ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-6874-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICASSP49660.2025.10890081

Point-UMAE: Unet-like Masked Autoencoders for Point Cloud Self-supervised Learning

Hongliang Zeng
South China University of Technology
Guangzhou, China
scutzenghongl@gmail.com

Ping Zhang
South China University of Technology
Guangzhou, China
pzhang@scut.edu.cn

Fang Li
South China University of Technology
Guangzhou, China
cslifang@scut.edu.cn

Tingyu Ye
South China University of Technology
Guangzhou, China
cstingyuye@mail.scut.edu.cn

Jiahua Wang
South China University of Technology
Guangzhou, China
Jh_Wang_scut@outlook.com

Xianbo Yang
South China University of Technology
Guangzhou, China
yangxbscut@gmail.com

***Abstract*—Masked Autoencoders (MAE) demonstrated exceptional performance in natural language processing and 2D vision tasks and have now been introduced into point cloud representation learning. We propose Point-UMAE, a novel self-supervised learning method based on a Unet-like structure, designed to enhance the capture of local details and global semantics in point clouds. Point-UMAE employs an asymmetric encoder-decoder architecture with a top-down fine-grained masking strategy to improve multi-scale consistency. The pre-trained model achieves state-of-the-art performance across various downstream tasks. Compared to the baseline Point-BERT, our method achieves a classification performance improvement of 1% and 4.14% on the ModelNet40 and ScanObjectNN datasets, respectively. We also investigate the impact of masking strategies and encoder structures on performance.**

***Index Terms*—Point cloud, Self supervised learning, MAE.**

I. Introduction

Self-supervised learning (SSL) [1]–[5] aims to enable models to effectively learn feature representations from data through pre-training tasks. Recently, the success of MAE [6] has inspired researchers to introduce it into 3D signal processing [7]–[12]. However, the irregular sampling and sparsity of point clouds present additional challenges, requiring algorithms not only to capture local features but also to understand the global structure and geometric relationships of the point clouds.

We note that the success of Unet [13] in medical image segmentation is attributed to its design for multi-scale feature fusion and spatial information preservation. Swin-Unet [14] further demonstrated the potential of combining transformer architectures with Unet design principles, which inspired us to incorporate these advantages into self-supervised learning for point cloud signals. To the best of our knowledge, the most relevant prior work to our research is Point-M2AE [10], which was the first to introduce a multi-scale pre-training framework for point cloud SSL. However, Point-M2AE has a drawback of position information leakage at low scales. Additionally, Point-M2AE fails to fully transfer the advantages of the Unet structure to downstream models. This results in the framework overemphasizing the aggregation of global features while neglecting the construction of local detail features in downstream tasks.

In response, this paper proposes Point-UMAE, designed to effectively integrate the advantages of multi-scale structures. Our approach employs a top-down masking strategy to ensure mask consistency across different scales. Specifically, we begin by randomly masking point cloud patches at the lowest scale according to a predetermined masking ratio. Next, we apply the farthest point sampling (FPS) algorithm to determine the downsampled point cloud at higher scales, followed by using the k-nearest neighbors (KNN) algorithm to identify feature merging groups. This strategy effectively prevents position information leakage at low scales and enhances multi-scale masking efficiency.

Following the design principles of MAE, we constructed an asymmetric encoder-decoder structure during the pre-training phase. The decoder is equipped with a simplified multi-layer transformer module, specifically designed for feature interpolation of masked point cloud patches at the lowest scale. After the pre-training phase, the complete Unet-like encoder is transferred to the fine-tuning stage. This indicates that, apart from the task-specific prediction head, the weights of the downstream network model are fully initialized through the pre-training process.

The experiments demonstrate significant achievements of our method in multiple downstream tasks. After pre-training, Point-UMAE achieves an accuracy of 93.1% on the ModelNet40 [15] dataset using linear Support Vector Machine (SVM) for classification, surpassing many fully supervised learning methods. After fine-tuning, Point-UMAE achieves classification accuracies of 94.2% on the ModelNet40 [15] dataset and 91.57% on the ScanObjectNN [16] dataset. Additionally, we evaluate Point-UMAE's performance in tasks such as few-shot learning, part segmentation, and 3D object detection, where Point-UMAE demonstrates the best performance.

