SAM-Guided Masked Token Prediction for 3D Scene Understanding

Zhimin Chen[1]
Clemson University

Liang Yang[2]
The City University of New York

Yingwei Li[3]
Johns Hopkins University

Longlong Jing[2]
The City University of New York

Bing Li[✉1]
Clemson University

Abstract

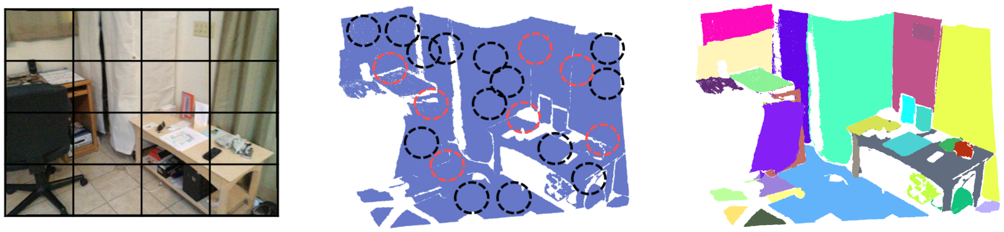
Foundation models have significantly enhanced 2D task performance, and recent works like Bridge3D have successfully applied these models to improve 3D scene understanding through knowledge distillation, marking considerable advancements. Nonetheless, challenges such as the misalignment between 2D and 3D representations and the persistent long-tail distribution in 3D datasets still restrict the effectiveness of knowledge distillation from 2D to 3D using foundation models. To tackle these issues, we introduce a novel SAM-guided tokenization method that seamlessly aligns 3D transformer structures with region-level knowledge distillation, replacing the traditional KNN-based tokenization techniques. Additionally, we implement a group-balanced re-weighting strategy to effectively address the long-tail problem in knowledge distillation. Furthermore, inspired by the recent success of masked feature prediction, our framework incorporates a two-stage masked token prediction process in which the student model predicts both the global embeddings and the token-wise local embeddings derived from the teacher models trained in the first stage. Our methodology has been validated across multiple datasets, including SUN RGB-D, ScanNet, and S3DIS, for tasks like 3D object detection and semantic segmentation. The results demonstrate significant improvements over current State-of-the-art self-supervised methods, establishing new benchmarks in this field.

1 Introduction

3D computer vision plays a critical role in domains such as autonomous driving and robotics. Despite its importance, this field faces significant challenges in acquiring and annotating 3D data due to the high costs and complex technical requirements involved. These challenges have led to a notable scarcity of large-scale annotated datasets. To address these issues, there has been a growing shift towards self-supervised learning (SSL) strategies, including contrastive learning and masked autoencoders (MAE), which aim to improve the learning efficiency of networks and reduce reliance on labeled data. Recently, the success of 2D foundation models like Contrastive Vision-Language Pre-training (CLIP) [41] and Segment Anything (SAM) [29] has led to significant progress in image understanding. However, large-scale 3D foundation models have not yet been proposed, primarily due to the scarcity of 3D datasets. Therefore, leveraging these powerful 2D foundation models for 3D scene understanding via self-supervised learning remains an open question.

Recent works, such as CLIP2Scene [8], Seal [32], and Bridge3D [10], have made significant progress in enhancing 3D scene understanding through the use of foundation models. CLIP2Scene effectively integrates CLIP with 3D models by implementing pixel-to-point distillation via foundation models.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



(a) Patch based 2D tokenization method.

(b) KNN-based 3D tokenization method.

(c) Proposed SAM-guided 3D tokenization method.

Figure 1: **The comparison of tokenization methods.** In Section 3.2, we present a detailed comparison of our proposed tokenization method to the previous KNN-based approach. *As shown in the red circle, the KNN-based method may inadvertently group points from different SAM regions into the same tokens, leading to potential confusion within the 3D network.* In contrast, our method effectively employs SAM masks in tokenization to ensure seamless region-level knowledge distillation, thereby avoiding these issues.

Seal distills the knowledge from 2D foundation models into the 3D network for semantic segmentation. Bridge3D introduces an innovative pre-training strategy for 3D models by utilizing features, semantic masks, and captions derived from foundation models. However, significant challenges still remain in leveraging foundation models for 3D scene understanding. Specifically, while CLIP2Scene employs point-to-text contrastive learning, it does not utilize critical region-specific information, which is essential for dense representation learning. Seal leveraged the 3D U-Net as backbone which struggles with scalability and flexibility, making it less effective for scaling and handling tasks such as detection. Bridge3D attempts to address this by using SAM-generated masks to distill vision and language representations to point tokens at the regional level. However, as illustrated in Figure 1, both Bridge3D and earlier 3D transformer-based methods employ KNN-based point tokenization strategies that can result in information conflicts during SAM-guided region-level knowledge distillation. This conflict arises when points from different SAM regions are grouped into the same 3D tokens, thereby confusing the 3D network. Furthermore, both CLIP2Scene and Bridge3D do not take into account the inherently long-tail property of 3D datasets. Giving equal weight to all samples causes the model to be predominantly driven by gradients from a few over-represented samples, leading to poor performance on under-represented samples.

