger proportion of patches are dropped, the context size is reduced, thus creating a harder task. On the contrary, when we discard only a small number of patches, the network makes predictions based on more input contexts. Though reducing the training difficulty, it may not generalize well during inference. Hence, exploring a suitable masking ratio is quite important, but not easy. Existing masking strategies can be broadly classified into stochastic masking (used in Point-BERT [5] and Point-McBert [6]) and fixed ratio masking (used in Point-MAE [8] and MaskPoint [7]). The former uses a varying but random ratio during training, which may lead to unstable training due to the random changes in pretext task difficulty. The latter with a single fixed ratio circumvents the training instability but sacrifices the model generalization.

In our work, instead of using a single-ratio masking like [5], [6], [7], [8], we propose to use multi-ratio masking to balance training stability and model generalization, though simple yet quite effective. During each training iteration, we assign multiple different but fixed masking ratios to generate a set of masked patches, and then design our network to learn tokens under each masking ratio. Intuitively, compared to the single-ratio masking, our multi-ratio masking scheme enables multiple pretext tasks with different difficulties, thus promoting the encoder to fully exploit and extract representative features thanks to pretext tasks of different difficulties, thus facilitating the downstream tasks. Also, similar to the idea of Dropout layer, the trained network model after multi-ratio masking scheme can be regarded as averaging the predictions of all possible settings of the parameters, thus avoiding the overfitting to a fixed ratio and increasing the generalization ability. Specifically, we set totally three different masking ratios $\{r_m\} = \{0.3, 0.6, 0.9\}$, and then randomly select and drop patches from P_t according to each ratio to generate the associated anchor patch P_a.

Patch tokenization: The purpose of this step is to embed each point patch into a token. Here, we follow Point-BERT [5] to use the mini-PointNet [30] to project each sub-cloud (i.e., point patch) into point embedding (i.e., patch token). More specifically, given the unmasked target point patches P_t, we obtain the corresponding target tokens T_t using mini-PointNet. While for the multiple anchor point patches $\{P_a\}$ after masking, we obtain the corresponding multiple anchor tokens $\{T_a\}$.

C. Transformer Encoder

This work designs two encoders, i.e., the target encoder and the anchor encoder, obtaining the self-supervised signal of contrastive learning. In detail, during the training phase, the parameters of the anchor encoder θ_a are trained with back-propagation updates, which are exponentially moving averaged (EMA) to the parameters of the target encoder θ_t. Formally,

$$\theta_t = \alpha\theta_t + (1-\alpha)\theta_a, \quad \text{where} \quad \alpha = 0.999. \tag{1}$$

As shown in Fig. 2, both the target encoder and the anchor encoder consume patch tokens as well as class tokens as inputs. More specifically, let's take the anchor encoder as an example for explanation. The inputs consist of multiple class tokens $\{C_a\}$ which are learnable parameters, and multiple anchor patch tokens $\{T_a\}$. For the processing of masked patches, instead of using learnable parameters to replace the masked tokens as input to the encoder in Point-BERT, we directly discard the masked tokens, and only feed the tokens of visible patches and their corresponding position embeddings as inputs. The operation of discarding masked tokens reduces the data volume of inputs and significantly improves the training speed.

The backbones of both encoders use the same architecture based on the standard Transformer [31], consisting of multiple blocks that are composed of multi-head self-attention layer and feed-forward network.

D. End-to-End Network Training

To make the pre-trained network adaptable to various downstream tasks, the network should capture both global and local features for point clouds during self-supervised learning. Therefore, we design a compound loss function, which consists of a global representation contrastive loss $\mathcal{L}_{GRC}$ and a point cloud prediction loss $\mathcal{L}_{PCP}$.

Global representation contrastive loss: $\mathcal{L}_{GRC}$ aims to enforce consistency between cluster assignments produced by target and anchor representations. Specifically, inspired by MSN [29], we design learnable prototypes $W \in \mathbb{R}^{T\times D}$ as cluster centers, where T is the number of prototypes and D is the dimension of each prototype. We empirically set $T = 40$ in experiments. Then, as shown in Fig. 2, for the anchor encoder's output $\widetilde{C}_a$, we

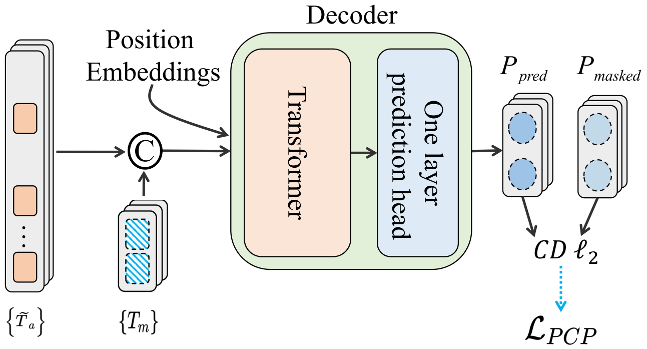

Fig. 3. Detailed architecture for point cloud prediction.