II. Method

As shown in Figure 1, during the pre-training phase, Point-UMAE first segments and masks the input point cloud, then learns its multi-scale features and contextual information through the encoder. Subsequently, the decoder reconstructs the original point cloud. After pre-training, the features learned by the encoder are utilized in the model structure for downstream tasks.

A. Point Patch Embedding

For a given point cloud $X \in \mathbb{R}^{N\times 3}$, we apply the farthest point sampling (FPS) strategy to downsample and obtain m_1 representative center points. Then, using the k-nearest neighbors (KNN) algorithm, we select and aggregate k_1 neighboring points for each center point, constructing local point cloud patches $P \in \mathbb{R}^{m_1\times k_1\times 3}$,

$$P = \mathrm{KNN}\left(\mathrm{FPS}\left(X\right), X\right), \quad P \in \mathbb{R}^{m_1\times k_1\times 3}. \tag{1}$$

This work was supported by the Guangdong Major Project of Basic and Applied Basic Research (2023B0303000016). *(Corresponding author: Ping Zhang.)*

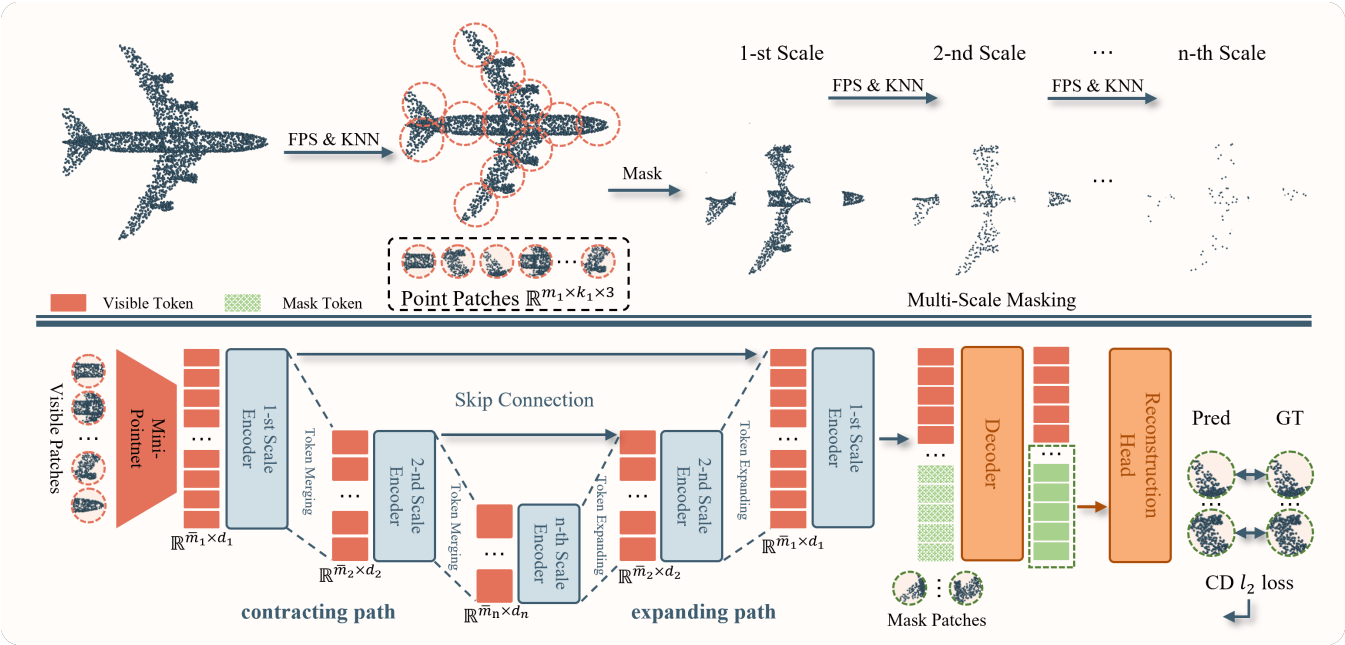

Fig. 1. **Overall architecture of our Point-UMAE**. The upper part of the figure illustrates the block division of point clouds and the top-down multi-scale masking strategy employed. Below, we elaborate on the detailed process of our method, where the encoder consists of a contraction path and an expansion path, responsible for multi-scale feature extraction and interaction of contextual information. Subsequently, a structurally simple decoder and a reconstruction head are used to reconstruct the masked point cloud.

