Segment3D: Learning Fine-Grained Class-Agnostic 3D Segmentation without Manual Labels

Rui Huang[1], Songyou Peng[2], Ayça Takmaz[2], Federico Tombari[3], Marc Pollefeys[2,4], Shiji Song[1], Gao Huang[1,*], and Francis Engelmann[2,3]

[1]Tsinghua University [2]ETH Zurich
[3]Google [4]Microsoft [*]Corr. Author
https://segment3d.github.io

Fig. 1: Fine-Grained Class-Agnostic 3D Point Cloud Segmentation. Segment3D predicts highly accurate segmentation masks (*right*), improves over state-of-the-art 3D segmentation methods (e.g., Mask3D [46], *left*), and does not require manually labeled 3D training data. This is achieved through the automatic generation of high-quality training masks using foundation models for image segmentation [28].

Abstract. Current 3D scene segmentation methods are heavily dependent on manually annotated 3D training datasets. Such manual annotations are labor-intensive, and often lack fine-grained details. Furthermore, models trained on this data typically struggle to recognize object classes beyond the annotated training classes, i.e., they do not generalize well to unseen domains and require additional domain-specific annotations. In contrast, recent 2D foundation models have demonstrated strong generalization and impressive zero-shot abilities, inspiring us to incorporate these characteristics from 2D models into 3D models. Therefore, we explore the use of image segmentation foundation models to automatically generate high-quality training labels for 3D segmentation models. The resulting model, Segment3D, generalizes significantly better than the models trained on costly manual 3D labels and enables easily adding new training data to further boost the segmentation performance.

Keywords: Class-agnostic 3D segmentation · 3D Scene Understanding

1 Introduction

In this work, we propose Segment3D, a method for fine-grained class-agnostic 3D segmentation. In particular, dividing the space into coherent segments aligned with both the scene geometry and its semantics is a key challenge. This ability to accurately segment and interpret 3D scenes is fundamental for numerous downstream tasks [52, 61], intelligent assistants and autonomous robots [30, 64].

Current methods for 3D indoor-scene understanding mostly focus on semantic [35, 39, 40, 47, 53] and instance segmentation [9, 17, 31, 32, 46, 54]. These approaches, while effective on popular benchmarks [13, 15, 44, 50], have limitations stemming from their training. Primarily, they depend on extensive manually labeled 3D training sets that are both time-consuming and challenging to annotate. Importantly, their performance often deteriorates when applied to scenarios beyond their training data, limiting their effectiveness in diverse, real-world scenarios. This becomes particularly apparent under the recently emerging task of *open-vocabulary* 3D scene understanding [18, 27, 33, 38, 51] that aims to segment arbitrary user queries, which naturally go beyond the pre-defined set of training-set classes. Concurrently, the recent surge in foundation models, especially 2D vision-language models [25, 28, 41, 60], demonstrates remarkable potential. Trained on internet-scale data, these models exhibit an extraordinary ability to generalize, even in a zero-shot setting, to new and different input distributions. However, their application has been predominantly confined to 2D data. For instance, SAM [28] has shown impressive results in 2D image segmentation, but its applicability to 3D scene understanding remains largely under-explored. All these factors give rise to the research question:

How to leverage 2D foundation models for class-agnostic 3D scene segmentation without requiring manually labeled 3D data?

Recently, SAM3D [57] has proposed a straightforward method that uses posed RGB-D images corresponding to a 3D scene. They predict segmentation masks for all input RGB-D images with SAM [28], and projected them into the 3D space. Next, through an iterative bottom-up process, the 3D masks of partial scenes are merged to derive the final 3D segmentation (see Fig. 2). However, variations in perspectives across frames can lead to conflicting segmentations, which introduces inconsistency during the merging process. Besides, the inference speed of SAM3D is slow, due to running SAM on the fly for a large number of RGB images and the cumbersome non-learned merging process. In this paper, we seek to circumvent both the slow speed and the issue of merging inconsistent segmentation across frames.