To overcome these challenges, we propose a SAM-guided masked token prediction method that facilitates region-level 2D-to-3D knowledge distillation using foundation models. Unlike traditional 3D transformer methods that rely on KNN for tokenization, our approach employs masks obtained from SAM to tokenize points. This strategy effectively prevents conflicts between different regions and points, ensuring a seamless integration of point tokenization with region-level knowledge distillation. Additionally, SAM masks improve the representation of homogeneous neighboring points by more effectively leveraging boundary regularities compared to KNN-based methods. Furthermore, to address the issue of representation imbalance, we introduce a group-balanced re-weighting strategy that adjusts the distillation loss weights between 2D and 3D representations at the region level. In the self-supervised phase, where labels are absent, we utilize well-trained 2D foundation models to cluster region-level 2D representations using K-means, assigning pseudo-labels based on their cluster index. During training, we enhance the weights for tail groups while reducing them for head groups. Inspired by the recent success of masked feature prediction [56] in cross-modality learning, we introduce a two-stage masked token prediction framework. In the first stage, we perform dense region-level knowledge distillation using the SAM-guided tokenization method to transfer well-learned knowledge from the foundation model to the 3D network. In the second stage, we have the student model predict both the instance-level global embedding and the token-wise local embeddings obtained from the teacher model in the first stage based on visible 3D input patches. This approach ensures that the student model learns well-aligned and contextualized local and global representations, thereby improving its performance on downstream tasks.

We validated our methodology across multiple datasets and tasks, including SUN RGB-D [46] and ScanNet [14] for 3D object detection, and S3DIS [4] and ScanNet [14] for 3D semantic segmentation.

Our approach outperforms current state-of-the-art self-supervised learning methods, underlining the effectiveness of our proposed framework.

The key contributions of our work are summarized as follows:

- We introduce a novel two-stage SAM-guided masked token prediction framework that leverages foundation models for 3D scene understanding.
- We present a group-balanced re-weighting method for long-tail representation distillation and a SAM-guided tokenization method to seamlessly align 2D and 3D region-level features.
- Extensive experiments have been conducted to demonstrate the significance of our approach in various 3D downstream tasks.

2 Related Work

3D Self-supervised Pre-training. The field of self-supervised learning for point clouds has witnessed substantial advancements, with researchers exploring a variety of pre-training strategies to enhance the transferability and initialization quality of networks for downstream tasks [1, 21, 11, 57, 28, 19]. These strategies range from learning the relative positions of points [43] to deploying multiple pretext tasks [22] and employing contrastive learning approaches [17, 27, 42, 49, 2, 28, 20, 18]. Innovatively, Info3D [42] applies InfoMax principles and contrastive learning to 3D shapes, enhancing feature extraction from complex geometries. PointContrast [49] performs point-level contrastive learning across transformed views of a single point cloud, promoting robustness to spatial alterations. Meanwhile, the work by Zhang [55] contrasts instance-level representations derived from different architectural processes within identical scenarios. Additionally, CrossPoint [2] pioneers a multi-modal contrastive framework that establishes 3D-2D correspondences, capitalizing on the complementary attributes of point clouds and images.

Masked Autoencoder To enhance masked image modeling, the Masked Autoencoder [23] (MAE) was initially introduced for 2D images, utilizing an asymmetric encoder-decoder transformer architecture [16, 6]. This process starts with an encoder that receives a randomly masked image to extract high-level latent representations. A lightweight decoder then processes these representations, reconstructing the raw RGB pixels of the masked patches. Demonstrating exceptional performance across various downstream tasks, MAE has inspired a range of innovative adaptations. Expanding to 3D data, some adaptation methods [37, 12, 53] have been proposed to apply MAE-style pre-training to 3D point clouds. These methods involve sampling visible point tokens for the encoder and reconstructing masked 3D coordinates with the decoder. However, these 3D MAE applications have predominantly focused on masked point reconstruction. Recent studies [5, 56] have shown that masked feature prediction can be a more effective strategy for representation learning. Building on this insight, our work introduces a two-stage framework specifically designed to enable masked token prediction in 3D scene understanding, aiming to enhance the learning efficiency and applicability of MAE in complex 3D environments. This novel approach promises to push the boundaries of how deep learning models perceive and interpret three-dimensional data.

Multi-modality Learning for 3D Scene Understanding Numerous studies have explored knowledge transfer from pre-trained 2D foundation models to 3D representations at the object level [24, 25, 54, 52, 38]. For comprehensive scene understanding, SlidR [44] initially employs the super-pixel technique to define regions and subsequently utilizes the InfoNCE loss for region-level contrastive learning between point cloud and 2D representations. The CLIP2Scene approach [8] leverages the MaskClip model [59] to generate dense semantic predictions. However, it lacks the capability to produce instance-specific semantic results, which is crucial for distilling object-level visual representations into 3D models. Bridge3D [10] proposes an innovative pre-training strategy for 3D models using features, semantic masks, and captions derived from foundation models. It integrates region-level knowledge distillation with a masked autoencoder for 3D scene understanding, achieving state-of-the-art performance. Despite these advancements, the KNN-based tokenization method employed in Bridge3D and other existing 3D transformer-based methods faces challenges. Specifically, the mismatch between KNN grouping and predefined mask regions can lead to representational conflicts, ultimately degrading performance. To address these limitations, we innovatively propose a SAM-guided tokenization method that seamlessly integrates region-level distillation with a