For a predefined masking ratio α, at the initial scale s_1, we mask $m_1 \times \alpha$ proportion of point cloud patches and use these masked blocks to calculate the reconstruction loss. Starting from the second scale s_2, for each subsequent scale s_i, we first perform FPS from the set of visible point cloud blocks at the previous scale $m_{i-1} \times (1-\alpha)$ to downsample to $\bar{m}_i = m_i \times (1-\alpha)$ center points. Then, using the KNN algorithm, we determine k_i neighboring points for each center point based on these center points for feature merging.

Consistent with Point-MAE [7], we employ a Mini-PointNet [17] architecture to extract initial features for each point cloud patch. To facilitate model convergence, we choose to represent the positional information of points within each patch using relative positions, i.e., by computing the difference between a point and its corresponding patch center and normalizing the result. This relative positional representation helps the model capture local structural features and enhance its representation capabilities for point cloud data. The mathematical expression of this process can be formalized as:

$$T_1^C = \text{Mini-Pointnet}(P_{\text{vis}}), \quad T_1^C \in \mathbb{R}^{\bar{m}_1 \times d_1}. \tag{2}$$

B. U-shaped Transformer Encoder

The encoder receives tokenized input T_1^C and integrates local features into global features through a contracting path. Subsequently, the expanding path, through skip connections and feature propagation mechanisms, gradually backpropagates high-level abstract features to the original scale, thus achieving comprehensive extraction of contextual information from the input data and deep interaction between multi-scale features.

During the interaction between adjacent scale encoders, the encoder of a higher-level scale s_i receives input tokens T_i^C, which are merged from the output $\bar{T}_{i-1}^C$ of the encoder at the previous scale. In the top-down masking strategy, the neighbors for token merging for each center point at the current scale have been determined. In this process, a multi-layer perceptron (MLP) and max-pooling layer are used to fuse these neighboring tokens. This integration process can be described by the following mathematical expression:

$$T_i^C = \theta_{\max}(\text{MLP}(\text{KNN}(\bar{T}_{i-1}^C)), \quad T_i^C \in \mathbb{R}^{\bar{m}_i \times d_i} \tag{3}$$

In this process, $\theta_{\max}$ represents the operation of max pooling. Subsequently, we deploy a network structure consisting of l standard vision transformer (Vit) blocks at the current scale to compute the attention mechanism and interact with context information. The description of this process is as follows:

$$\bar{T}_i^C = \text{Vit}_i(T_i^C), \quad \bar{T}_i^C \in \mathbb{R}^{\bar{m}_i \times d_i}. \tag{4}$$

The expansion path is responsible for gradually passing features from the highest scale s_n back to the initial scale, ensuring that the features of each local block can be deeply fused with global information from higher levels. This design strategy not only enhances the model's multi-scale understanding of point cloud data but also, thanks to the consistency between the input and output dimensions of the encoder, allows the model to flexibly expand and adapt to a variety of downstream application tasks. Specifically, this process is achieved through token upsampling techniques and skip-connection mechanisms. Between scales s_{i+1} and s_i, we use a token propagation module to handle T_{i+1}^E at scale s_i. This module first looks for k_{i+1} nearest neighbors for each center point at scale s_{i+1}, then uses a weighted interpolation technique inspired by PointNet++ [18] to reconstruct the tokens $\dot{T}_i^E$ at scale s_i. Subsequently, the skip-connected $\bar{T}_i^C$ and $\dot{T}_i^E$ are fused using a linear projection layer,

$$T_i^E = \theta_{proj}(\bar{T}_i^C, \dot{T}_i^E), \quad T_i^E \in \mathbb{R}^{\bar{m}_i \times d_i}; \tag{5}$$

$$\bar{T}_i^E = \text{Vit}_i(T_i^E), \quad \bar{T}_i^E \in \mathbb{R}^{\bar{m}_i \times d_i}. \tag{6}$$

TABLE I
CLASSIFICATION ON MODELNET40 [15] AND SCANOBJECTNN [16]. WE REPORT THE SHAPE CLASSIFICATION ACCURACY (%).