first use an MLP projection head to project it into the prototype space to obtain the anchor patches representation $\widetilde{C}_{a,p}$. Next, we calculate the cosine similarity between $\widetilde{C}_{a,p}$ and W to get the similarity distribution S_a of anchor patches:

$$S_a = \text{softmax}\left(\frac{\widetilde{C}_{a,p} \cdot W}{t_a}\right), \tag{2}$$

where $t_a \in (0, 1)$ is a temperature parameter. Similarly, we can use the same way to calculate the similarity distribution S_t of target patches. Then, to encourage to consistent similarity distribution of target patches and anchor patches, we follow MSN [29] to formulate $\mathcal{L}_{GRC}$ as

$$\mathcal{L}_{GRC} = \frac{1}{M}\left(\sum_{Z\in\{S_a\}} H\left(Z, S_t\right)\right) - \lambda H\left(\overline{S_a}\right), \tag{3}$$

where M represents the number of different masking ratios in $\{r_m\}$, i.e., $M = 3$ in experiments, $H(\cdot,\cdot)$ is the standard cross-entropy to measure the consistency of the two distributions. $H(\overline{S_a})$ is a regularization term weighted by $\lambda > 0$, where $\overline{S_a}$ denotes the averaged similarity distribution. To encourage the model to utilize the full set of prototypes, we maximize the entropy of the averaged similarity distribution. For more details, please refer to [29].

Point cloud prediction loss: $\mathcal{L}_{PCP}$ aims to measure the similarity between the predicted point clouds and the associated ground truths, thus encouraging the network to better perceive local features. Specifically, as shown in Fig. 3, we first concatenate the anchor encoder's output $\{\widetilde{T}_a\}$ with learnable placeholders $\{T_m\}$ of the masked tokens, then feed them and the position embeddings P_e into a decoder that contains four layers of Transformer and one layer of prediction header. Next, the decoder outputs the predicted point clouds P_{pred}. Finally, we use the L_2 Chamfer Distance [32] to compute the errors between P_{pred} and its masked (ground-truth) point clouds P_{masked}. Formally,

$$P_{\text{pred}} = \text{Decoder}\left(\text{Concat}\left(\{\widetilde{T}_a\}, \{T_m\}\right), P_e\right), \tag{4}$$

$$\mathcal{L}_{PCP} = \frac{1}{|P_{\text{pred}}|}\sum_{b\in P_{\text{pred}}} \min_{c\in P_{\text{masked}}} \|b - c\|_2^2 + \frac{1}{|P_{\text{masked}}|}\sum_{c\in P_{\text{masked}}} \min_{b\in P_{\text{pred}}} \|b - c\|_2^2. \tag{5}$$

Compound loss: Overall, we train Point-LGMask end-to-end by minimizing the compound loss function:

$$\mathcal{L} = \mathcal{L}_{GRC} + \beta\mathcal{L}_{PCP}, \tag{6}$$

where β is a weight to balance the importance of each loss term, and we set it to be 1,000 to make them equal.

IV. Experiments and Results

A. Pre-Training Settings

Pre-training datasets: We follow existing works [5], [6], [7], [8] to also pre-train our Point-LGMask on ShapeNet [33], which consists of over 50,000 3D models from 55 object categories. We sample 1,024 points on each 3D model, and crop each point cloud into $N = 64$ point patches following the aforementioned procedure, where each patch contains 32 points.

Implementation details: We implement our Point-LGMask on PyTorch and pre-train it for 300 epochs with a batch size of 128. Besides, we use AdamW [34] optimizer with a cosine learning rate scheduler, and the initial learning rate and weight decay are set to be 0.001 and 0.05, respectively. Generally, it takes about 20 hours to pre-train our network on ShapeNet [33] dataset using one RTX 3090 GPU.

B. Evaluation on Few-Shot Classification

SSL is expected to extract representative features from a large amount of unlabeled data, and then successfully transfer it to even a smaller dataset with promising performance. Motivated by this, we would like to delve into the performance of our approach on particularly small datasets by conducting few-shot classification task. Specifically, following previous works [5], [14], we used the standard "K-way N-shot" experimental setting, where K classes were first randomly selected and then $N + 20$ objects were randomly sampled from each class. We trained the model with the $K \times N$ samples (support set), and evaluated on the remaining $K \times 20$ samples (query set). To avoid randomness, we repeated each "K-way N-shot" experiment 10 times by independently sampling samples, then reported the mean accuracy and standard deviation over 10 times.

