# Towards Compact 3D Representations via Point Feature Enhancement Masked Autoencoders

**Yaohua Zha[1,2], Huizhen Ji[1], Jinmin Li[1], Rongsheng Li[1], Tao Dai[3*], Bin Chen[4], Zhi Wang[1], Shu-Tao Xia[1,2]**

[1]Tsinghua Shenzhen International Graduate School, Tsinghua University
[2]Research Center of Artificial Intelligence, Peng Cheng Laboratory
[3]College of Computer Science and Software Engineering, Shenzhen University
[4]Harbin Institute of Technology, Shenzhen
chayh21@mails.tsinghua.edu.cn

## Abstract

Learning 3D representation plays a critical role in masked autoencoder (MAE) based pre-training methods for point cloud, including single-modal and cross-modal based MAE. Specifically, although cross-modal MAE methods learn strong 3D representations via the auxiliary of other modal knowledge, they often suffer from heavy computational burdens and heavily rely on massive cross-modal data pairs that are often unavailable, which hinders their applications in practice. Instead, single-modal methods with solely point clouds as input are preferred in real applications due to their simplicity and efficiency. However, such methods easily suffer from *limited 3D representations* with global random mask input. To learn compact 3D representations, we propose a simple yet effective Point Feature Enhancement Masked Autoencoders (Point-FEMAE), which mainly consists of a global branch and a local branch to capture latent semantic features. Specifically, to learn more compact features, a share-parameter Transformer encoder is introduced to extract point features from the global and local unmasked patches obtained by global random and local block mask strategies, followed by a specific decoder to reconstruct. Meanwhile, to further enhance features in the local branch, we propose a Local Enhancement Module with local patch convolution to perceive fine-grained local context at larger scales. Our method significantly improves the pre-training efficiency compared to cross-modal alternatives, and extensive downstream experiments underscore the state-of-the-art effectiveness, particularly outperforming our baseline (Point-MAE) by *5.16%*, *5.00%*, and *5.04%* in three variants of ScanObjectNN, respectively. Code is available at https://github.com/zyh16143998882/AAAI24-PointFEMAE.

## Introduction

Point cloud, as an efficient representation of 3D objects, has been widely used in extensive applications like autonomous driving, robotics, and the metaverse for its rich geometric, shape, and structural details. Recently, with the rapid advancements of deep learning-based point cloud understanding (Qi et al. 2017a; Wang et al. 2019; Xiong et al. 2023; Gao et al. 2023), masked autoencoder (MAE) based pre-training methods (Pang et al. 2022; Zhang et al. 2022b; Dong et al. 2022; Zhang et al. 2022c; Qi et al. 2023), which aim to learn latent 3D representations from vast unlabeled point clouds, have received much attention, and can be categorized into two classes, *i.e.*, single-modal (Pang et al. 2022; Zhang et al. 2022b) and cross-modal (Dong et al. 2022; Guo, Li, and Heng 2023; Zhang et al. 2022c; Qi et al. 2023) methods.

Among them, cross-modal MAE methods, leveraging insights from other modalities, have achieved remarkable performance by acquiring holistic 3D representations. However, these methods rely heavily on transferring knowledge from massive pair images or texts, which are often unavailable in practice. Specifically, they utilize pre-trained image or language models to extract cross-modal knowledge, along with techniques like projection or knowledge distillation for cross-modal knowledge transfer. Such complex operations require heavy computational cost and thus hinders their applications in practice. As shown in Table 1, cross-modal methods like Recon (Qi et al. 2023) have obtained performance gains by 5% on ScanObjectNN while requiring $5\times$ pre-training parameters, compared to the single-modal Point-MAE (Pang et al. 2022).

For these reasons, single-modal methods with solely point clouds as input are preferred in real applications due to their simplicity and efficiency (Table 1). However, existing single-modal methods rely heavily on the global random masked point cloud (shown in Figure 1 (a)) generated by the global random masking strategy to learn 3D representations, which makes the model have robust global shape perception but insufficient local detail representation. As shown in Table 2, such single-modal methods can work well on global masked point cloud (GMPC), while failing in local masked point cloud (LMPC), thus resulting in *limited 3D representations* for single-modal MAE models.

To learn compact 3D representations for point cloud, we propose a simple yet highly effective Point Feature Enhancement Masked Autoencoders (Point-FEMAE), which mainly consists of a global branch and a local branch to capture latent global and local features, respectively. Specifically, during the pre-training stage, as illustrated in Figure 2 (a), we subject a complete point cloud to both global ran-

*Corresponding author. (daitao.edu@gmail.com)

| Method | Input | Mask Strategy | Pretrain Efficiency | | | Efficacy | |
|---|---|---|---|---|---|---|---|
| | | | #Params (M) | GFLOPS | Times (h) | ScanObjectNN | ModelNet40 |
| *Single-Modal MAE-based Method* | | | | | | | |
| Point-MAE (baseline) | PC | Global Random | 29.0 | 2.3 | 13 | 85.18 | 93.8 |
| Point-M2AE | PC | Multi-Scale | 15.3 (0.5 ×) | 3.7 (1.6 ×) | 29 (2.2 ×) | 86.43 (↑ 1.25) | 94.0 (↑ 0.2) |
| *Cross-Modal MAE-based Method* | | | | | | | |
| ACT | PC | Global Random | 135.5 (4.7 ×) | 31.0 (13.5 ×) | 52 (4.0 ×) | 88.21 (↑ 3.03) | 93.7 (↓ 0.1) |
| Joint-MAE | PC & I | Global Random | - | - | - | 86.07 (↑ 0.89) | 94.0 (↑ 0.2) |
| I2P | PC & I | 2D-Guided | 74.9 (2.6 ×) | 16.8 (7.3 ×) | 64 (4.9 ×) | 90.11 (↑ 4.93) | 94.1 (↑ 0.3) |
| Recon | PC & I & L | Global Random | 140.9 (4.9 ×) | 20.9 (9.1 ×) | 34 (2.6 ×) | **90.63** (↑ 5.45) | **94.5** (↑ 0.7) |
| **Point-FEMAE** | PC | Hybrid | 41.5 (1.4 ×) | 5.0 (2.2 ×) | 21 (1.6 ×) | 90.22 (↑ 5.04) | **94.5** (↑ 0.7) |