Methods	Flops	ModelNet40		ScanObjectNN		
		w/o Vote	w/ Vote	OBJ_BG	OBJ_ONLY	PB_T50_RS
PointNet [17]	-	-	89.2	73.3	79.2	68.0
PointNet++ [18]	-	-	90.7	82.3	84.3	77.9
DGCNN [19]	-	-	92.9	82.8	86.2	78.1
with Self-Supervised Representation Learning						
Point-BERT [8]	4.8G	92.7	93.2	87.43	88.12	83.07
Mask-Point [9]	4.8G	-	93.8	89.70	**89.30**	84.60
Point-MAE [7]	4.8G	93.2	93.8	90.02	88.29	85.18
Point-M2AE [10]	3.6G	93.4	94.0	91.22	88.81	86.43
Point-UMAE	**1.9G**	**93.7**	**94.2**	**91.57**	88.98	**86.95**

TABLE II
FEW-SHOT CLASSIFICATION ON MODELNET40 [15]. WE REPORT THE AVERAGE ACCURACY (%) AND STANDARD DEVIATION (%) OF 10 INDEPENDENT EXPERIMENTS.

Methods	5-way		10-way	
	10-shot	20-shot	10-shot	20-shot
DGCNN [19]	31.6 ± 2.8	40.8 ± 4.6	19.9 ± 2.1	16.9 ± 1.5
OcCo [20]	90.6 ± 2.8	92.5 ± 1.9	82.9 ± 1.3	86.5 ± 2.2
with Self-Supervised Representation Learning				
Point-BERT [8]	94.6 ± 3.1	96.3 ± 2.7	91.0 ± 5.4	92.7 ± 5.1
Mask-Point [9]	95.0 ± 3.7	97.2 ± 1.7	91.4 ± 4.0	93.4 ± 3.5
Point-MAE [7]	96.3 ± 2.5	97.8 ± 1.8	92.6 ± 4.1	95.0 ± 3.0
Point-M2AE [10]	96.8 ± 1.8	98.3 ± 1.4	92.3 ± 4.5	95.0 ± 3.0
Point-UMAE	**97.1 ± 1.9**	**98.6 ± 0.7**	**92.6 ± 4.0**	**95.1 ± 3.0**

TABLE III
PART SEGMENTATION ON SHAPENETPART [23]. WE REPORT THE MEAN IOU (%) OF ALL PART CATEGORIES ('$mIoU_C$') AND THE MEAN IOU OF ALL INSTANCES ('$mIoU_I$') IN THE DATASET.

Methods	mIoU_C	mIoU_I
PointNet [17]	80.39	83.70
PointNet++ [18]	81.85	85.10
DGCNN [19]	82.33	85.20
with Self-Supervised Representation Learning		
Transformer + OcCo [8]	83.42	85.10
Point-BERT [8]	84.11	85.60
Mask-Point [9]	84.40	86.00
Point-MAE [7]	-	86.10
GPM [24]	84.20	85.80
Point-UMAE	**84.56**	**86.13**

TABLE IV
3D OBJECT DETECTION ON SCANNETV2 [25]. WE REPORT THE AVERAGE PRECISION AT TWO DIFFERENT IOU THRESHOLDS.

Methods	AP_{25}	AP_{50}
VoteNet [26]	58.6	33.5
STRL [27]	59.5	38.4
PointContrast [28]	59.2	38.0
DepthContrast [29]	61.3	–
3DETR [30]	62.1	37.9
3DETR-m [30]	65.0	47.0
Point-BERT [8]	61.0	38.3
Mask-Point [9]	64.2	42.1
Point-UMAE	**66.1**	**48.4**

C. Point Cloud Reconstruction

We define the reconstruction of masked regions in the point cloud as the objective of our SSL approach. This process involves a lightweight decoder tasked with efficiently interpolating the masked areas' features. Additionally, the reconstruction head uses the tokens output by the decoder to recover the relative positional information. To effectively handle masked positions, we introduce a shared and trainable masking token mechanism. At the initial scale, we append $m_1 \times \alpha$ masking tokens T_{mask} after the visible tokens. These tokens are then passed to the decoder for processing, which consists of two standard Transformer blocks:

$$T^D = \text{Decoder}(\bar{T}_1^E, T_{\text{mask}}), \quad T^D \in \mathbb{R}^{m_1 \times d_1}. \tag{7}$$