Evaluation on real-world data: We first conduct few-shot classification on the real-captured ScanObjectNN [10] which is collected from the real world including background and occlusions. We used the pre-trained network parameters for initialization and re-trained our top branch in Fig. 2 on the same training samples following CrossPoint [14], which is connected with a classification head. We reproduced Point-BERT and Point-MAE using the code and pre-trained weights provided in their papers. As shown in Table I, our method yields the highest accuracies across every few-shot setting on real-captured samples. Particularly, we can achieve a 4.2% performance improvement on "5-way 1-shot" setting against the second-best method. We believe that the excellent performance of Point-LGMask compared to other pre-training methods is mainly because of extracting global and local features in the pre-training stage.

Evaluation on synthetic data: We further conduct few-shot classification on ModelNet40 [37] and the results are shown in

TABLE I
COMPARING THE FEW-SHOT CLASSIFICATION RESULTS ON SCANOBJECTNN

Methods	5-way			
	1-shot	3-shot	5-shot	10-shot
Point-BERT [5]	47.1 ± 7.1	66.5 ± 6.7	75.9 ± 4.4	82.8 ± 6.3
Point-MAE [8]	43.4 ± 9.5	62.1 ± 6.5	73.9 ± 8.3	82.0 ± 5.5
Point-LGMask (Ours)	**51.3 ± 9.0**	**66.8 ± 7.5**	**79.6 ± 3.9**	**86.0 ± 6.5**
Methods	10-way			
	1-shot	3-shot	5-shot	10-shot
Point-BERT [5]	39.4 ± 5.3	51.5 ± 5.0	63.4 ± 2.8	70.5 ± 4.0
Point-MAE [8]	34.9 ± 4.3	49.0 ± 4.6	61.4 ± 3.7	72.5 ± 2.7
Point-LGMask (Ours)	**40.9 ± 6.0**	**53.9 ± 6.0**	**65.0 ± 3.5**	**73.9 ± 4.0**
Methods	15-way			
	1-shot	3-shot	5-shot	10-shot
Point-BERT [5]	27.6 ± 3.1	47.2 ± 2.6	54.5 ± 3.7	66.2 ± 2.2
Point-MAE [8]	27.3 ± 2.9	41.4 ± 3.1	54.9 ± 3.7	68.3 ± 2.3
Point-LGMask (Ours)	**29.6 ± 3.1**	**48.9 ± 3.1**	**58.5 ± 2.9**	**70.4 ± 2.2**

We report the mean accuracy (%) and standard deviation over 10 independent experiments.

TABLE II
COMPARING THE FEW-SHOT CLASSIFICATION RESULTS ON MODELNET40

Methods	5-way		10-way	
	10-shot	20-shot	10-shot	20-shot
DGCNN [35]	91.8 ± 3.7	93.4 ± 3.2	86.3 ± 6.2	90.9 ± 5.1
DGCNN-OcCo [36]	91.9 ± 3.3	93.9 ± 3.2	86.4 ± 5.4	91.3 ± 4.6
Transformer + OcCo [5]	94.0 ± 3.6	95.9 ± 2.3	89.4 ± 5.1	92.4 ± 4.6
Point-BERT [5]	94.6 ± 3.1	96.3 ± 2.7	91.0 ± 5.4	92.7 ± 5.1
MaskPoint [7]	95.0 ± 3.7	97.2 ± 1.7	91.4 ± 4.0	93.4 ± 3.5
Point-McBert [6]	97.1 ± 1.8	**98.3 ± 1.2**	92.4 ± 4.3	94.9 ± 3.7
Point-MAE [8]	96.3 ± 2.5	97.8 ± 1.8	**92.6 ± 4.1**	95.0 ± 3.0
Point-LGMask (Ours)	**97.4 ± 2.0**	98.1 ± 1.4	**92.6 ± 4.3**	**95.1 ± 3.4**

We report the mean accuracy (%) and standard deviation over 10 independent experiments.

Table II. Clearly, our method still achieves the highest mean accuracies compared to all the existing methods over most few-shot settings. The excellent results of few-shot classification show that our Point-LGMask can balance training stability and model generalization ability on datasets with few training samples by virtue of our proposed multi-ratio masking.

C. Evaluation on 3D Shape Classification

We evaluate our Point-LGMask's generalization ability in transferring knowledge from synthetic data to real-world data. Here, we performed classification on ScanObjectNN [10] which has 2,902 3D objects in 15 categories, and we use the main three variants: OBJ-BG, OBJ-ONLY, and PB-T50-RS. Note that, our model is pre-trained on the synthetic ShapeNet dataset. Table III shows the experimental results. As we can see, Point-LGMask achieves the highest accuracies under PB-T50-RS setting (the hardest variant), and yields the second-best accuracy on OBJ-BG and OBJ-ONLY settings. This indicates that our Point-LGMask has strong knowledge generalization ability even on real-scanned data.