Towards this goal, we introduce Segment3D, a clean and efficient approach that utilizes a 3D model to achieve class-agnostic, fine-grained 3D segmentation at speed. Segment3D employs a two-stage training approach that requires no hand-annotated labels. We first train our class-agnostic 3D segmentation model

with RGB-D data, which are transformed into partial RGB-D point clouds. The supervision is provided by the segmentation masks produced by SAM and projected into 3D. While SAM can automatically generate masks for entire images, sometimes the results may suffer from over-segmentation (see Fig. 4), which is not desired in our context. Therefore, we propose a Mask Generation Module to generate high-quality, complete masks of objects for a given RGB image. Since there is readily available large-scale RGB-D data, additional data can be effortlessly integrated to further enhance segmentation performance. This pre-training stage lays the groundwork for understanding the 3D structure from 2D annotations. However, as our ultimate objective is the segmentation of full 3D scenes, we must bridge the domain gap between partial point clouds and the more comprehensive 3D point clouds obtained from 3D scanners or reconstruction techniques [14, 24]. To this end, in the second stage, we fine-tune the model on full 3D point clouds in a self-supervised manner, utilizing high-confidence mask predictions from the pre-trained model as training signal.

We demonstrate strong performance in class-agnostic segmentation on ScanNet [13, 44] and the newly released ScanNet++ [58]. Though Segment3D is trained on the indoor dataset, transferring it directly to the outdoor nuScenes dataset [4] also yields surprisingly good results. Finally, we show the use of Segment3D for improving open-vocabulary 3D instance segmentation [51]. Overall, the contributions of this paper are as follows:

- We introduce Segment3D, a novel approach and training strategy for fine-grained class-agnostic 3D point cloud segmentation without manually annotated labels.
- We propose a Mask Generation Module that utilizes the 2D foundation model to automatically generate high-quality, complete training masks.
- We show that Segment3D demonstrates strong generalization compared to a wide range of baselines, including fully supervised methods trained on carefully annotated datasets.

2 Related Work

3D Instance Segmentation. Current models have seen a significant development over the last years, from proposal-based [56, 59], over grouping-based [9, 26, 31, 54], to recent Transformer-based [32, 46] methods. In this work, we follow the currently best-performing Transformer-based paradigm. Despite impressive advancements, a shared limitation is the dependence on costly manual ground-truth annotations. Recently, there have been efforts to automate the annotation process by bundling state-of-the-art segmentation models [55], but they are still limited to pre-defined object classes. However, in the context of open-world 3D scene understanding, the importance of semantic classification for closed-set categories has diminished. Instead, in this work, we propose a general 3D segmentation method trained on automatically generated labels.

Foundation Models. Foundation models, particularly those that are multi-modal [25,41], have revolutionized the field of AI by leveraging extensive image-text pre-training. These models can derive rich image representations guided by natural language descriptions, enabling a variety of downstream tasks [21,37,42]. Another line of work [6,36], based on self-supervised learning, employs image training and yields high-performance features directly applicable as inputs for linear classifiers. Segment Anything Model (SAM) [28] has recently advanced the performance of foundation models for image segmentation. SAM has undergone training on a diverse, high-quality dataset comprising more than 1 billion masks. This training equips SAM with the ability to generalize to novel object types and images, surpassing the scope of its observations during the training process. Additionally, SAM can generate high-quality, fine-grained masks. Our work leverages the power of SAM for pre-training a 3D segmentation model, which is later also used for generating supervision signals for fine-tuning on scene-level point clouds.

Open-Vocabulary 3D Scene Understanding. Recently, there has been an increased interest in 3D open-vocabulary scene understanding. This new field utilizes the zero-shot recognition abilities of 2D vision-language models [25,41], enabling a more comprehensive understanding of diverse and previously unseen 3D environments [8,16,22,23,27,29,33,38,51,62,63]. PLA [16] aligns point cloud features with captions extracted from multi-view images of a scene to enable open-vocabulary recognition. OpenScene [38] distills per-pixel image features to 3D point clouds, generating point-wise scene representations co-embedded with text and image pixels in CLIP feature space. However, it mainly focuses on semantic segmentation and exhibits a limited understanding of object instances. To this end, OpenMask3D [51] predicts class-agnostic 3D instance masks and aggregates per-mask features via multi-view fusion of CLIP-based image embeddings. Nevertheless, the segmentation model it uses is trained on closed-set labels, lacking generalization to the open world. Our method contributes to this area by providing open-set class-agnostic masks, which can serve as foundational inputs for models such as OpenMask3D.