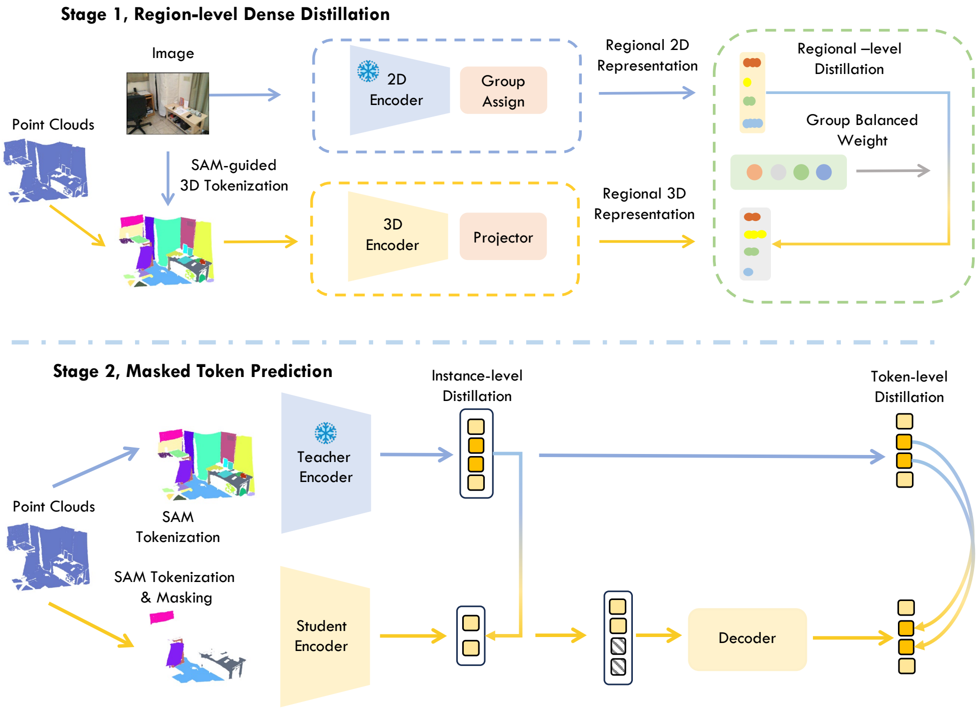

Stage 1, Region-level Dense Distillation
Image
Point Clouds
2D Encoder
Group Assign
Regional 2D Representation
Regional –level Distillation
SAM-guided 3D Tokenization
Group Balanced Weight
3D Encoder
Projector
Regional 3D Representation
Stage 2, Masked Token Prediction
Instance-level Distillation
Token-level Distillation
Teacher Encoder
Point Clouds
SAM Tokenization
SAM Tokenization & Masking
Student Encoder
Decoder

Figure 2: **Overall framework of the proposed method.** Our method introduces a two-stage masked token prediction framework for learning from foundation models. In the first stage, we input complete point clouds and leverage SAM masks to guide the point cloud tokenization, thereby seamlessly aligning the 2D and 3D region-level features for dense prediction. A group-balanced weight is applied during distillation to prevent bias towards the head representations. In the second stage, we freeze the models trained in the first stage and have the student models predict instance-level features and masked tokens obtained from the teacher models.

plain transformer architecture. This method ensures coherent region-level knowledge transfer and enhances the overall efficacy of the learning process in 3D scene understanding.

3 Methodology

In this section, we outline our methodology, starting with how we utilize foundation model SAM [29] to obtain masks offline. We then describe our approach to tokenize point clouds using SAM instead of KNN to address the misalignment between 2D and 3D representations. Next, we introduce the group-balanced re-weighting strategy to address the long-tail representation issue in knowledge distillation. Finally, we detail our two-stage masked token framework, which facilitates the model in learning well-aligned and contextualized representations through a process of two-stage masked token prediction.

3.1 Mask Generation

To enable region-level distillation, we utilized the foundation model SAM [29] to generate masks within visual images. SAM-generated masks provide comprehensive coverage of both the object and its surrounding context. This integration enables us to obtain a cohesive set of segmentation masks $\mathcal{O}_1, \ldots, \mathcal{O}_N$. To establish a precise correspondence between mask-level visual features and point tokens $\{x_i, p_i\}$, we align the point cloud tokens with the respective SAM masks, where x_i and p_i represent paired image and point features, respectively. This process is conducted offline, and the resulting labels are stored locally for easy access during the self-supervised training phase.

3.2 SAM-guided Point Tokenization

Recent state-of-the-art method Bridge3D [10] adapts the 3D plain transformer architecture for knowledge distillation from foundation models. Like other 3D transformer-based methods, Bridge3D directly tokenizes point clouds using farthest point sampling (FPS) and K-nearest neighbors (KNN) algorithms. Specifically, given a point cloud $X^i \in \mathbb{R}^{N\times 3}$ with N points, FPS is used to select n centroid points (CT) for forming point patches. Subsequently, KNN determines the k nearest points to each centroid, forming the corresponding token P.

To enable efficient distillation from 2D to 3D using foundation models, Bridge3D proposes a region-level distillation strategy. However, we find that this strategy cannot effectively align 2D and 3D information when using KNN-based point tokenization. As depicted in Figure 1, KNN-based point tokenization strategies can group points from different SAM regions into the same 3D tokens. This misalignment causes information conflict, which confuses the 3D network and degrades the distillation performance.

To address this challenge, we introduce a novel SAM-guided point patch generation method tailored for multi-modality region-level knowledge distillation. We start by projecting the 3D point cloud onto corresponding 2D images. Then, we assign points to tokens based on their positions within the SAM-defined regions in the 2D images. Each patch's centroid is calculated as the average position of all points within that patch. Features are then extracted using PointNet, ensuring that each token's representation is both cohesive and regionally consistent. This methodology not only enhances the alignment between 3D points and their corresponding 2D regions but also significantly improves the performance of our knowledge distillation framework by leveraging boundary regularities provided by SAM.

3.3 Dense Feature Distillation

To enhance the distillation of rich representations from 2D to 3D, we build on the methodologies proposed in Bridge3D [10], utilizing region proposals from the SAM vision foundation model. Our method primarily differs from Bridge3D in our approach to aligning 2D and 3D representations. Unlike Bridge3D, which employs the traditional KNN method to generate point tokens, often resulting in the aggregation of points from disparate regions as illustrated in Figure 1, our method uses the SAM model to guide point cloud tokenization. This ensures a one-to-one correspondence between point tokens and region-level 2D features, avoiding the representation conflicts that impair 3D understanding in the Bridge3D approach. This targeted tokenization results in a comprehensive set of segmentation masks $\mathcal{O}_1, \ldots, \mathcal{O}_N$, each enriched with its corresponding textual narrative, thereby providing a deeper contextual dataset.