Table 1: Comparison of single-modal and cross-modal MAE methods in terms of pre-training efficiency and representational. For pre-training efficiency, we evaluate parameters, GFLOPS, and actual pre-training time. For representational capability, we fine-tuned the pre-trained models to evaluate classification accuracy. PC is point cloud, I is images and L is language.

dom masking and local block masking to generate globally-biased and locally-biased inputs, respectively. Subsequently, a partially parameter-shared encoder is employed to capture latent global and local features in the global and local branches and rebuild the masked inputs with a branch-independent decoder. Our encoder in both branches shares the same Transformer parameters to ensure comprehensive comprehension of the global points. Furthermore, an additional Local Enhancement Module (LEM) with local patch convolution is introduced within the local branch to perceive fine-grained local context at larger scales. During the fine-tuning phase, as depicted in Figure 2(b), owing to the availability of comprehensive global and local information in the complete input point cloud, we employ the encoder from the local branch to learn compact 3D representations of the downstream task point clouds. Our main contributions are summarized as follows:

- We have found that existing single-modal MAE-based point cloud pre-training methods suffer from limited 3D representations, due to the use of a global random masking strategy.
- We propose a Point Feature Enhancement Masked Autoencoders (Point-FEMAE), which combines global and local mask reconstruction to capture latent enhanced point features. Besides, a Local Enhancement Module (LEM) is introduced into the encoder to perceive fine-grained local context at larger scales.
- Our method significantly improves the pre-training efficiency compared to cross-modal methods. Notably, extensive experiments demonstrate the effectiveness of our method over other MAE-based methods. Particularly, our method significantly outperforms Point-MAE by 5.16%, 5.00, and 5.04% in three variants of ScanObjectNN, respectively.

# Related Work

## Point Cloud Self-supervised Learning

Self-supervised Learning (SSL) has achieved remarkable success in many fields such as NLP and computer vision. This approach first applies a pretext task to learn the latent semantic information and then fine-tunes the weights of the model in the target task to achieve higher performance. Existing pretext tasks can be divided into discriminative tasks (Becker and Hinton 1992; Wu et al. 2018; Chen et al. 2020; Zhang et al. 2023) and generative tasks (He et al. 2022; Lin, Wang, and Liu 2021; Baevski et al. 2022). The discriminative approach (Xie et al. 2020) distinguishes different views of the same instance from other instances, and in the point cloud field, PointContrast (Xie et al. 2020) first explores learning 3D representations using contrast learning of features of the same points in different views. CrossPoint (Afham et al. 2022) learns point cloud representations within the 3D domain by contrast learning, and then performs further cross-mode contrast learning. Generation methods (Vincent et al. 2008; Radford et al. 2018; Devlin et al. 2018; Ferles, Papanikolaou, and Naidoo 2018; Zhang et al. 2022a) typically rely on an autoencoder to learn the latent features of the data by reconstructing the original input. Masked autoencoders (MAE) (He et al. 2022), a classical autoencoder that tries to recover the original input from a masked version, which allows the model to learn more robust features, has received a lot of research attention.

## MAE-based Point Cloud Pre-training

MAE-based point cloud pre-training methods can be grouped into two categories, *i.e.*, *single-modal* (Pang et al. 2022; Zhang et al. 2022b) and *cross-modal* (Dong et al. 2022; Guo, Li, and Heng 2023; Zhang et al. 2022c; Qi et al. 2023) methods. Point-MAE (Pang et al. 2022) pioneered the use of masked autoencoders for self-supervised pre-training in point clouds. It divides point clouds into patches and employs mini-Point-Net to extract patch embeddings. Then a mask reconstruction was performed with standard transformers and the results were impressive. Afterward, Point-M2AE (Zhang et al. 2022b) proposes a multi-scale masking strategy, but still relies on a global random masking strategy at the first scale. Subsequent work mainly focused on using cross-modal knowledge to aid point cloud model learning. For instance, ACT (Dong et al. 2022) utilized a pre-trained

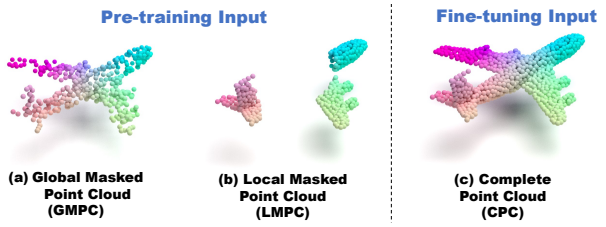


Figure 1: Differences in data distribution between pre-training and fine-tuning. (a) Global Masked Point Cloud (GMPC) input during pre-training with global random masking. (b) Local Masked Point Cloud (LMPC) input during pre-training with local block masking. (c) Complete Point Cloud (CPC) input during downstream fine-tuning.

ViT (Dosovitskiy et al. 2020) as a teacher network to guide the learning of the point cloud student network. I2P-MAE (Zhang et al. 2022c) proposed 2D-guided masking and 2D semantic reconstruction to assist point cloud model learning. Recon (Qi et al. 2023) learn from both generative modeling teachers and cross-modal contrastive teachers through ensemble distillation. Other MAE-based works (Chen et al. 2023; Yang et al. 2023; Tian et al. 2023) focus on using scene and LiDAR point clouds for pre-training, specifically for detection tasks. IDPT (Zha et al. 2023b) first proposed to introduce prompt tuning in pre-trained point cloud models. Our work focuses on single-modality point cloud pre-training to learn compact 3D representations.

## Methodology

### Observations

Despite the high efficiency, existing single-modal MAE-based pre-training pipelines with global/local random mask strategies obtain much worse performance than cross-modal methods (as shown in Table 1). It still remains unknown how the random mask strategies affect the single-modal MAE models. To this end, we first identified a substantial gap in the data distribution between the input data during pre-training and fine-tuning in the context of existing MAE-based methods. During the pre-training stage, conventional masked autoencoders typically employ a global random masking strategy to learn 3D representations, as shown in Figure 1(a), where a portion of the points is randomly masked. This masking strategy retains the global shape of the point cloud while sacrificing local details. Another strategy of local block masking randomly masks entire point blocks from the complete point cloud at the same ratio, preserving some local details but disrupting global shapes, as shown in Figure 1 (b), which has been demonstrated to yield limited performance (Yu et al. 2022; Pang et al. 2022). However, during the fine-tuning stage, complete point clouds containing full information are often utilized to learn 3D representations, as depicted in Figure 1 (c).