TABLE III
COMPARING THE SHAPE CLASSIFICATION RESULTS ON THE REAL-SCANNED SCANOBJECTNN DATASET

Methods	OBJ-BG	OBJ-ONLY	PB-T50-RS
PointNet [30]	73.3	79.2	68.0
PointNet++ [9]	82.3	84.3	77.9
SpiderCNN [38]	77.1	79.5	73.7
PointCNN [39]	86.1	85.5	78.5
DGCNN [35]	82.8	86.2	78.1
BGA-DGCNN [10]	-	-	79.7
BGA-PN++ [10]	-	-	80.2
Transformer + OcCo [5]	84.9	85.5	78.8
Point-BERT [5]	87.4	88.1	83.1
MaskPoint [7]	88.1	89.3	84.3
Point-McBert [6]	89.0	**90.0**	84.3
Point-MAE [8]	**90.0**	88.3	85.2
Point-LGMask (Ours)	89.8	89.3	**85.3**

Bold indicates the best result and underline indicates the second-best result.

D. Evaluation on Object Part Segmentation

Next, we evaluate the performance of our method against others on object part segmentation, which can be regarded

TABLE IV
COMPARING THE PART SEGMENTATION RESULTS ON SHAPENETPART DATASET

Methods	mIoU$_C$	mIoU$_I$	aero	bag	cap	car	chair	earp.	guit.	knif.	lamp	lapt.	mot.	mug	pist.	rock.	skt.	table
PointNet [30]	80.4	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++ [9]	81.9	85.1	82.4	79.0	87.7	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 [35]	82.3	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 + OcCo [5]	83.4	85.1	83.3	85.2	88.3	79.9	90.7	74.1	91.9	87.6	84.7	95.4	75.5	94.4	84.1	63.1	75.7	80.8
Point-BERT [5]	84.1	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
MaskPoint [7]	**84.4**	86.0	84.2	85.6	88.1	80.3	91.2	79.5	91.9	87.8	86.2	95.3	76.9	95.0	85.3	64.4	76.9	81.8
Point-McBert [6]	**84.4**	**86.1**	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Point-MAE [8]	-	**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
Point-LGMask (Ours)	**84.4**	**86.1**	84.9	84.7	88.4	80.5	91.5	78.4	92.1	87.5	86.0	96.0	78.1	94.6	85.1	64.4	76.4	81.8

We report the mean IOU across all part categories MIOUC (%) and the mean IOU across all instance MIOU$_i$ (%), as well as the IOU (%) for each categories.

TABLE V
COMPARING 3D OBJECT DETECTION RESULTS ON SCANNET V2 DATASET

Methods	SSL	AP_{25}	AP_{50}
VoteNet [41]		58.6	33.5
STRL [13]	✓	59.5	38.4
Implicit Autoencoder [42]	✓	61.5	39.8
RandomRooms [43]	✓	61.3	36.2
PointContrast [44]	✓	59.2	38.0
DepthContrast [45]	✓	61.3	-
3DETR [46]		62.1	37.9
Point-BERT [5]	✓	61.0	38.3
MaskPoint [7]	✓	$\underline{63.4}$	40.6
Point-MAE [8]	✓	63.0	$\underline{42.4}$
Point-LGMask (Ours)	✓	**63.8**	**43.2**

TABLE VI
SEMANTIC SEGMENTATION RESULTS ON THE S3DIS AREA 5

Methods	Input	mAcc(%)	mIoU(%)
PointNet [30]	xyz+rgb	49.0	41.1
PointNet++ [9]	xyz+rgb	67.1	53.5
PointCNN [39]	xyz+rgb	63.9	57.3
PCT [48]	xyz+rgb	67.7	61.3
Transformer [5]	xyz	68.6	60.0
Point-BERT [5]	xyz	69.7	60.5
Point-MAE [8]	xyz	$\underline{69.9}$	$\underline{60.8}$
Point-LGMask (Ours)	xyz	**70.3**	**61.3**

The mean accuracy (MACC, %) and mean intersection-over-union (NIOU, %) across all categories are reported. "XYZ" indicates that the model only takes point cloud coordinates as input, "XYZ+RGB" means that the model requires both point cloud coordinates and RGB color information as input.

as per-point classification, thus requiring methods to extract representative local features. Here, we conduct experiment on ShapeNetPart [40] dataset, which contains 16,881 models from 16 categories. Table IV shows the results and we report the mean IoU across all part categories mIoU$_C$ (%) and the mean IoU across all instance mIoU$_I$ (%), as well as the IoU (%) for each category. Comparing the methods trained from scratch, our approach gets the highest mIoU$_C$ and mIoU$_I$, and obtains better or on-par performance compared with other pre-training methods. We believe that such promising performance benefits from our loss design that encourages both the local and global contexts embedding.