3 Method

We aim to develop a method capable of segmenting any object within a given 3D scene. Relying on existing 3D training datasets cannot accomplish this goal, as their mask annotations are limited to a predefined set of object classes [1,2,13,48]. Consequently, models trained on such datasets fail to generalize effectively to new, unseen classes. Recently, with SAM [28] showing extraordinary generalization ability in 2D segmentation, SAM3D [57] has proposed to simply project and merge SAM predictions in 3D from posed RGB-D images, as illustrated in Fig. 2. However, it suffers from inconsistent results and slow inference speed, limiting its practical applicability.

To fully exploit SAM for consistent 3D segmentation at speed, we introduce a clean and efficient method which utilizes a 3D model to directly segment the

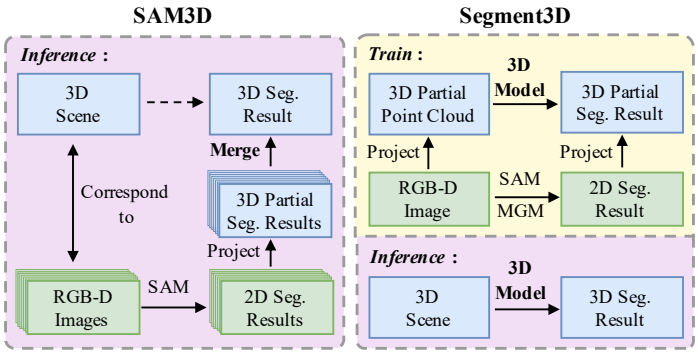

Fig. 2: SAM3D vs. Segment3D. SAM3D [57] (*left*) merges 2D segmentation masks of RGB-D images generated by SAM to obtain 3D segmentation of entire scenes. However, conflicting segmentations across frames introduce inconsistency during the merging process. Moreover, it is slow because of extensive image inference and the cumbersome merging procedure. Instead, our Segment3D (*right*) utilizes a 3D model to directly segment entire 3D scenes which is clean and efficient.

entire scene. First, we propose a Mask Generation Module to generate high-quality, complete object masks from SAM predictions (Sec. 3.1). Next, we train our 3D segmentation model with RGB-D data and the generated object masks, which are transformed into *partial* point clouds (Sec. 3.2). Since the final goal is to segment 3D scenes, we need to bridge the inevitable domain gap between *partial* RGB-D point clouds and *full* 3D point clouds from 3D scanners or reconstruction methods. We therefore fine-tune our model on *full* 3D point clouds using high-confident mask predictions from our pre-trained model (Sec. 3.3). The overall framework is illustrated in Fig. 3.

3.1 Generating Pseudo Ground-Truth Masks

Review on SAM. We first review the automatic mask generation pipeline of SAM [28]. SAM is prompted with a regular grid of 32×32 points on the RGB image. Each point predicts masks at three different granularities: whole, part, and subpart. Next, all three masks for all points are filtered by the model's predicted IoU score as well as the stability of the masks with respect to the binarization threshold. Then, the remaining masks undergo Non-Maximum Suppression (NMS). Finally, if a pixel is encompassed by multiple masks, the mask ID with the highest predicted IoU is assigned to the pixel.

However, as shown in Fig. 4, the results obtained in this way could suffer from over-segmentation. A naive idea is to only use the masks generated by the "whole" branch. Nevertheless, we find that the "whole" branch typically generates high-quality masks for large objects while overlooking small objects in the image. In contrast, the "part" and "subpart" branches not only generate masks for parts of large objects but also predict masks for small and tiny objects. Thus, to enable our 3D model to achieve *fine-grained* segmentation capabilities for details and

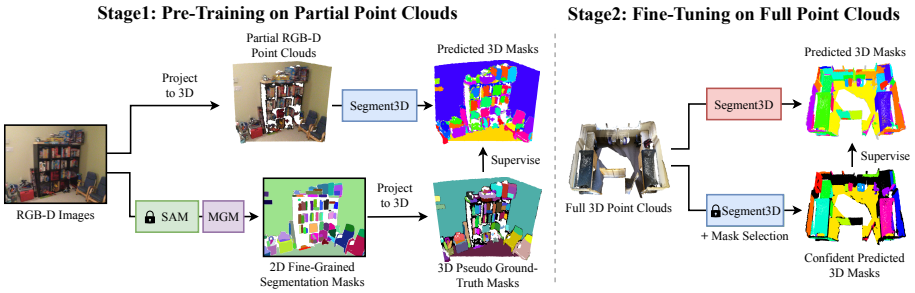

Fig. 3: Method Overview. Training Segment3D involves two stages: The first stage (*left*) relies on largely available RGB-D image sequences and SAM ▢, a pre-trained foundation model for 2D image segmentation [28]. Segment3D ▢ is pre-trained on partial RGB-D point clouds and supervised with pseudo ground-truth masks generated by Mask Generation Module (MGM) ▢. Due to the domain gap between partial and full point clouds, in the second stage (*right*), Segment3D ▢ is fine-tuned with confident masks predicted by the pre-trained Segment3D ▢.