We use a reconstruction head consisting of a linear projection layer to reconstruct the k_1 nearest neighbor points at the initial scale. The loss calculation is based on the L2 Chamfer distance [21], measuring the difference between the reconstructed points and the original points. In the self-supervised pretraining phase, we do not use a contrastive loss, but focus on learning the intrinsic feature representation of the point cloud through the pure masked autoencoder. The mathematical expression for this process is as follows:

$$P = \theta_{rec}(T^D), \quad P \in \mathbb{R}^{(m_1 \times \alpha) \times k_1 \times 3} \tag{8}$$

$$\mathcal{L}_{CD}(P, \hat{P}) = \frac{1}{|P|} \sum_{p \in P} \min_{\hat{p} \in \hat{P}} \|p - \hat{p}\|_2^2 + \frac{1}{|\hat{P}|} \sum_{\hat{p} \in \hat{P}} \min_{p \in P} \|\hat{p} - p\|_2^2 \tag{9}$$

where, $\hat{P}$ denotes the ground truth.

III. EXPERIMENTAL EVALUATION

A. Pre-training

We pretrained the Point-UMAE model on the ShapeNet [22] dataset, which contains 55 different categories totaling 57,448 3D shapes. During the pretraining process, we set the scale value to 3 and used a top-down random masking strategy with a masking rate of 0.75 to mask the point cloud blocks. For the multiscale structure, we designed the corresponding numbers of point cloud blocks as $m = [128, 64, 32]$, and set the token dimensions as $d = [384, 576, 768]$. At each scale, the encoder consists of two transformer blocks. Additionally, the decoder also consists of two transformer blocks.

B. Shape Classification

Table I presents the evaluation results for shape classification accuracy. On the ModelNet40 dataset [15], our method achieved the highest accuracy in both settings, with and without voting, surpassing the baseline Point-BERT [8] by 1% in the voting scenario. On the ScanObjectNN dataset [16], across three different partition settings, our method achieved the best performance in both the "OBJ-BG" and the most challenging "PB_T50_RS" sections, with accuracies of 91.57% and 86.95%, respectively. Under the "OBJ-ONLY" condition, Point-UMAE achieved the second-best accuracy at 88.98%. In all three cases, our method outperformed Point-M2AE [10].

C. Few-shot Classification

We evaluated the few-shot learning performance of Point-UMAE by conducting N-way, K-shot experiments on the ModelNet40 [15] dataset. Specifically, we randomly selected N classes from ModelNet40 and sampled K objects from each class. Each experiment was

TABLE V

ABLATION EXPERIMENTS WITH POINT-UMAE PERTAINING ON THE SHAPENET [22] DATASET. THE FINE-TUNED ACCURACY (%) ACHIEVED IS REPORTED. DEFAULT SETTINGS ARE MARKED IN GRAY .

(a) Encoder depth

Blocks	Acc (%)
1	93.79
2	94.24
3	94.21
4	**94.25**

(b) Decoder depth

Blocks	Acc (%)
1	93.89
2	**94.24**
3	94.22
4	94.17

(c) Block Mask

Ratio	Loss	Acc (%)
0.40	1.73	93.53
0.60	1.94	**93.64**
0.70	2.07	93.21
0.80	2.15	92.79

(d) Random Mask

Ratio	Loss	Acc (%)
0.60	1.43	93.72
0.70	1.51	94.13
0.75	1.53	**94.24**
0.80	1.67	94.19

(e) Number of Scales

Scales	Acc (%)
1	93.68
2	93.94
3	**94.24**
4	94.23

(f) Pre-training losses

L2 CD	L1 CD	EMD	Acc (%)
✓	-	-	**94.24**
-	✓	-	93.71
-	-	✓	93.74
✓	-	✓	94.06

(g) Token Dimension

1-st Scale	2-nd Scale	3-rd Scale	Acc (%)
384	384	384	93.82
384	576	768	**94.24**
256	512	1024	93.95
384	768	1536	94.21