E. Evaluation on 3D Object Detection

We further evaluate the performance of our Point-LGMask on 3D object detection task, which requires methods to have a strong understanding ability of large-scale scenes. Here, we conduct an experiment on ScanNet V2 [47], which is a widely-used real-world dataset. ScanNet V2 has 1513 scene data covering a total of 21 categories of objects, where 1201 scenes are used for training and 312 scenes are used for testing. Table V shows the results in terms of AP_{25} and AP_{50}. Comparing both the methods trained from scratch and the pre-training methods, our approach gets the highest AP_{25} and AP_{50}. Particularly, our model achieves 0.4% AP_{25} and 0.8% AP_{50} gains compared to the second-best method.

F. Evaluation on 3D Semantic Segmentation

At last, we evaluate the performance of our Point-LGMask on the 3D semantic segmentation of large-scale scenes, which is a challenging task that requires an understanding of both global semantics and local geometric information. The S3DIS [49] (Stanford Large-Scale 3D Indoor Spaces) dataset provides instance-level semantic segmentation for 6 large indoor areas, comprising a total of 271 rooms and 13 semantic categories. Following common practice, we reserved area 5 for testing while using the remaining areas for training.

Table VI reports the results of our experiment. We observed significant improvement of our Point-LGMask compared to the Transformer [5] trained from scratch, with a performance gain of 1.7% mAcc and 1.3% mIoU. This result demonstrates that our Point-LGMask can substantially enhance the Transformer's capabilities in addressing such challenging downstream task. Further, our Point-LGMask also outperformed other self-supervised methods, achieving the best performance by improving 0.4% mAcc and 0.5% mIoU, compared to the second-best result offered by Point-MAE. Even when compared to approaches that rely on scene geometric features and colors (top four methods in Table VI), our Point-LGMask still exhibits comparable or even superior performance.

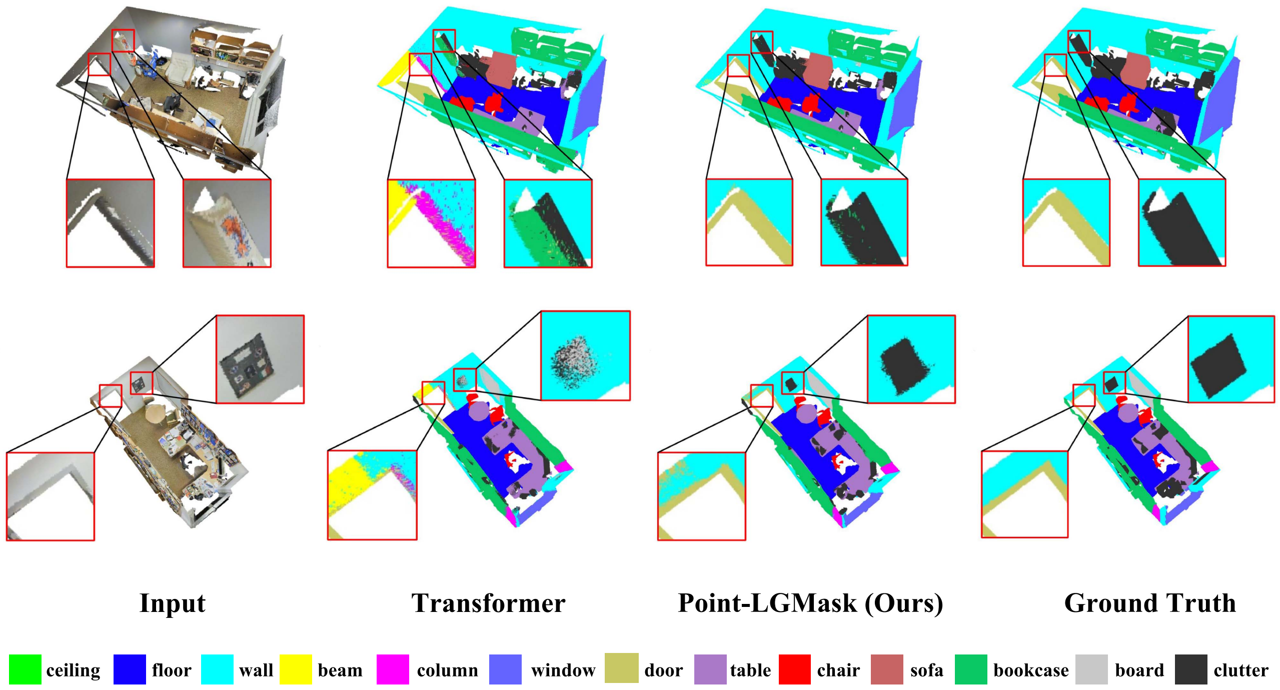

Fig. 4. Our qualitative results on 3D real-world semantic segmentation. Clearly, our Point-LGMask effectively segmented objects such as doors and clutter with high accuracy, outperforming the Transformer significantly.