For detailed feature extraction, we define E^{θ}_{3D} as the trainable 3D network and E^{θ}_{2D} as the frozen pre-trained 2D encoder. The 3D network processes the point cloud, while the 2D encoder manages the corresponding images. These components generate 3D point token features $H \in \mathbb{R}^{M\times L}$ and 2D image pixel features $I_j \in \mathbb{R}^{h\times w\times L}$, where M represents the number of regions, equivalent to the segmentation masks produced by SAM, and L denotes the feature dimension; h and w are the height and width of the image features, respectively. We project the point token features to 2D space via a projection layer to obtain projected 3D features F_{3D}. We then pool pixel representations within the same SAM-generated region to derive region-level 2D representations $F_{2D} \in \mathbb{R}^{M\times L}$. Simultaneously, we establish correspondences between each 3D token and its matching 2D regional representation $(H_i, F_i)_{i=1}^{M}$, where H_i and F_i are the paired 3D and 2D regional features, ensuring a direct and meaningful alignment between the modalities. This setup allows for robust region-level feature-dense distillation, effectively training our model to better understand and interpret complex 3D scenes. The distillation process is formulated as follows:

$$F_{2D,i} = \frac{1}{\mathcal{O}_i} \sum_{i\in\mathcal{O}_j} (I_j) \tag{1}$$

$$F_{3D,i} = Proj(H_j) \tag{2}$$

Where H_i represents point tokens, $Proj$ is the projection layer. The objective function is as follows:

$$\mathcal{L}_{distill} = \frac{1}{M}\sum_{i}^{M} L_1(F_{2D,i}, F_{3D,i}) \tag{3}$$

The L_1 is the smooth L_1 loss.

3.4 Group Balanced Re-weighting

Training models on 3D datasets, which are inherently highly imbalanced, presents significant challenges. Directly applying approaches that are not specifically designed for imbalanced datasets often results in sub-optimal network performance and thus fails to deliver satisfactory outcomes. To address this issue, recent studies, as noted by Cui et al. [13], Yu et al. [50], and Alshammari et al. [3], have introduced class-balanced loss strategies. These strategies involve recalibrating the weights in the loss function to focus more on underrepresented (tail) classes and reduce the emphasis on overrepresented (head) classes. Such adjustments aim to establish a more equitable training environment, enhancing fairness and boosting the robustness of the model.

In our 2D to 3D pre-training tasks, the lack of explicit labels complicates the identification of head and tail representations within the data. To tackle this challenge of imbalance in the representation learning stage, we propose a prototype-level re-weighting method. Leveraging the discriminative features provided by foundation models, we can directly cluster these features and use their cluster indices as pseudo-labels. Specifically, we utilize foundation models such as DINOv2 [36] or CLIP [41] to extract visual features, which are then resized to the original image dimensions using interpolation. Next, we apply max pooling across features within regions defined by SAM-generated masks to obtain region-level features. These features are grouped into K clusters using KNN, categorizing them into distinct groups. Each region-level feature is assigned a group index, which we treat as a pseudo-label for applying a class-level reweighted loss. We then count the number of regions in group i as n_i. This innovative approach allows us to effectively address the long-tail problem in representation learning by balancing the influence of each group during the learning process. $k_i = \frac{n_i - n_{\min}}{n_{\max}}$, Where $\tau_i = 1.0 - k_i$ and $w_i = \frac{\tau_i}{\sum_{j=0}^{K}\tau_i}$. Hence, the dense distillation loss for the first stage is:

$$\mathcal{L}_{distill} = \frac{1}{M}\sum_{i}^{M} w_i L_1(F_{2D,i}, F_{3D,i}) \tag{4}$$

3.5 Maksed Token Prediction

As illustrated by VideoPrism [56], latent space reconstruction is an effective method for cross-modality knowledge distillation. Inspired by this, we propose a two-stage framework to integrate latent space prediction within the MAE framework for 2D to 3D knowledge distillation. This approach differs from previous 3D masked autoencoder methods like Point-MAE [37] and Bridge3D [10], which reconstruct raw, masked inputs as their targets.

As depicted in Fig. 2, our approach involves a two-stage process. In the first stage, we perform dense feature knowledge distillation from 2D to 3D using foundation models with the proposed SAM-guided tokenization method. In the second stage, we implement a teacher-student framework. The model from the first stage serves as the teacher and is frozen for the second stage. During training, all tokens are processed by the teacher model to generate token features, and the student model is tasked with reconstructing these detailed 3D token representations using only the visible parts of the data. We introduce an instance-level distillation loss to guide the student model's learning, pushing the limits of self-supervised learning in comprehending 3D spaces.

Specifically, we send complete point tokens to the teacher models and masked point tokens to the student models. For the instance-level knowledge prediction, we pool all point token features after the teacher encoder as $F_{ins}^{teacher}$ and after the student encoder as $F_{ins}^{student}$. The student model then predicts $F_{ins}^{teacher}$ using MLP layers. The instance prediction is formulated as follows:

$$\mathcal{L}_{ins} = MSE(MLP(F_{ins}^{student}), F_{ins}^{teacher}) \tag{5}$$

		SUN RGB-D		ScanNetV2	
Methods	Pre-trained	AP_{25}	AP_{50}	AP_{25}	AP_{50}
VoteNet [39]	*None*	57.7	32.9	58.6	33.5
PointContrast [49]	✓	57.5	34.5	59.2	38.0
Hou et al. [26]	✓	-	36.4	-	39.3
4DContrast [9]	✓	-	38.2	-	40.0
DepthContrast [55]	✓	61.6	35.5	64.0	42.9
DPCo [30]	✓	60.2	35.5	64.2	41.5
3DETR [35]	*None*	58.0	30.3	62.1	37.9
+Plain Transformer	*None*	57.6	31.9	61.1	38.6
+Point-BERT[51]	-	-	-	61.0	38.3
+Point-MAE [37]	✓	-	-	63.4	40.6
+MaskPoint [31]	✓	-	-	63.4	40.6
+ACT [15]	✓	-	-	63.5	41.0
+PiMAE [7]	✓	59.9	33.7	63.0	40.2
+Bridge3D [10]	✓	61.8	37.1	65.3	44.2
+Ours	✓	**63.5(+1.7)**	**39.5(+2.4)**	**68.2 (+2.9)**	**48.4(+4.2)**
GroupFree3D [33]	*None*	63.0	45.2	67.3	48.9
+Plain Transformer	*None*	62.2	45.0	66.1	48.3
+Point-MAE [37]	✓	63.9	46.1	67.4	49.8
+PiMAE [7]	✓	65.0	46.8	67.9	50.5
+Bridge3D [10]	✓	67.9	48.5	69.1	51.9
+Ours	✓	**68.9(+1.0)**	**52.1(+3.6)**	**72.3(+3.2)**	**55.7(+3.8)**

Table 1: **3D object detection results on ScanNet and SUN RGB-D dataset.** We adopt the average precision with 3D IoU thresholds of 0.25 (AP_{25}) and 0.5 (AP_{50}) for the evaluation metrics.

We use the global features of the student model with only visible inputs to predict the global features of the teacher model with complete inputs. Additionally, we employ a token-level prediction loss to ensure that the student models can predict the masked tokens obtained from the teacher model's decoder.

$$\mathcal{L}_{token} = \frac{1}{N_m} \sum_{i=1}^{N_m} MSE(F_i^{student}, F_i^{teacher}). \tag{6}$$

Where N_m is the number of masked tokens. The effectiveness of this learning setup is evaluated through a defined reconstruction loss, ensuring high precision in the alignment between the student's and teacher's outputs. Notably, we continue to employ the SAM-guided tokenization method to facilitate this process. The final loss for the second stage is formulated as:

$$\mathcal{L}_{final} = \mathcal{L}_{ins} + \mathcal{L}_{token} \tag{7}$$

4 Experiments

This section begins with an overview of the pre-training and fine-tuning configurations for our method. Subsequently, we demonstrate the method's effectiveness through its application to several prominent downstream tasks, such as 3D object detection and 3D semantic segmentation. Finally, we present comprehensive ablation studies to validate the impact and contribution of each component within our approach.

4.1 Self-supervised Pre-training and Fine-tuning

Pre-training. In our pre-training stage, we leverage the ScanNet dataset [14], aligning with approaches adopted in prior research [39, 55] to obtain the corresponding image and point cloud pairs. ScanNet is an indoor dataset that includes approximately 1,500 scans derived from 2.5 million RGB-D frames. We follow the official protocol for training/validation splits, extracting 78,000 frames from the training subset by sampling one frame every 25 frames to construct our dataset. For the

		S3DIS		ScanNetV2	
Methods	Pre-trained	$mIoU$	$mAcc$	$mIoU$	$mAcc$
SR-UNet [49]	*None*	68.2	75.5	72.1	80.7
PointContrast [49]	✓	70.9	77.0	74.1	81.6
DepthContrast [55]	✓	70.6	-	73.1	-
Hou et al. [26]	✓	72.2	-	73.8	-
Standard Transformer [51]	*None*	60.0	68.6	-	-
PointBert [51]	✓	60.8	69.9	-	-
PViT [40]	*None*	64.4	69.9	-	-
PViT+Pix4Point [40]	✓	69.6	75.2	-	-
Plain Transformer	*None*	61.1	67.2	67.3	73.1
+Point-MAE [37]	✓	64.8	70.2	-	-
+Bridge3D [10]	✓	70.2	76.1	73.9	80.2
+Ours	✓	**71.8 (+1.6)**	**78.2(+2.1)**	**75.4(+1.5)**	**81.5(+1.3)**

Table 2: **3D semantic segmentation results on S3DIS and ScanNet dataset.** We adopt the mean accuracy (mAcc) and mean IoU (mIoU) for the evaluation metrics.

optimization process, we utilize the AdamW optimizer [34] throughout both stages of our training, starting with a base learning rate of 0.001 and a weight decay set at 0.05. Our data is processed in batches of 64. During the second stage of training, we increase the masking ratio (r_w) to 60%. To further enhance the training dynamics, we implement a cosine learning rate scheduler coupled with a drop path rate of 0.1 and include a warm-up phase of 10 epochs to facilitate a smooth adjustment to the training conditions. For the 3D backbone encoder, we adopt the plain transformer structure used in Bridge3D [10]. On the image processing side, we employ the DINOV2 ViT-B model [36] to extract features. To adapt these features back to the original input size, we apply interpolation-based up-sampling techniques. The training is conducted using four A100 GPUs.

Fine-tuning. Following Bridge3D [10], we remove the decoders used in pre-training and introduce task-specific decoders for various downstream tasks. A key distinction between our fine-tuning approach and Bridge3D is the use of SAM-guided tokenization to generate tokens, rather than the traditional KNN-based tokenization methods employed by Bridge3D. Additionally, for detection tasks, we do not introduce new query embeddings. Instead, we use the tokens generated through SAM-guided tokenization as queries for self-attention. These tokens represent features of homogeneous neighboring regions defined by precise boundary regularities from SAM, making them suitable for 3D object detection tasks. Apart from these adjustments, we adhere to the same fine-tuning settings as Bridge3D for downstream tasks.