Our empirical observations suggest that such masked input during the pre-training stage may learn limited 3D representation due to the lack of complete information. Specifically, we employ two straightforward masking strategies: global random mask and local block mask, illustrated in Figure 1 (a) and (b), to dissect the representation efficacy of Point-MAE models pre-trained with these inputs. We assess the models' performance across reconstruction and classification tasks on pertinent test datasets. By introducing point cloud inputs biased toward local details (LMPC) and biased toward global shapes (GMPC) into the model, we gauge its competence in capturing both global and local point representations.

| Pre-training Model | Reconstruction (↓) | | Classification (↑) | |
|---|---|---|---|---|
| | GMPC | LMPC | GMPC | LMPC |
| Point-MAE w/ GM | 2.1902 | 2.8538 | 92.77 | 88.98 |
| Point-MAE w/ LM | 2.3533 | 2.4064 | 92.08 | 88.81 |
| **Point-FEMAE** | **2.1880** | **2.3941** | **93.46** | **89.33** |

Table 2: Models with varying mask strategies are assessed using LMPC and GMPC for classification and reconstruction. We measure the reconstructed CD distance on the ShapeNet test set. Additionally, we gauge the classification accuracy on ScanObjectNN (OBG-BG).

The rationale behind this is as follows: for a model utilizing a global random masking strategy, the GMPC inputs are sparsely and randomly spread across the entire object, causing local details to be severely disrupted. Despite this, the overall global shape remains preserved, leading the model to prioritize extracting global features. Conversely, in the case of LMPC inputs, all points are clustered within a few local regions, prompting the model to emphasize learning representations centered on the local surface. Consequently, models exhibiting proficiency in GMPC highlight strong global representation, while those excelling in LMPC underscore potent local representation capabilities.

As illustrated in Table 2, the Point-MAE w/ global random masking, demonstrates impressive reconstruction and classification results when tested on GMPC, but its performance is subpar on LMPC. This observation suggests that the model excels in global representation capabilities. Conversely, the Point-MAE w/ local block masking also displays superior performance on GMPC as opposed to LMPC. However, in comparison to global random masking, local block masking encounters a more substantial decline in GMPC performance and a greater enhancement in LMPC performance.

The above observations indicate that existing single modal pre-trained models employing these two straightforward masking strategies lack the ability to excel simultaneously in both LMPC and GMPC, *i.e.*, these models fail to effectively capture both local and global representations. Previous research (Qi et al. 2017b; Wang et al. 2019; Li et al. 2021; Wu, Qi, and Fuxin 2019) has demonstrated that models capable of robustly representing both global and local features exhibit higher potential. This insight motivates us to develop a model that learns compact 3D representations by comprehensively exploring global and local information.

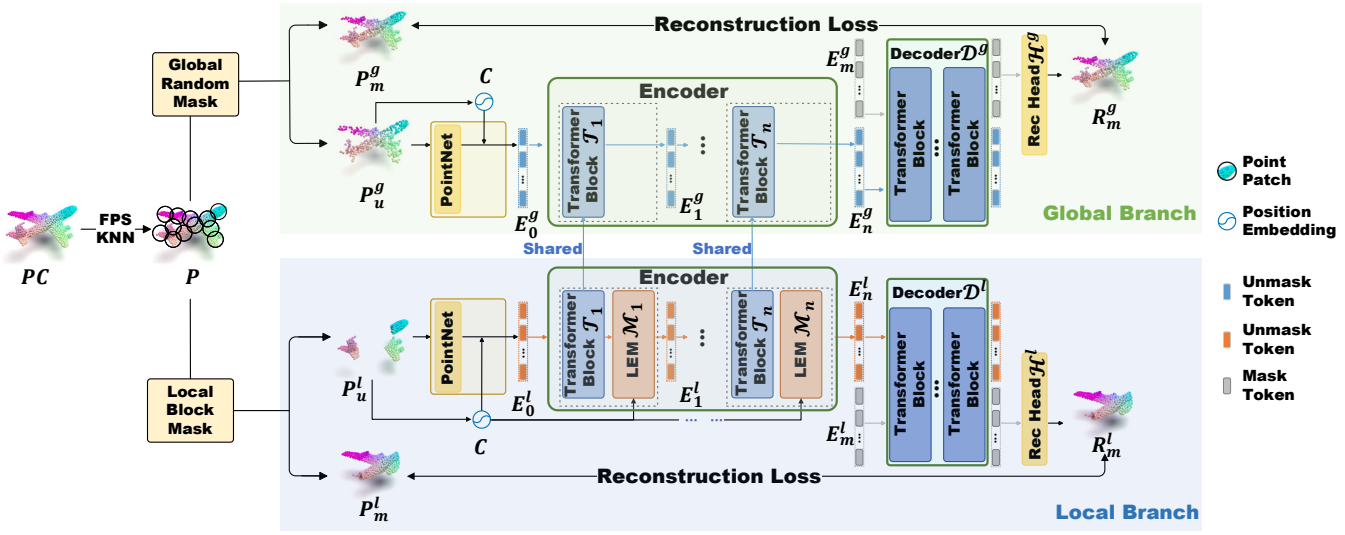


Figure 2: The pipeline of our Point-FEMAE. During the pre-training stage, we perform mask reconstruction in both the global and local branches to learn compact 3D representations. During the fine-tuning stage, we only employ the encoder of the local branch to learn the 3D representation of downstream data.

## Point Feature Enhancement Mask Autoencoders

The overall pipeline of our point feature enhancement masked autoencoders (Point-FEMAE) is shown in Figure 2. During the pre-training stage, due to the issue of information loss in masked inputs, we performed mask reconstruction in both the global and local branches to learn compact 3D representations. During the fine-tuning stage, owing to the complete input, we only employ the encoder of the local branch to learn the 3D representation of downstream data.