Fig. 4 illustrates the qualitative results of our Point-LGMask on S3DIS. The results demonstrate that Point-LGMask achieves more accurate and precise semantic segmentation compared to the Transformer (trained from scratch). This is evidenced by the closer proximity of Point-LGMask to the ground truth labels. Our Point-LGMask effectively segmented objects such as doors and clutter with high accuracy.

G. Ablation Study

Multi-ratio Masking Strategies: First of all, we conduct ablation studies to validate the effectiveness of our introduced multi-ratio masking scheme. As shown in Table VII, we pre-trained our Point-LGMask using different masking strategies and then compared the shape classification accuracy on ScanObjectNN [10]. By analyzing the results, we can draw some discoveries as follows:

TABLE VII
COMPARING DIFFERENT MASKING STRATEGIES ON SHAPE CLASSIFICATION USING SCANOBJECTNN

Masking Strategy					Acc.
Row No.	Ratio 1	Ratio 2	Ratio 3	Mask Type	
1	0.3	-	-	Random	83.7
2	-	0.6	-	Random	84.1
3	-	-	0.9	Random	83.3
4	-	0.6	0.9	Random	84.0
5	0.3	-	0.9	Random	84.4
6	0.3	0.6	-	Random	84.8
7	0.3	0.3	0.3	Random	84.4
8	0.6	0.6	0.6	Random	84.6
9	0.9	0.9	0.9	Random	83.5
10	0.3	0.6	0.7	Random	84.9
11	0.1	0.6	0.9	Random	84.6
12	0.3	0.6	0.9	Random	**85.3**
13	0.3	0.6	0.7	Block	84.4
14	0.1	0.6	0.9	Block	83.7
15	0.3	0.6	0.9	Block	84.5

1) *When the number of mask ratios is fixed, different values of mask ratios affect the results:* For example, when there is only one single mask ratio, by comparing Rows #1-#3, we can find that using a mask ratio of 0.6 results in a better performance than a ratio of 0.3 or 0.9. This may be attributed to the fact that a mask ratio of 0.6 maintains a suitable balance between preserving geometric information and generating more difficult self-supervised signals.
2) *To some extent, increasing the number of mask ratios leads to a better performance:* However, we have to clarify that too many mask ratios will definitely increase the training time, thus degrading the model performance. Experimentally, we find that the setting of {0.3, 0.6, 0.9} is more suitable.
3) *Under the same number of masks, using different mask ratios outperforms using the same ratios:* For example, by comparing Rows #7-#9 vs. Rows #10-#12, we can observe that even if both use three masks, increasing the diversity inside the masks works better.
4) *With the same value of mask ratio, increasing the number of masks leads to a better performance:* For example, by comparing Row #1 vs. #7, Row #2 vs. #8, or Row #3 vs. #9,

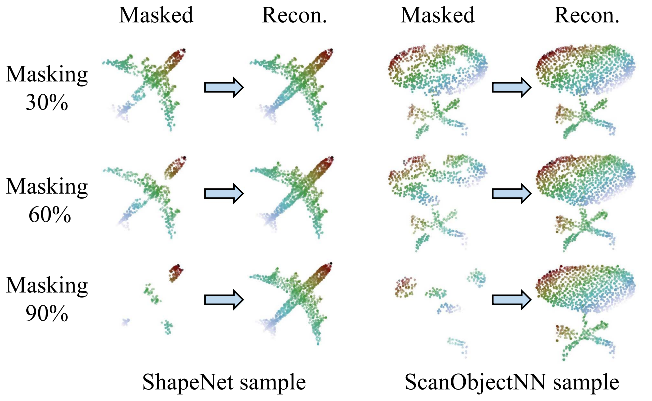

Fig. 5. Our reconstruction results from masked point clouds under different masking ratios. Clearly, even masking 90% of the total points, our Point-LGMask can still successfully recover the overall shape with fine details.

even if the values of mask ratios are the same, increasing the number of masks works better.

5) *Generally, random masking works better than block masking:* In the limited multi-ratio masking strategy search experiment, we find that random masking achieved the highest performance than that of block masking; see particularly Row #12 vs. #15.

Loss Design: We further validate the contribution of each term in (6) by removing either $\mathcal{L}_{PCP}$ or $\mathcal{L}_{GRC}$, which results in the accuracy of 83.5% and 84.7% respectively, as compared to 85.3% of our full pipeline on shape classification task using ScanObjectNN [10]. Results show that each loss term contributes to a better point cloud representation.