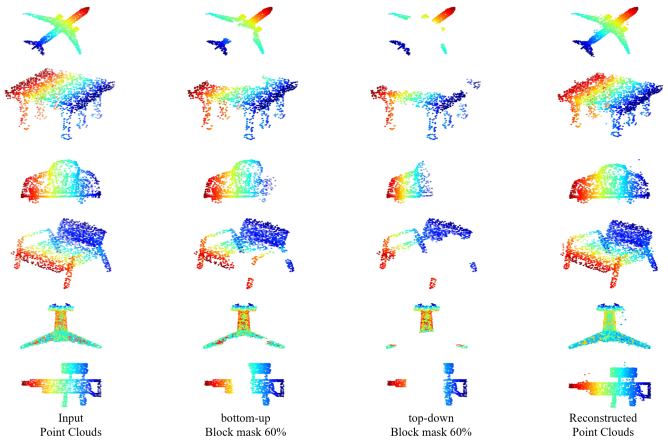

Fig. 2. **Visualization of reconstruction.** We compared the differences between the top-down and the bottom-up masking strategy at the lowest scale.

repeated 10 times, and the average and standard deviation of the results were calculated. As shown in Table II, we achieved state-of-the-art (SOTA) performance in all four settings.

D. Part Segmentation

We evaluated the part segmentation performance of Point-UMAE on the ShapeNetPart [23] dataset. Table III presents the mean Intersection over Union for all classes (mIoU_C) and all instances (mIoU_I). Compared to the fully supervised method DGCNN [19], Point-UMAE achieved significant improvements of 2.23% and 0.93% in these two key evaluation metrics, respectively. In the experimental results, our method achieved an mIoU_C of 84.65% and an mIoU_I of 86.13%, outperforming other point cloud self-supervised learning methods.

E. 3D Object Detection

To evaluate the performance of the model on the challenging task of 3D object detection, we conducted experiments on the ScanNetV2 [25] dataset. In this experiment, we used the same encoder architecture as 3DETR [30] at various scales and retrained the model specifically for the ScanNetV2 dataset. The experimental results, as shown in Table IV, demonstrate a significant improvement over the baseline algorithm 3DETR [30], with a 4% increase in AP_{25} and a 10.5% performance gain in AP_{50}. Additionally, our method outperformed Point-BERT [8] and Mask-Point [9], which do not have the ability to extract multi-scale features.

F. Visualization

We visually compared the top-down masking strategy employed by Point-UMAE with the bottom-up masking strategy used by Point-M2AE [10]. As shown in Figure 2, the bottom-up masking strategy used by Point-M2AE exhibits significant point cloud position leakage, as this strategy can only guarantee the predetermined masking ratio at the highest scale. In contrast, our top-down masking strategy demonstrates good consistency across all scales. Additionally, we show that under a 60% masking ratio condition, Point-UMAE achieves high-quality point cloud reconstruction.

G. Ablation Study

To determine the optimal parameter settings for our method, we conducted a comparative analysis of different configurations, as shown in Table V. These configurations include various aspects such as transformer depth, masking strategies, number of scales, loss functions, and feature dimensions. The default settings for our method are highlighted in gray in the table. Notably, an encoder depth of 4 yielded the best performance; however, the performance gain compared to a 2-block configuration was minimal, while the number of parameters nearly doubled. We confirmed that a random masking strategy, compared to block masking, is more effective at preserving global information integrity at high masking ratios, thereby enhancing the model's ability to capture and understand global features during pre-training. Additionally, L2 Chamfer Distance (CD) loss proved to be more effective in capturing subtle differences between point clouds. Finally, the pyramid structure demonstrated a significant advantage in multi-scale feature encoding.

IV. CONCLUSION

In this paper, we introduce Point-UMAE, an innovative framework for point cloud processing. This method adopts a top-down masking strategy, ensuring consistency in multi-scale masking, and leverages the advantages of a multi-scale architecture by successfully transferring the Unet-like encoder architecture in both pre-training and fine-tuning stages. Point-UMAE demonstrates competitiveness in various downstream tasks, including but not limited to 3D shape classification, part segmentation, and object detection. Looking ahead, we plan to further enhance the performance of Point-UMAE by expanding the scale of training data and exploring its application in a wider range of point cloud-based perception tasks.