**Masking and Embedding.** Given a point cloud $\boldsymbol{PC} \in \mathbb{R}^{N\times 3}$ with $\boldsymbol{N}$ points, we initially divide it into $\boldsymbol{p}$ point patches $\boldsymbol{P} \in \mathbb{R}^{p\times m\times 3}$ by farthest point sampling (FPS) and K-Nearest Neighborhood (KNN), with each point patch comprising $\boldsymbol{m}$ local points. Subsequently, in the global branch, we apply global random patch masking to yield unmasked patches $\boldsymbol{P}_{\boldsymbol{u}}^{\boldsymbol{g}} \in \mathbb{R}^{(1-r)p\times m\times 3}$ and masked patches $\boldsymbol{P}_{\boldsymbol{m}}^{\boldsymbol{g}} \in \mathbb{R}^{rp\times m\times 3}$, where $\boldsymbol{r}$ denotes the mask ratio. Analogously, within the local branch, we utilize random local block masking to generate $\boldsymbol{P}_{\boldsymbol{u}}^{\boldsymbol{l}} \in \mathbb{R}^{(1-r)p\times m\times 3}$ and $\boldsymbol{P}_{\boldsymbol{m}}^{\boldsymbol{l}} \in \mathbb{R}^{rp\times m\times 3}$. Finally, $\boldsymbol{P}_{\boldsymbol{u}}^{\boldsymbol{g}}$ and $\boldsymbol{P}_{\boldsymbol{u}}^{\boldsymbol{l}}$ are embedded via a light PointNet, and positional encodings are incorporated to derive block tokens $\boldsymbol{E}_{\boldsymbol{0}}^{\boldsymbol{g}} \in \mathbb{R}^{(1-r)p\times C}$ and $\boldsymbol{E}_{\boldsymbol{0}}^{\boldsymbol{l}} \in \mathbb{R}^{(1-r)p\times C}$ for the global and local branches.

**Encoder.** We employ a share-parameter Transformer encode to extract features from the unmasked patches in both the global and local branches. This encoder consists of a series of $\boldsymbol{n}$ encoder layers, each incorporating a standard Transformer block and a Local Enhancement Module (LEM), as depicted in Figure 3. The Transformer layer integrates multi-head Self-Attention and a feed-forward network, predominantly focused on perceiving global information. The local enhancement module (LEM), situated after the Transformer Block, is mainly designed to capture local information about the object, during the fine-tuning phase and the local branch of pre-training.

Specifically, for the global branch, during the $\boldsymbol{i}$-th layer forward phase, the feature $\boldsymbol{E}_{\boldsymbol{n}}^{\boldsymbol{g}}$ only passes through the $\boldsymbol{i}$-th standard Transformer block $\mathcal{T}_i$, allowing the standard Transformer to focus more on the global feature representations. For the local branch, the feature $\boldsymbol{E}_{\boldsymbol{n}}^{\boldsymbol{l}}$ passes through the $\boldsymbol{i}$-th standard Transformer block and is then fed into the $\boldsymbol{i}$-th Local Enhancement Module $\mathcal{M}_i$, enabling the Local Enhancement Module to focus more on representing local features. Finally, after $\boldsymbol{n}$ layers of forward propagation, the two branches yield the features $\boldsymbol{E}_{\boldsymbol{n}}^{\boldsymbol{g}}$ and $\boldsymbol{E}_{\boldsymbol{n}}^{\boldsymbol{l}}$, respectively. The forward process of each layer is defined as

$$[\boldsymbol{E}_{\boldsymbol{i}}^{\boldsymbol{g}}; \boldsymbol{E}_{\boldsymbol{i}}^{\boldsymbol{l}}]_0 = [\mathcal{T}_i(\boldsymbol{E}_{\boldsymbol{i-1}}^{\boldsymbol{g}}); \mathcal{T}_i(\mathcal{M}_i(\boldsymbol{E}_{\boldsymbol{i-1}}^{\boldsymbol{g}}))]_0, \qquad (1)$$

where $\boldsymbol{i}$ takes values from $\mathbf{1}$ to $\boldsymbol{n}$, and $[;]_0$ denotes concatenation along the batch dimension.

**Local Enhancement Module.** Existing MAE-based methods have exhibited limited local representation, primarily relying on PointNet (Qi et al. 2017a) for extracting patch embeddings to represent limited local contexts. This approach is hindered by two key issues: 1) PointNet inherently lacks localization capabilities, and 2) it struggles to effectively capture localization at broader scales. To tackle these issues, drawing inspiration from Edge-Conv (Wang et al. 2019), we introduce a local patch convolution with coordinate-based nearest neighbors at the patch scale as a dedicated local enhancement module (LEM), to perceive fine-grained local context at larger scales.

Specifically, for each patch token $E_{i-1}^{l}$, it first undergoes a Transformer Block to yield the current patch tokens $E_i^l$. The patch coordinates $C$ of this patch undergo $K$-Nearest

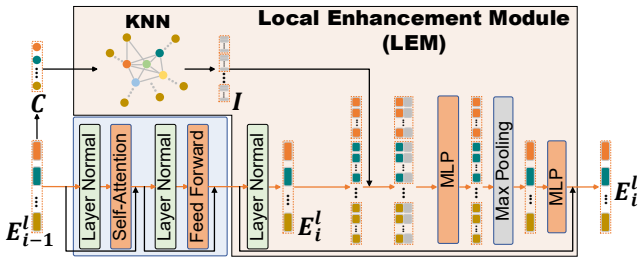


Figure 3: The Encoder Layer's structure, where each layer incorporates a globally oriented Transformer Block and a locally oriented LEM. Within the LEM, information from $k$ nearest neighbor patches is fused based on the patches' coordinates, facilitating a broader scope of local perception.