H. Network Analysis and Discussions

Visualization of reconstruction results: Fig. 5 shows our predicted point clouds P_{pred} given unseen ShapeNet sample (left) and ScanObjectNN [10] sample (right) under different masking ratios. Clearly, our Point-LGMask can successfully reconstruct the missing parts even masking 90%, validating that it has learned rich geometrical knowledge via our designed SSL scheme and shows strong generalization ability on even unseen real-world samples.

Effect of the number of prototypes: In experiments, we set the number of learnable prototypes to be $T = 40$. We also conduct experiments to test the sensitivity of our network to different values of T. Results show that when $T \in [20, 64]$, the accuracy fluctuation is less than 0.3%.

Limitations: Despite the promising performance that our method has achieved, one limitation is that insufficient training samples or severely uneven class size would certainly affect the network's capability. However, such a requirement for training data also appears in typical self-supervised methods. On the other hand, compared to single-ratio masking, our multi-ratio masking scheme is more time-consuming in the pre-training stage. This also restricts us to conduct experiments only on the combination of three different ratios, and it is possible that the combination of more than three ratios may lead to a better performance.

V. Conclusion

In this work, we propose Point-LGMask, a novel method to embed both local and global contexts with multi-ratio masking scheme, which is quite effective for self-supervised feature learning of point clouds. We introduce multiple masking ratios to replace a single fixed ratio, and also formulate a global representation pretext task and a local perception pretext task to drive the model to extract knowledge from large unlabeled samples. Extensive experiments on few-shot classification, shape classification, object part segmentation, 3D object detection, and 3D semantic segmentation show that our Point-LGMask has strong representation ability, particularly facilitating the challenging few-shot task. In the future, we plan to investigate the potential of multi-modality feature learning by incorporating 2D images into pretext tasks to enrich the knowledge learning, instead of only relying on 3D point clouds. Furthermore, we intend to explore the extending of our approach to process 3D CAD data by leveraging existing methods [50], [51]. We believe that these extensions will further advance geometric deep learning.

References

[1] J. Devlin et al., "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2019, pp. 4171–4186.

[2] K. He et al., "Masked autoencoders are scalable vision learners," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16000–16009.

[3] Z. Xie et al., "SimMIM: A simple framework for masked image modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9653–9663.

[4] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[5] X. Yu et al., "Point-BERT: Pre-training 3D point cloud transformers with masked point modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 19313–19322.

[6] K. Fu et al., "Point-MCBERT: A multi-choice self-supervised framework for point cloud pre-training," 2022, *arXiv:2207.13226*.

[7] H. Liu et al., "Masked discrimination for self-supervised learning on point clouds," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 657–675.

[8] Y. Pang et al., "Masked autoencoders for point cloud self-supervised learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 604–621.

[9] C. R. Qi et al., "PointNet : Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[10] M. A. Uy et al., "Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1588–1597.

[11] Z. Han, X. Wang, Y. -S. Liu, and M. Zwicker, "Multi-angle point cloud-VAE: Unsupervised feature learning for 3D point clouds from multiple angles by joint self-reconstruction and half-to-half prediction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 10441–10450.

[12] Y. Cai et al., "Learning a structured latent space for unsupervised point cloud completion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 5543–5553.

[13] S. Huang et al., "Spatio-temporal self-supervised representation learning for 3d point clouds," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 6535–6545.

[14] M. Afham et al., "CrossPoint: Self-supervised cross-modal contrastive learning for 3D point cloud understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9902–9912.

[15] G. Mei et al., "Unsupervised point cloud pre-training via contrasting and clustering," in *Proc. IEEE Int. Conf. Image Process.*, 2022, pp. 66–70.

[16] O. Poursaeed et al., "Self-supervised learning of point clouds via orientation estimation," in *Proc. Int. Conf. 3D Vis.*, 2020, pp. 1018–1028.

[17] X. Li et al., "Leveraging se(3) equivariance for self-supervised category-level object pose estimation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 15370–15381.

[18] X. Dong et al., "PECO: Perceptual codebook for BERT pre-training of vision transformers," 2021, *arXiv:2111.12710*.

[19] J. Zhou et al., "iBOT: Image BERT pre-training with online tokenizer," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–12.

[20] H. Bao et al., "BEiT: BERT pre-training of image transformers," in *Proc. Int. Conf. Learn. Representations*, 2022, pp. 1–13.

[21] Z. Wu et al., "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3733–3742.

[22] Y. Tian et al., "Contrastive multiview coding," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 776–794.

[23] T. Chen et al., "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[24] O. Henaff, "Data-efficient image recognition with contrastive predictive coding," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 4182–4192.

[25] J.-B. Grill et al., "Bootstrap your own latent a new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 21271–21284.

[26] X. Chen and K. He, "Exploring simple Siamese representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15750–15758.

[27] X. Chen et al., "An empirical study of training self-supervised vision transformers," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 9640–9649.

[28] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–12.