4.2 Results on Downstream Tasks

Object Detection. We demonstrate the generality of our proposed method by conducting pre-training on the indoor ScanNetV2 dataset [14] and subsequently fine-tuning it for object detection tasks in both the ScanNetV2 and SUN-RGBD [58] datasets. Building upon the baseline detection methods 3DETR [35] and GroupFree3D [33], our method significantly outperforms the previous state-of-the-art method Bridge3D. Specifically, our performance surpasses Bridge3D by 2.9 and 4.2 in AP$_{25}$ and AP$_{50}$ using the 3DETR baseline, and by 3.2 and 3.8 in AP$_{25}$ and AP$_{50}$ using the GroupFree3D baseline on the ScanNetV2 dataset. This consistent improvement over Bridge3D underscores the efficacy of our method in learning advanced 3D representations for object detection, indicating its potential for enhancing 3D scene understanding tasks.

Semantic Segmentation. In Table 2, we present the semantic segmentation results on the S3DIS [4] and ScanNet [14] datasets. Although Bridge3D [10] has improved the plain transformer baseline by a large margin, our method still outperforms Bridge3D by 1.6 and 1.5 $mIoU$ on the ScanNet and S3DIS datasets, respectively. It should be noted that Bridge3D utilizes foundation models with both 2D and text modalities, incorporating complex architectures. In contrast, our method, which utilizes only 2D foundation models, still outperforms Bridge3D in 3D semantic segmentation tasks. This demonstrates the efficiency of our proposed knowledge distillation strategy for enhancing 3D representation learning for semantic segmentation.

Dense	Masked Token	Balanced	SAM-Guided	ScanNetV2		S3DIS	
Distillation	Prediction	Re-weight	Tokenzie	AP_{25}	AP_{50}	$mIoU$	$mAcc$
				61.1	38.6	61.1	67.2
✓				62.4	41.7	66.2	71.3
✓	✓			64.5	44.3	68.7	74.1
✓	✓	✓		66.0	46.1	69.7	75.9
✓	✓		✓	67.1	47.0	70.9	77.0
✓	✓	✓	✓	**68.2**	**48.4**	**71.8**	**78.2**

Table 3: **The effectiveness of each component.** Ablation study on the effectiveness of each component on 3D object detection and semantic segmentation tasks.

	ScanNetV2		S3DIS	
	AP_{25}	AP_{50}	$mIoU$	$mAcc$
Stage 1	65.2	45.1	69.1	75.3
Stage 1 + MTP in same stage	66.0	46.3	69.9	76.1
Stage 1 + Stage 2 (Ours)	**68.2**	**48.4**	**71.8**	**78.2**

Table 4: **The effectiveness of Stage.** Ablation study on the effectiveness of a two-stage framework on 3D object detection and semantic segmentation tasks. MTP here represents the masked token prediction

4.3 Ablation Study

The Effectiveness of Each Component. As illustrated in Table 3, the results effectively demonstrate the advantages of each component incorporated into our comprehensive framework. The detailed ablation study reveals that incorporating dense distillation significantly enhances 3D representation learning and improves overall system performance. Additionally, the implementation of a two-stage masked token prediction enables student models to learn well-aligned and highly contextualized representations across different modalities, thereby further enhancing overall system performance. Moreover, the introduction of balanced re-weighting mechanisms significantly boosts network performance by effectively mitigating the long-tail distribution challenge, which is inherently problematic in 3D datasets. Finally, the integration of SAM-guided tokenization marks the most substantial improvement within our framework, as it seamlessly aligns 2D and 3D features, thus avoiding potential conflicts and discrepancies in information transfer. In conclusion, each component of our proposed method is designed to be complementary to the others; their combined application not only achieves optimal results but also markedly enhances performance across both 3D object detection and semantic segmentation tasks, demonstrating the robustness and efficacy of our approach.

The Effectiveness of Each Stage. Table 4 underscores the effectiveness of our proposed two-stage method, which initiates with the distillation of dense representations from 2D foundation models into 3D models during the first stage. This initial phase of our study is meticulously designed to assess whether it is imperative to employ a teacher model or if the strategy of predicting masked 2D features directly within the first stage could potentially achieve superior or equivalent performance. The results derived from this meticulous ablation study suggest that merely combining the initial stage with masked token prediction yields only modest improvements in overall performance. However, a significant enhancement is observed with the addition of the second stage, which more thoroughly integrates our structured teacher-student framework into the process. This marked improvement distinctly highlights the critical importance of the teacher-student design within our innovative approach, confirming that the detailed and layered integration of these essential elements is absolutely vital for obtaining well-learned, robust representations that are crucial for effective 3D scene understanding tasks.

4.4 Apply on SOTA 3D Detectors

We recognize that applying our method to state-of-the-art detection models can further demonstrate its generality and robustness. Therefore, we applied our approach to two leading 3D detection methods: CAGroup3D [47] and VDETR [45]. Due to the specifically designed BiResNet backbone

Methods	**Pre-trained**	**ScanNet** (AP_{25})	**ScanNet** (AP_{50})
CAGroup3D [4]	None	75.1	61.3
+ Ours	✓	**76.5**	**62.4**
VDETR [5]	None	73.6	60.1
+ Ours	✓	**75.8**	**63.0**

Table 5: 3D object detection results on ScanNet dataset based on CAGroup3D and VDETR.