Neighbor (KNN) to obtain the indices $I$ of the $K$ nearest neighboring patches. Through these indices, the relative edges between patches are calculated (*e.g.*, for patches $a$ and $b$ as neighbors, the edge is computed as $E_i^l(a)$ - $E_i^l(b)$). Each patch in $E_i^l$ is then replicated $K$ times, and concatenated with the corresponding edges to form the final edge tensor $G_i$. We apply a single-layer MLP for dimension reduction and use Max pooling to aggregate the $K$ local edges. Lastly, the result goes through another MLP to yield the output tokens $E_i^l$ for the $i$-th layer.

**Decoder.** We employ two distinct decoders, $\mathcal{D}^g$ and $\mathcal{D}^l$, both structured identically. In the local branch, we first concatenate the encoder output $\boldsymbol{E_n^l}$ with randomly initialized learnable mask tokens $\boldsymbol{E_m^l}$ and direct this composite input into $\boldsymbol{D^l}$. Subsequently, we pass the output $\boldsymbol{E_m^l}$ through a Linear Head $\mathcal{H}^l$ for coordinate reconstruction, yielding $\boldsymbol{R_m^l}$. Finally, we calculate the reconstruction loss between $\boldsymbol{R_m^l} \in \mathbb{R}^{\boldsymbol{rp}\times\boldsymbol{m}\times\boldsymbol{3}}$ and the ground truth $\boldsymbol{P_m^l}$. Similar processes are undertaken for the global branch. Specifically, the forward process of each layer is defined as

$$\boldsymbol{R_m^l} = \mathcal{H}^l(\mathcal{D}^l([\boldsymbol{E_n^l}; \boldsymbol{E_m^l}]_1)[:, \boldsymbol{rp}\ :]) \quad (2)$$

$$\boldsymbol{R_m^g} = \mathcal{H}^g(\mathcal{D}^g([\boldsymbol{E_n^g}; \boldsymbol{E_m^g}]_1)[:, \boldsymbol{rp}\ :]) \quad (3)$$

where $[;]_1$ denotes concatenation along the token dimension and $[:, \boldsymbol{rp}\ :]$ denotes the last $\boldsymbol{rp}$ patch tokens.

$$\mathcal{L} = \mathcal{CD}(\boldsymbol{R_m^g}, \boldsymbol{P_m^g}) + \mathcal{CD}(\boldsymbol{R_m^l}, \boldsymbol{P_m^l}) \quad (4)$$

**Loss Function.** We use the $\boldsymbol{l_2}$ Chamfer Distance (Fan, Su, and Guibas 2017) ($\mathcal{CD}$) as our reconstruction loss. Our reconstruction target is to recover the coordinates of the local and global branch masked point patches. Our loss function $\mathcal{L}$ is given in Eq. 4

# Experiments

## Pre-training on ShapeNet

We use ShapeNet (Chang et al. 2015) as our pre-training dataset, encompassing over 50,000 distinct 3D models spanning 55 prevalent object categories. We extract 1024 points from each 3D model to serve as input for pre-training. The input point cloud is further divided into 64 point patches, with each patch containing 32 points. Table 1 presents a comparison of our method and other approaches concerning pre-training efficiency and efficacy.

**Single-Modal.** Compared to the single-modal baseline, Point-MAE (Pang et al. 2022), our method shows only slight increases in parameters, GFLOPS, and pre-training time, which are negligible considering the significant performance improvements. In contrast to Point-M2AE (Zhang et al. 2022b), while we possess more parameters and GFLOPS, our pre-training time is notably shorter. This variance arises from Point-M2AE's utilization of a larger input point count and patches (2048 points and 512 patches) in contrast to our utilization of 1024 points and 64 patches.

**Cross-Modal.** In comparison to cross-modal methods, our approach showcases a substantial reduction in parameters (30%∼60%), GFLOPS (16%∼30%), and pre-training times (30%∼60%) due to our simple pipeline and input. Remarkably, while maintaining pre-training efficiency, our method achieves comparable performance to the state-of-the-art cross-modal method, Recon (Qi et al. 2023), underscoring the excellence of our approach.

## Fine-tuning on Downstream Tasks

We assess the efficacy of our approach by fine-tuning our pre-trained models on downstream tasks, including classification, few-shot learning, and part segmentation.

**Object Classification.** We initially assess the overall classification accuracy of our pre-trained models on both real-scanned (ScanObjectNN (Uy et al. 2019)) and synthetic (ModelNet40 (Wu et al. 2015)) datasets. ScanObjectNN is a prevalent dataset consisting of approximately 15,000 real-world scanned point cloud samples from 15 categories. These objects represent indoor scenes and are often characterized by cluttered backgrounds and occlusions caused by other objects. ModelNet40 is a well-known synthetic point cloud dataset, comprising 12,311 meticulously crafted 3D CAD models distributed across 40 categories.

To ensure a fair comparison, we follow the practices of previous studies (Dong et al. 2022; Qi et al. 2023; Zhang et al. 2022c). For the ScanObjectNN dataset, we employ data augmentation through simple rotations and report results without voting mechanisms. Additionally, for each input point cloud, we sample 2048 points. Regarding the ModelNet40 dataset, we sample 1024 points for each input point cloud and report overall accuracy for both the without-vote and with-vote configurations and during the fine-tuning phase in ModelNet40, we only update the parameters of our local enhancement modules and the classification head to mitigate overfitting.

As presented in Table 3, in comparison to baseline Point-MAE, our method showcases substantial enhancements in accuracy across various datasets. Specifically, we observe improvements of 5.16%, 5.00%, and 5.04% on three variants of ScanObjectNN, as well as gains of 0.8% and 0.7% on the ModelNet40 (w/o vote and w/ vote respectively). Furthermore, when compared to the leading cross-modal method