small objects, we must retain the outputs from the "part" and "subpart" branches. The problem is how to preserve their segmentation for small and tiny objects while removing their segmentation for parts of large objects.

Mask Generation Module (MGM). Here we introduce our Mask Generation Module (MGM). Since SAM's automatic mask generation pipeline can provide reasonably high-quality masks for a given image, we start based on it. Next, we compare the result with the output of the "whole" branch to determine which masks are parts of large objects and remove them. Specifically, for each mask in the automatic output m_i^{auto}, we iterate through the masks of the entire branch, calculating the intersection m_{ij}^{inter} between m_i^{auto} and m_j^{whole}. If $m_{ij}^{\text{inter}}/m_i^{\text{auto}} \geq t$, then we consider m_i^{auto} to be a part of m_j^{whole}. Our preliminary results show that the process of extracting valid masks is highly robust to the hyper-parameter t and we set it as 0.9 by default. We remove the part masks and combine the remaining masks with the output of the "whole" branch, which will serve as our training supervision. With MGM, we observe notable improvements at all object mask sizes, especially on medium-sized and large objects (see Table 6).

3.2 Stage 1: Pre-Training on RGB-D Point Clouds

In contrast to the relatively scarce availability of 3D data, there is an abundance of 2D data, particularly RGB-D images, which are readily accessible. For example, ScanNet [13] comprises merely 1513 3D scans, compared to the substantially larger collection of 2.5 million RGB-D images. Therefore, we first pre-train our 3D segmentation model on partial RGB-D point clouds.

Training-Set Preparation. Starting from a collection of RGB-D frames, we create partial 3D point clouds and their corresponding pseudo ground-truth 3D masks. Note that the labels can be automatically obtained without manual effort.

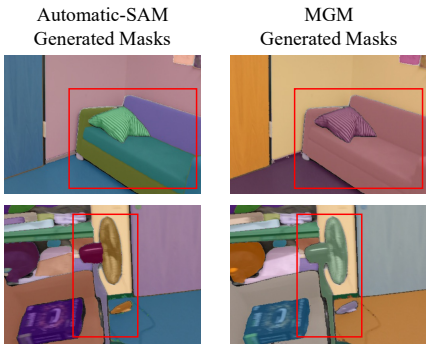

Fig. 4: Automatic-SAM Generated (*left*) **vs. MGM Generated** (*right*) **Masks from ScanNet** [13]. The former exhibits over-segmentation, while the latter does not.

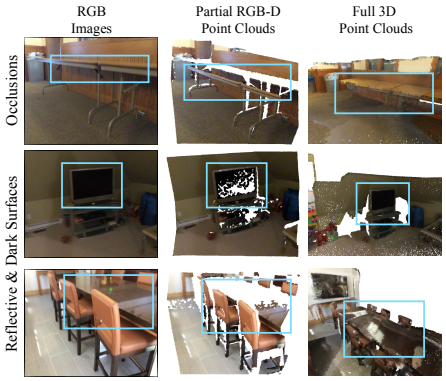

Fig. 5: Partial (*center*) **vs. Full** (*right*) **Point Clouds from ScanNet** [13]. Full point clouds are more complete and exhibit fewer occlusions due to reconstruction over multiple viewpoints.