Methods	**ScanNet** (AP_{25})	**ScanNet** (AP_{50})	**ScanNet** $(mIoU)$
Scratch	61.1	38.6	67.3
CLIP2Scene [8]	62.0	40.1	69.2
Seal [32]	62.7	41.3	70.3
PPT [48]	62.8	42.1	70.9
Ours	**68.2**	**48.4**	**75.4**

Table 6: Comparison with other pre-training methods with different backbones on ScanNet dataset in 3D detection and semantic segmentation tasks.

used by CAGroup3D, we were able to apply only our SAM-guided knowledge distillation and representation re-weighting techniques to it. For VDETR, which reports results using both a modified ResNet34 encoder and a plain transformer encoder, we replaced its encoder with our pre-trained encoder and compared the performance to the transformer backbone results reported in the original paper. The experimental results presented in Table. 5 show that our pre-training strategy enhances the performance of these state-of-the-art 3D detection models. Moreover, the performance improvement in VDETR, facilitated by our proposed SAM-guided tokenization and two-stage masked autoencoder, is greater than that observed in CAGroup3D, highlighting the effectiveness of our approach.

4.5 Results Comparison with Pre-Training Methods for Other Backbones

In the main paper, we did not compare our results with Seal [32], PPT [48], and CLIP2Scene [8] as they use 3D-UNet as the backbone and are exclusively fine-tuned for 3D semantic segmentation tasks. Most previous methods operate at the object-level or scene-level using transformer-based 3D models. To demonstrate the effectiveness of our approach specifically designed for transformers, we adapted the methodologies of Seal, PPT, and CLIP2Scene to the transformer structure, applying the same experimental settings as our method.

As shown in Table. 6, our method achieves the best performance, highlighting the advantages of our proposed strategies. In the revised version, we will cite these papers and include a discussion of their methodologies and results. We acknowledge that including comparisons with methods using different backbones could better illustrate the effectiveness of our approach, and therefore, we have undertaken this additional evaluation. The results clearly demonstrate that our approach outperforms these existing methods, emphasizing the robustness and generalizability of our pre-training strategy.

5 Conclusion

In conclusion, our study addresses the challenges of aligning 2D and 3D representations and enhances knowledge distillation in self-supervised learning for 3D scene understanding. We introduced a novel SAM-guided tokenization method that aligns 3D transformer structures with region-level insights, improving distillation effectiveness. Additionally, we propose a group-balanced re-weighting strategy to address the long-tail problem. Furthermore, we introduce a two-stage masked token prediction framework, enabling the student model to predict both global instance-level embeddings and local token-wise embeddings learned from the teacher model based on visible 3D input tokens. Experiments conducted on datasets such as SUN RGB-D, ScanNet, and S3DIS demonstrate the state-of-the-art performance of the proposed method in 3D object detection and semantic segmentation tasks. Our work is expected to have no negative societal implications.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. *arXiv preprint arXiv:1707.02392*, 2017.

[2] Mohamed Afham, Isuru Dissanayake, Dinithi Dissanayake, Amaya Dharmasiri, Kanchana Thilakarathna, and Ranga Rodrigo. Crosspoint: Self-supervised cross-modal contrastive learning for 3d point cloud understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9902–9912, 2022.

[3] Shaden Alshammari, Yu-Xiong Wang, Deva Ramanan, and Shu Kong. Long-tailed recognition via weight balancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6897–6907, 2022.

[4] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016.

[5] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023.

[6] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[7] Anthony Chen, Kevin Zhang, Renrui Zhang, Zihan Wang, Yuheng Lu, Yandong Guo, and Shanghang Zhang. Pimae: Point cloud and image interactive masked autoencoders for 3d object detection. *arXiv preprint arXiv:2303.08129*, 2023.

[8] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by clip. *arXiv preprint arXiv:2301.04926*, 2023.

[9] Yujin Chen, Matthias Nießner, and Angela Dai. 4dcontrast: Contrastive learning with dynamic correspondences for 3d scene understanding. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, pages 543–560. Springer, 2022.

[10] Zhimin Chen, Longlong Jing, Yingwei Li, and Bing Li. Bridging the domain gap: Self-supervised 3d scene understanding with foundation models. *Advances in Neural Information Processing Systems*, 36, 2024.

[11] Zhimin Chen, Longlong Jing, Liang Yang, Yingwei Li, and Bing Li. Class-level confidence based 3d semi-supervised learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 633–642, 2023.

[12] Zhimin Chen, Yingwei Li, Longlong Jing, Liang Yang, and Bing Li. Point cloud self-supervised learning via 3d to multi-view masked autoencoder. *arXiv preprint arXiv:2311.10887*, 2023.

[13] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

[14] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[15] Runpei Dong, Zekun Qi, Linfeng Zhang, Junbo Zhang, Jianjian Sun, Zheng Ge, Li Yi, and Kaisheng Ma. Autoencoders as cross-modal teachers: Can pretrained 2d image transformers help 3d representation learning? *arXiv preprint arXiv:2212.08320*, 2022.

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[17] Bi'an Du, Xiang Gao, Wei Hu, and Xin Li. Self-contrastive learning with hard negative sampling for self-supervised point cloud learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 3133–3142, 2021.

[18] Ziyue Feng, Longlong Jing, Peng Yin, Yingli Tian, and Bing Li. Advancing self-supervised monocular depth learning with sparse lidar. In *Conference on Robot Learning*, pages 685–694. PMLR, 2022.

[19] Ziyue Feng, Liang Yang, Pengsheng Guo, and Bing Li. Cvrecon: Rethinking 3d geometric feature learning for neural reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17750–17760, 2023.

[20] Ziyue Feng, Liang Yang, Longlong Jing, Haiyan Wang, YingLi Tian, and Bing Li. Disentangling object motion and occlusion for unsupervised multi-frame monocular depth. In *European Conference on Computer Vision*, pages 228–244. Springer, 2022.

[21] Matheus Gadelha, Rui Wang, and Subhransu Maji. Multiresolution tree networks for 3d point cloud processing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 103–118, 2018.

[22] Kaveh Hassani and Mike Haley. Unsupervised multi-task feature learning on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8160–8171, 2019.

[23] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022.