[29] M. Assran et al., "Masked Siamese networks for label-efficient learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, vol. 2022, pp. 456–473.

[30] C. R. Qi et al., "Pointnet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.

[31] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–11.

[32] H. Fan et al., "A point set generation network for 3D object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 605–613.

[33] A. X. Chang et al., "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*.

[34] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 1–10.

[35] Y. Wang et al., "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, 2019.

[36] H. Wang et al., "Unsupervised point cloud pre-training via occlusion completion," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 9782–9792.

[37] Z. Wu et al., "3D shapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1912–1920.

[38] Y. Xu et al., "SpiderCNN: Deep learning on point sets with parameterized convolutional filters," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 87–102.

[39] Y. Li et al., "PointCNN: Convolution on $\mathcal{X}$-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–11.

[40] L. Yi et al., "A scalable active framework for region annotation in 3D shape collections," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–12, 2016.

[41] C. R. Qi et al., "Deep hough voting for 3D object detection in point clouds," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9277–9286.

[42] S. Yan et al., "Implicit autoencoder for point cloud self-supervised representation learning," 2022, *arXiv:2201.00785*.

[43] Y. Rao et al., "Randomrooms: Unsupervised pre-training from synthetic shapes and randomized layouts for 3D object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 3283–3292.

[44] S. Xie et al., "Pointcontrast: Unsupervised pre-training for 3D point cloud understanding," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 574–591.

[45] Z. Zhang et al., "Self-supervised pretraining of 3D features on any point-cloud," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 10252–10263.

[46] I. Misra et al., "An end-to-end transformer model for 3D object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 2906–2917.

[47] A. Dai et al., "ScanNet: Richly-annotated 3D reconstructions of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5828–5839.

[48] M.-H. Guo et al., "PCT: Point cloud transformer," *Comput. Vis. Media*, vol. 7, no. 2, pp. 187–199, 2021.

[49] I. Armeni et al., "3D semantic parsing of large-scale indoor spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1534–1543.

[50] G. Sharma et al., "CSGNet: Neural shape parser for constructive solid geometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5515–5523.

[51] Y. Wu, F. He, D. Zhang, and X. Li, "Service-oriented feature-based data exchange for cloud-based design and manufacturing," *IEEE Trans. Serv. Comput.*, vol. 11, no. 2, pp. 341–353, Mar./Apr. 2018.

Yuan Tang received the bachelor's degree from the College of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China, in 2019. He is currently working toward the Ph.D. degree with the Embedded and Pervasive Computing Lab, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. His research interests include 3D representation learning, self-supervised learning.

Xianzhi Li received the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. She is currently an Associate Professor with the Huazhong University of Science and Technology, Wuhan, China. She was a Postdoctoral Fellow with The Chinese University of Hong Kong. Her research interests include 3D vision, computer graphics, and deep learning.

Jinfeng Xu received the bachelor's degree from the College of Control Science and Engineering, Shandong University, Jinan, China, in 2020. He is currently working toward the Ph.D. degree with the Embedded and Pervasive Computing Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. His research interests include deep learning, 3D vision and scene understanding.

Qiao Yu received the bachelor's degree from the Huazhong University of Science and Technology, Wuhan, China, in 2019. She is currently working toward the Ph.D. degree with the Embedded and Pervasive Computing Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology. Her research interests include deep learning, 3D vision and scene understanding.

Long Hu has been an Assistant Professor with the School of Computer Science and Technology, Huazhong University of Science and Technology (HUST), Wuhan, China, since 2017. From 2015 to April 2017, he was a Visiting Student with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada. His research interests include the Internet of Things, software defined networking, caching, 5G, body area networks, body sensor networks, and mobile cloud computing.

Yixue Hao (Member, IEEE) received the Ph.D. degree in computer science from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2017. He is currently an Associate Professor with the School of Computer Science and Technology, HUST. His research interests include 5G networks, the Internet of Things, edge computing, edge caching, and cognitive computing.

Min Chen (Fellow, IEEE) has been a Full Professor with School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. He is also the Director of Embedded and Pervasive Computing (EPIC) Lab with the Huazhong University of Science and Technology, Wuhan, China. He is the Founding Chair of IEEE Computer Society Special Technical Communities on Big Data. He was an Assistant Professor with the School of Computer Science and Engineering, Seoul National University, Seoul, South Korea, before he joined HUST. He is the Chair of IEEE Globecom 2022 eHealth Symposium. His Google Scholar Citations reached more than 39 000 with an h-index of 93. His top paper was cited more than 4090 times. From 2018 to 2022, he was selected as Highly Cited Researcher. He got IEEE Communications Society Fred W. Ellersick Prize in 2017, the IEEE Jack Neubauer Memorial Award in 2019, and IEEE ComSoc APB Oustanding Paper Award in 2022. He is a Fellow of IET.