| Method | #Params | ScanObjectNN | | | | ModelNet40 | | |
|---|---|---|---|---|---|---|---|---|
| | | Input | OBJ-BG | OBJ-ONLY | PB-T50-RS | Input | w/o Vote | w/ Vote |
| *Supervised Learning Only* | | | | | | | | |
| PointNet (Qi et al. 2017a) | 3.5 | 1k PC | 73.3 | 79.2 | 68.0 | 1k PC | 89.2 | - |
| PointNet++ (Qi et al. 2017b) | 1.5 | 1k PC | 82.3 | 84.3 | 77.9 | 1k PC | 90.7 | - |
| DGCNN (Wang et al. 2019) | 1.8 | 1k PC | 82.8 | 86.2 | 78.1 | 1k PC | 92.9 | - |
| SimpleView (Goyal et al. 2021) | - | 6 I | - | - | 80.5 | 6 I | 93.9 | - |
| PointMLP (Ma et al. 2022) | 12.6 | 1k PC | - | - | 85.2 | 1k PC | 94.1 | 94.5 |
| SFR (Zha et al. 2023a) | - | 20 I | - | - | 87.8 | 12 I | 93.9 | - |
| P2P-HorNet (Wang et al. 2022) | 195.8 | 40 I | - | - | 89.3 | 40 I | 94.0 | - |
| *Single-Modal Self-Supervised Learning* | | | | | | | | |
| Point-BERT (Yu et al. 2022) | 22.1 | 1k PC | 87.43 | 88.12 | 83.07 | 1k PC | 92.7 | 93.2 |
| MaskPoint (Liu, Cai, and Lee 2022) | 22.1 | 2k PC | 89.30 | 88.10 | 84.30 | 1k PC | - | 93.8 |
| Point-MAE (Pang et al. 2022) | 22.1 | 2k PC | 90.02 | 88.29 | 85.18 | 1k PC | 93.2 | 93.8 |
| Point-M2AE (Zhang et al. 2022b) | 15.3 | 2k PC | 91.22 | 88.81 | 86.43 | 1k PC | 93.4 | 94.0 |
| **Point-FEMAE** | 27.4 | 2k PC | **95.18** | **93.29** | 90.22 | 1k PC | 94.0 | **94.5** |
| *Improvement (baseline: Point-MAE)* | - | - | +5.16 | +5.00 | +5.04 | - | +0.8 | +0.7 |
| *Cross-Modal Self-Supervised Learning* | | | | | | | | |
| ACT (Dong et al. 2022) | 22.1 | 2k PC | 93.29 | 91.91 | 88.21 | 1k PC | 93.2 | 93.7 |
| Joint-MAE (Guo, Li, and Heng 2023) | - | 2k PC | 90.94 | 88.86 | 86.07 | 1k PC | - | 94.0 |
| I2P-MAE (Zhang et al. 2022c) | 15.3 | 2k PC | 94.15 | 91.57 | 90.11 | 1k PC | 93.7 | 94.1 |
| Recon (Qi et al. 2023) | 44.3 | 2k PC | **95.18** | **93.29** | **90.63** | 1k PC | **94.1** | **94.5** |

Table 3: Classification accuracy on real-scanned (ScanObjectNN) and synthetic (ModelNet40) point clouds. In ScanObjectNN, we report the overall accuracy (%) on three variants. In ModelNet40, we report the overall accuracy (%) for both without and with voting. "#Params" represents the model's parameters.

| Method | 5-way | | 10-way | |
|---|---|---|---|---|
| | 10-shot | 20-shot | 10-shot | 20-shot |
| *Single-Modal Self-Supervised Learning* | | | | |
| Point-BERT | 94.6±3.1 | 96.3±2.7 | 91.0±5.4 | 92.7±5.1 |
| MaskPoint | 95.0±3.7 | 97.2±1.7 | 91.4±4.0 | 93.4±3.5 |
| Point-MAE | 96.3±2.5 | 97.8±1.8 | 92.6±4.1 | 95.0±3.0 |
| Point-M2AE | 96.8±1.8 | 98.3±1.4 | 92.3±4.5 | 95.0±3.0 |
| **PointFEMAE** | 97.2±1.9 | 98.6±1.3 | **94.0±3.3** | **95.8±2.8** |
| *Improvement* | +0.9 | +0.8 | +1.4 | +0.8 |
| *Cross-Modal Self-Supervised Learning* | | | | |
| ACT | 96.8±2.3 | 98.0±1.4 | 93.3±4.0 | 95.6±2.8 |
| Joint-MAE | 96.7±2.2 | 97.9±1.8 | 92.6±3.7 | 95.1±2.6 |
| I2P-MAE | 97.0±1.8 | 98.3±1.3 | 92.6±5.0 | 95.5±3.0 |
| Recon | **97.3±1.9** | **98.9±1.2** | 93.3±3.9 | **95.8±3.0** |

Table 4: Few-shot learning on ModelNet40. We report the average classification accuracy (%) with the standard deviation (%) of 10 independent experiments.

Recon (Qi et al. 2023), our approach achieves almost equivalent accuracy, while requiring only 62% of the parameters. These results underscore the unmatched efficiency and efficacy of our pre-trained models, affirming the superiority of our design.

**Few-shot Learning.** Following previous works (Pang et al. 2022; Qi et al. 2023), we conduct few-shot learning experiments on the ModelNet40 (Wu et al. 2015) dataset using the "$n$-way, $m$-shot" configuration, where $n$ is the number of randomly sampled categories and $m$ is the number of samples in each category. We use the above-mentioned $n \times m$ samples for training, while 20 unseen samples from each category for testing. Following standard protocol, we conducted 10 independent experiments for each setting and reported mean accuracy with standard deviation.

As indicated in Table 4, with limited downstream fine-tuning data, our Point-FEMAE exhibits competitive performance among existing single-modal and cross-modal methods, *e.g.*+1.4% classification accuracy to Point-MAE on the 10-way 10-shot split.

**Part Segmentation.** We assess the performance of Point-FEMAE in part segmentation using the ShapeNetPart dataset (Chang et al. 2015), comprising 16,881 samples across 16 categories. Employing the same experimental settings and segmentation head as Point-MAE and the mean IoU across all categories, i.e., $\mathrm{mIoU}_c$ (%), and the mean IoU across all instances, i.e., $\mathrm{mIoU}_I$ (%) are reported. We did not include the results for Point-M2AE and I2P-MAE due to their utilization of a more intricate segmentation head.