[24] Deepti Hegde, Jeya Maria Jose Valanarasu, and Vishal M Patel. Clip goes 3d: Leveraging prompt tuning for language grounded 3d recognition. *arXiv preprint arXiv:2303.11313*, 2023.

[25] Georg Hess, Adam Tonderski, Christoffer Petersson, Lennart Svensson, and Kalle Åström. Lidarclip or: How i learned to talk to point clouds. *arXiv preprint arXiv:2212.06858*, 2022.

[26] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15587–15597, 2021.

[27] Siyuan Huang, Yichen Xie, Song-Chun Zhu, and Yixin Zhu. Spatio-temporal self-supervised representation learning for 3d point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6535–6545, 2021.

[28] Longlong Jing, Yucheng Chen, Ling Zhang, Mingyi He, and Yingli Tian. Self-supervised modal and view invariant feature learning. *arXiv preprint arXiv:2005.14169*, 2020.

[29] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

[30] Lanxiao Li and Michael Heizmann. A closer look at invariances in self-supervised pre-training for 3d vision. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXX*, pages 656–673. Springer, 2022.

[31] Haotian Liu, Mu Cai, and Yong Jae Lee. 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*, pages 657–675. Springer, 2022.

[32] Youquan Liu, Lingdong Kong, Jun Cen, Runnan Chen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. Segment any point cloud sequences by distilling vision foundation models. *Advances in Neural Information Processing Systems*, 36, 2024.

[33] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2949–2958, 2021.

[34] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[35] Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2906–2917, 2021.

[36] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[37] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 604–621. Springer, 2022.

[38] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. Openscene: 3d scene understanding with open vocabularies. *arXiv preprint arXiv:2211.15654*, 2022.

[39] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9277–9286, 2019.

[40] Guocheng Qian, Xingdi Zhang, Abdullah Hamdi, and Bernard Ghanem. Pix4point: Image pretrained transformers for 3d point cloud understanding. *arXiv preprint arXiv:2208.12259*, 2022.

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[42] Aditya Sanghi. Info3d: Representation learning on 3d objects using mutual information maximization and contrastive learning. In *European Conference on Computer Vision*, pages 626–642. Springer, 2020.

[43] Jonathan Sauder and Bjarne Sievers. Self-supervised deep learning on point clouds by reconstructing space. *Advances in Neural Information Processing Systems*, 32, 2019.

[44] Corentin Sautier, Gilles Puy, Spyros Gidaris, Alexandre Boulch, Andrei Bursuc, and Renaud Marlet. Image-to-lidar self-supervised distillation for autonomous driving data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9891–9901, 2022.

[45] Yichao Shen, Zigang Geng, Yuhui Yuan, Yutong Lin, Ze Liu, Chunyu Wang, Han Hu, Nanning Zheng, and Baining Guo. V-detr: Detr with vertex relative position encoding for 3d object detection. *arXiv preprint arXiv:2308.04409*, 2023.

[46] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[47] Haiyang Wang, Shaocong Dong, Shaoshuai Shi, Aoxue Li, Jianan Li, Zhenguo Li, Liwei Wang, et al. Cagroup3d: Class-aware grouping for 3d object detection on point clouds. *Advances in Neural Information Processing Systems*, 35:29975–29988, 2022.

[48] Xiaoyang Wu, Zhuotao Tian, Xin Wen, Bohao Peng, Xihui Liu, Kaicheng Yu, and Hengshuang Zhao. Towards large-scale 3d representation learning with multi-dataset point prompt training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19551–19562, 2024.

[49] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *European conference on computer vision*, pages 574–591. Springer, 2020.

[50] Sihao Yu, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Zizhen Wang, and Xueqi Cheng. A re-balancing strategy for class-imbalanced classification based on instance difficulty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 70–79, 2022.

[51] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen 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*, pages 19313–19322, 2022.

[52] Junbo Zhang, Runpei Dong, and Kaisheng Ma. Clip-fo3d: Learning free open-world 3d scene representations from 2d dense clip. *arXiv preprint arXiv:2303.04748*, 2023.

[53] Renrui Zhang, Ziyu Guo, Peng Gao, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, and Hongsheng Li. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *Advances in neural information processing systems*, 35:27061–27074, 2022.

[54] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8552–8562, 2022.

[55] Zaiwei Zhang, Rohit Girdhar, Armand Joulin, and Ishan Misra. Self-supervised pretraining of 3d features on any point-cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10252–10263, 2021.

[56] Long Zhao, Nitesh B Gundavarapu, Liangzhe Yuan, Hao Zhou, Shen Yan, Jennifer J Sun, Luke Friedman, Rui Qian, Tobias Weyand, Yue Zhao, et al. Videoprism: A foundational visual encoder for video understanding. *arXiv preprint arXiv:2402.13217*, 2024.

[57] Yongheng Zhao, Tolga Birdal, Haowen Deng, and Federico Tombari. 3d point capsule networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1009–1018, 2019.

[58] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. *Advances in neural information processing systems*, 27, 2014.

[59] Chong Zhou, Chen Change Loy, and Bo Dai. Denseclip: Extract free dense labels from clip. *arXiv preprint arXiv:2112.01071*, 2021.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading “NeurIPS paper checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

   Answer: [Yes]

   Justification: in Abstract and Introduction

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: in Limitations and Future Work

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: No theory assumptions and proofs.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Disclose in Self-supervised Pre-training and Fine-tuning

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Code will be released upon acceptance.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Specify in Self-supervised Pre-training and Fine-tuning

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: No error bars

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Provided in Self-supervised Pre-training and Fine-tuning.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Readed and conform with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Experiments done with public dataset and open-source foundation models.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Have no risk

    Guidelines:

    - The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Cited all relative works.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assests are introduced.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.