REFERENCES

[1] Y. Yang, C. Feng, Y. Shen, and D. Tian, "Foldingnet: Point cloud auto-encoder via deep grid deformation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

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

[3] J. Sauder and B. Sievers, "Self-supervised deep learning on point clouds by reconstructing space," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[4] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, and H. Hu, "Simmim: A simple framework for masked image modeling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 9653–9663.

[5] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A general framework for self-supervised learning in speech, vision and language," in *International Conference on Machine Learning*. PMLR, 2022, pp. 1298–1312.

[6] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 16 000–16 009.

[7] Y. Pang, W. Wang, F. E. Tay, W. Liu, Y. Tian, and L. Yuan, "Masked autoencoders for point cloud self-supervised learning," in *European conference on computer vision*. Springer, 2022, pp. 604–621.

[8] X. Yu, L. Tang, Y. Rao, T. Huang, J. Zhou, and J. Lu, "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 (CVPR)*, June 2022, pp. 19 313–19 322.

[9] H. Liu, M. Cai, and Y. J. Lee, "Masked discrimination for self-supervised learning on point clouds," in *European Conference on Computer Vision*. Springer, 2022, pp. 657–675.

[10] R. Zhang, Z. Guo, P. Gao, R. Fang, B. Zhao, D. Wang, Y. Qiao, and H. Li, "Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training," *Advances in neural information processing systems*, vol. 35, pp. 27 061–27 074, 2022.

[11] Z. Guo, R. Zhang, L. Qiu, X. Li, and P.-A. Heng, "Joint-mae: 2d-3d joint masked autoencoders for 3d point cloud pre-training," *arXiv preprint arXiv:2302.14007*, 2023.

[12] H. Zeng, P. Zhang, F. Li, J. Wang, T. Ye, and P. Guo, "Masked generative extractor for synergistic representation and 3d generation of point clouds," *arXiv preprint arXiv:2406.17342*, 2024.

[13] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015,Lecture Notes in Computer Science*. Springer, 2015, pp. 234–241.

[14] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang, "Swin-unet: Unet-like pure transformer for medical image segmentation," in *European conference on computer vision*. Springer, 2022, pp. 205–218.

[15] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[16] M. A. Uy, Q.-H. Pham, B.-S. Hua, T. Nguyen, and S.-K. Yeung, "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 (ICCV)*, October 2019.

[17] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

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

[19] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[20] H. Wang, Q. Liu, X. Yue, J. Lasenby, and M. J. Kusner, "Unsupervised point cloud pre-training via occlusion completion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 9782–9792.

[21] H. Fan, H. Su, and L. J. Guibas, "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 (CVPR)*, July 2017.

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

[23] L. Yi, V. G. Kim, D. Ceylan, I.-C. Shen, M. Yan, H. Su, C. Lu, Q. Huang, A. Sheffer, and L. Guibas, "A scalable active framework for region annotation in 3d shape collections," *ACM Transactions on Graphics (ToG)*, vol. 35, no. 6, pp. 1–12, 2016.

[24] Z. Li, Z. Gao, C. Tan, B. Ren, L. T. Yang, and S. Z. Li, "General point model pretraining with autoencoding and autoregressive," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 954–20 964.

[25] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Niessner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[26] Z. Ding, X. Han, and M. Niethammer, "Votenet: A deep learning label fusion method for multi-atlas segmentation," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part III 22*. Springer, 2019, pp. 202–210.

[27] S. Huang, Y. Xie, S.-C. Zhu, and Y. Zhu, "Spatio-temporal self-supervised representation learning for 3d point clouds," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 6535–6545.

[28] S. Xie, J. Gu, D. Guo, C. R. Qi, L. Guibas, and O. Litany, "Pointcontrast: Unsupervised pre-training for 3d point cloud understanding," in *Computer Vision – ECCV 2020,Lecture Notes in Computer Science*. Springer, 2020, pp. 574–591.

[29] Z. Zhang, R. Girdhar, A. Joulin, and I. Misra, "Self-supervised pretraining of 3d features on any point-cloud," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 10 252–10 263.

[30] I. Misra, R. Girdhar, and A. Joulin, "An end-to-end transformer model for 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 2906–2917.