As shown in Table 5, our Point-FEMAE exhibits competitive performance among both existing single-modal and cross-modal methods, *e.g.*+0.7% $\mathrm{mIoU}_c$ to Point-MAE (Pang et al. 2022) and slightly improvement compared to Recon (Qi et al. 2023). These results demonstrate that our approach exhibits superior performance in tasks such as part segmentation, which demands a more fine-grained understanding of point clouds, demonstrating the superiority of

| Methods | Reference | $\mathrm{mIoU}_c$ | $\mathrm{mIoU}_I$ |
|---|---|---|---|
| *Supervised Learning Only* | | | |
| PointNet (Qi et al. 2017a) | CVPR'17 | 80.4 | 83.7 |
| PointNet++ (Qi et al. 2017b) | NIPS'17 | 81.9 | 85.1 |
| PointMLP (Ma et al. 2022) | ICLR'22 | 84.6 | 86.1 |
| *Single-Modal Self-Supervised Learning* | | | |
| Transformer | NIPS'17 | 83.4 | 84.7 |
| Transformer-OcCo | ICCV'21 | 83.4 | 85.1 |
| Point-BERT | CVPR'22 | 84.1 | 85.6 |
| MaskPoint | ECCV'22 | 84.4 | 86.0 |
| Point-MAE | ECCV'22 | 84.2 | 86.1 |
| **Point-FEMAE** | - | **84.9** | 86.3 |
| *Improvement* | - | +0.7 | +0.2 |
| *Cross-Modal Self-Supervised Learning* | | | |
| ACT | ICLR'23 | 84.7 | 86.1 |
| Recon | ICML'23 | 84.8 | **86.4** |

Table 5: Part segmentation results on the ShapeNetPart. The mean IoU across all categories, i.e., $\mathrm{mIoU}_c$ (%), and the mean IoU across all instances, i.e., $\mathrm{mIoU}_I$ (%) are reported.

the compact representations learned by our method.

## Ablation Study

**Effects of data augmentation, masking strategy, and LEM.** Comparing our fine-tuning with the baseline Point-MAE on ScanObjectNN (Uy et al. 2019), our method has two main differences. 1) *Masking strategy*: we use a hybrid global and local branch point masking strategy (*e.g.*Hybrid Mask). 2) *Network architecture*: we add our Local Enhancement Module (LEM) after each standard Transformer block in the local branch. We examined the effect of each factor separately.

We designed four different structures to explore the effects of these factors, as shown in Table 6, A uses Point-MAE as the baseline, B has a simple hybrid global and local branch mask reconstruction without local enhancement module (LEM), C add our LEM at each layer of the Encoder based on the Point-MAE with global random mask, and D is our Point-FEMAE model. 1 and 2 indicate two different data augmentations.

Table 6 reports our ablation results, we can discover that: 1) simply combining two mask reconstructions can lead to a suboptimal encoder (comparing A-B). 2) introducing LEM to Point-MAE provides a slight improvement (comparing A-C), and this improvement may be due to the introduction of additional parameters, we will discuss this issue in the next subsection. 4) Comparing D with other results, we can discover a significant improvement, which illustrates the superiority of our design, which artfully combines a hybrid global and local branch masking strategy and local enhancement modules.

**Effects of Additional Parameters.** To illustrate whether our improvement is due to more parameters, we introduced the patch-independent MLP and Self-Attention module that focuses on global patches to replace our Local Enhancement Module, respectively, within our masking and reconstruction pipeline for pre-training. We reported their respective fine-tuned results on the ScanObjectNN in Table 7.

| | #Params | Hybrid Mask | LEM | PB-T50-RS |
|---|---|---|---|---|
| A | 22.1 | ✘ | ✘ | 88.41 (baseline) |
| B | 22.1 | ✔ | ✘ | 88.75 (↑ 0.34) |
| C | 27.4 | ✘ | ✔ | 89.17 (↑ 0.76) |
| D | 27.4 | ✔ | ✔ | 90.22 (↑ 1.81) |

Table 6: Effects of data augmentation, hybrid masking strategy, and LEM on the ScanObjectNN dataset.

| Addition Module | #Params (M) | PB-T50-RS |
|---|---|---|
| Hybrid Mask w/o LEM | 22.1 | 88.75 |
| Hybrid Mask w/ 1-layer MLP | 23.9 | 89.14 |
| Hybrid Mask w/ 3-layer MLPs | 27.4 | 89.17 |
| Hybrid Mask w/ Self-Attention | 29.2 | 89.42 |
| Hybrid Mask w/ LEM | 27.4 | 90.22 |

Table 7: Effects of additional network and parameters.

These outcomes demonstrate that incorporating an additional 1-layer MLP exhibits some enhancement when compared to the Hybrid Mask w/o LEM. However, with the escalation of parameters, the model exhibits a limited potential, likely due to the MLP employing shared parameters for individual patch processing, regardless of patch correlations, similar to the Transformer's feed-forward network. Similarly, the additional Self-Attention layer, requiring more parameters, yields a certain improvement, yet it parallels the behavior of the Self-Attention layer within the Transformer, consequently capping potential. These comparisons underscore that the advancement of our approach stems from the excellence of ingeniously combining the strategy of hybrid global and local branch mask reconstruction with the design based on local patch convolution, rather than being driven by additional parameters.

## Conclusions

In this paper, we first compare the pre-training efficiency and efficacy of current single-modal and cross-modal MAE-based point cloud pre-training pipelines and experimentally demonstrate that the limited 3D representation of existing single-modal MAE-based point cloud pre-training methods is due to biases in the existing masking strategies towards global and local representations. To address this issue, we propose to learn compact 3D representations via effective Point Feature Enhancement Masked Autoencoders, which mainly consist of a global branch and local branch to capture latent semantic features. Meanwhile, to further perceive fine-grained local context at larger scales, we propose a Local Enhancement Module with local patch convolution in the local branch. Extensive experiments demonstrate the advancement of our design.

## Acknowledgments

This work is supported in part by the National Key Research and Development Program of China, under Grant No. 2023YFF0905502, National Natural Science Foundation of China, under Grant (62302309,62171248), Shenzhen Science and Technology Program (Grant No. RCYX20200714114523079, JCYJ20220818101014030, JCYJ20220818101012025), and the PCNL KEY project (PCL2023AS6-1), and Tencent "Rhinoceros Birds" - Scientific Research Foundation for Young Teachers of Shenzhen University.

## References

Afham, M.; Dissanayake, I.; Dissanayake, D.; Dharmasiri, A.; Thilakarathna, K.; and Rodrigo, R. 2022. 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*, 9902–9912.

Baevski, A.; Hsu, W.-N.; Xu, Q.; Babu, A.; Gu, J.; and Auli, M. 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. *arXiv preprint arXiv:2202.03555*.

Becker, S.; and Hinton, G. E. 1992. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355(6356): 161–163.

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

Chen, A.; Zhang, K.; Zhang, R.; Wang, Z.; Lu, Y.; Guo, Y.; and Zhang, S. 2023. Pimae: Point cloud and image interactive masked autoencoders for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5291–5301.

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

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dong, R.; Qi, Z.; Zhang, L.; Zhang, J.; Sun, J.; Ge, Z.; Yi, L.; and Ma, K. 2022. Autoencoders as Cross-Modal Teachers: Can Pretrained 2D Image Transformers Help 3D Representation Learning? *arXiv preprint arXiv:2212.08320*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

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

Ferles, C.; Papanikolaou, Y.; and Naidoo, K. J. 2018. Denoising autoencoder self-organizing map (DASOM). *Neural Networks*, 105: 112–131.

Gao, K.; Bai, J.; Wu, B.; Ya, M.; and Xia, S.-T. 2023. Imperceptible and Robust Backdoor Attack in 3D Point Cloud. *IEEE Transactions on Information Forensics and Security*.

Goyal, A.; Law, H.; Liu, B.; Newell, A.; and Deng, J. 2021. Revisiting point cloud shape classification with a simple and effective baseline. In *International Conference on Machine Learning*, 3809–3820. PMLR.

Guo, Z.; Li, X.; and Heng, P. A. 2023. Joint-mae: 2d-3d joint masked autoencoders for 3d point cloud pre-training. *arXiv preprint arXiv:2302.14007*.

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

Li, G.; Müller, M.; Qian, G.; Perez, I. C. D.; Abualshour, A.; Thabet, A. K.; and Ghanem, B. 2021. Deepgcns: Making gcns go as deep as cnns. *IEEE transactions on pattern analysis and machine intelligence*.

Lin, K.; Wang, L.; and Liu, Z. 2021. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1954–1963.

Liu, H.; Cai, M.; and Lee, Y. J. 2022. Masked discrimination for self-supervised learning on point clouds. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, 657–675. Springer.

Ma, X.; Qin, C.; You, H.; Ran, H.; and Fu, Y. 2022. Rethinking network design and local geometry in point cloud: A simple residual MLP framework. *arXiv preprint arXiv:2202.07123*.

Pang, Y.; Wang, W.; Tay, F. E.; Liu, W.; Tian, Y.; and Yuan, L. 2022. Masked autoencoders for point cloud self-supervised learning. *arXiv preprint arXiv:2203.06604*.

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

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

Qi, Z.; Dong, R.; Fan, G.; Ge, Z.; Zhang, X.; Ma, K.; and Yi, L. 2023. Contrast with Reconstruct: Contrastive 3D Representation Learning Guided by Generative Pretraining. *arXiv preprint arXiv:2302.02318*.

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

Tian, X.; Ran, H.; Wang, Y.; and Zhao, H. 2023. GeoMAE: Masked Geometric Target Prediction for Self-supervised Point Cloud Pre-Training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13570–13580.

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

Vincent, P.; Larochelle, H.; Bengio, Y.; and Manzagol, P.-A. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, 1096–1103.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5): 1–12.

Wang, Z.; Yu, X.; Rao, Y.; Zhou, J.; and Lu, J. 2022. P2p: Tuning pre-trained image models for point cloud analysis with point-to-pixel prompting. *arXiv preprint arXiv:2208.02812*.

Wu, W.; Qi, Z.; and Fuxin, L. 2019. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 9621–9630.

Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; and Xiao, J. 2015. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1912–1920.

Wu, Z.; Xiong, Y.; Yu, S. X.; and Lin, D. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3733–3742.

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

Xiong, J.; Dai, T.; Zha, Y.; Wang, X.; and Xia, S.-T. 2023. Semantic Preserving Learning for Task-Oriented Point Cloud Downsampling. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Yang, H.; He, T.; Liu, J.; Chen, H.; Wu, B.; Lin, B.; He, X.; and Ouyang, W. 2023. GD-MAE: generative decoder for MAE pre-training on lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9403–9414.

Yu, X.; Tang, L.; Rao, Y.; Huang, T.; Zhou, J.; and Lu, J. 2022. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19313–19322.

Zha, Y.; Li, R.; Dai, T.; Xiong, J.; Wang, X.; and Xia, S.-T. 2023a. SFR: Semantic-Aware Feature Rendering of Point Cloud. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Zha, Y.; Wang, J.; Dai, T.; Chen, B.; Wang, Z.; and Xia, S.-T. 2023b. Instance-aware Dynamic Prompt Tuning for Pre-trained Point Cloud Models. *arXiv preprint arXiv:2304.07221*.

Zhang, C.; Zhang, C.; Song, J.; Yi, J. S. K.; Zhang, K.; and Kweon, I. S. 2022a. A survey on masked autoencoder for self-supervised learning in vision and beyond. *arXiv preprint arXiv:2208.00173*.

Zhang, R.; Guo, Z.; Gao, P.; Fang, R.; Zhao, B.; Wang, D.; Qiao, Y.; and Li, H. 2022b. Point-M2AE: Multi-scale Masked Autoencoders for Hierarchical Point Cloud Pretraining. *arXiv preprint arXiv:2205.14401*.

Zhang, R.; Wang, L.; Qiao, Y.; Gao, P.; and Li, H. 2022c. Learning 3D Representations from 2D Pre-trained Models via Image-to-Point Masked Autoencoders. *arXiv preprint arXiv:2212.06785*.

Zhang, T.; He, S.; Tao, D.; Chen, B.; Wang, Z.; and Xia, S.-T. 2023. Vision-Language Pre-training with Object Contrastive Learning for 3D Scene Understanding. *arXiv preprint arXiv:2305.10714*.