For each frame in a large RGB-D dataset, we first transform the 2D depth map to a partial 3D point cloud. To do so, we need to know the intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$ and extrinsic matrix $\mathbf{T} = \begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} \in \mathbb{R}^{3\times4}$. For each pixel $\mathbf{p} = (u, v)$, we can transform it with its depth value $D_{\mathbf{p}}$ into a 3D point $\mathbf{P}$ in world coordinates as follows:

$$\mathbf{P} = \mathbf{R}^{\top} \cdot (D_{\mathbf{p}} \cdot \mathbf{K}^{-1} \cdot \tilde{\mathbf{p}}) - \mathbf{R}^{\top}\mathbf{t}, \tag{1}$$

where $\tilde{\mathbf{p}}$ is the homogeneous coordinate of $\mathbf{p}$. By applying Eq. (1) to all pixels in the depth map D and associating their per-pixel RGB value, we obtain the input partial 3D point cloud. Next, we obtain the pseudo ground-truth 3D segmentation masks for this point cloud with the 2D masks generated by MGM in the same way. Since we know the one-to-one mapping between 2D pixel and 3D points from Eq. (1), we directly obtain the per-point 3D mask labels.

Model Architecture. We use a model inspired by Mask3D [46] to train a class-agnostic 3D segmentation model. The model comprises a sparse convolutional backbone derived from MinkowskiUNet [12] and a transformer decoder, as in MaskFormer [10, 11]. We adopt a set of queries to represent the masks, each of which is initialized with a positional embedding. Specifically, we select query positions with furthest point sampling (FPS) and use their Fourier positional encodings as the query embeddings. Leveraging the transformer decoder, all the mask queries are refined by progressively attending to point cloud features across multiple scales in parallel. Each mask query is subsequently decoded into both a mask feature and a binary label to predict whether the given query corresponds to a valid object or not. By computing cosine similarity scores between a mask feature and all point features within the point cloud, a heatmap is generated over

the point cloud. This heatmap is input to a sigmoid function, and thresholded at 0.5, resulting in the final binary mask.

Training with SAM Generated Masks. We supervise the model with two losses: the per-point supervision loss $\mathcal{L}_{\text{mask}}$ and a per-query supervision loss $\mathcal{L}_{\text{obj}}$. The loss $\mathcal{L}_{\text{mask}}$ enables learning a foreground-background segmentation for each mask, and is composed of a dice loss $\mathcal{L}_{\text{dice}}$ [34] and a binary cross-entropy loss $\mathcal{L}_{\text{ce}}$ for each point. The $\mathcal{L}_{\text{obj}}$ is a binary classification loss that indicates whether a query represents a valid "object" or "no object". This mechanism allows for the prediction of a variable number of masks, depending on the underlying scene content and geometry. Following prior work [5, 10, 46], we first adopt bipartite graph matching to establish correspondences between the set of predicted masks and the set of target masks provided by MGM as described before. If the predicted instance finds a matching target mask, then we assign it an "object" label; conversely, if there is no match, we assign "no object". In summary, we optimize the following losses:

$$\mathcal{L} = \mathcal{L}_{\text{mask}} + \lambda_{\text{obj}}\mathcal{L}_{\text{obj}}, \tag{2}$$

$$\mathcal{L}_{\text{mask}} = \lambda_{\text{dice}}\mathcal{L}_{\text{dice}} + \lambda_{\text{ce}}\mathcal{L}_{\text{ce}}, \tag{3}$$

where λ_* are hyperparameters that balance the contribution of each component in the loss. The binary classification loss $\mathcal{L}_{\text{obj}}$ is applied to all queries, while the mask loss $\mathcal{L}_{\text{mask}}$ is specifically applied to masks labeled as "object".

3.3 Stage 2: Self-Supervised Scene Fine-Tuning

After Stage 1, we obtain a class-agnostic 3D segmentation model by pre-training solely on RGB-D images and automatically generated labels from MGM. However, a fundamental domain gap persists between partial point clouds derived from RGB-D data and full point clouds acquired through 3D scanners or reconstruction methods [14, 24] (See Fig. 5). This gap exists mostly because of object occlusions, but also due to challenges of depth cameras to capture dark or reflective surfaces from a single viewpoint. Hence, depending solely on RGB-D frames for training a 3D segmentation model intended for full 3D scans proves inadequate. Therefore, we propose to further fine-tune our model on scene-level full 3D point clouds. The key idea to obtain 3D mask annotations for training on full point clouds is to use selected, high-confidence masks generated by the pre-trained model itself. Note that this stage requires no manual labels on full 3D scenes and proves essential for the performance of Segment3D (see Tab. 5).

Confidence-Based Mask Generation. Next, we outline the process of generating the supervision signal for the fine-tuning stage. The pre-trained model processes point clouds independently of their nature, be it partial or full. Therefore, when presented with a full 3D point cloud, the pre-trained 3D model produces a set of masks, each with a binary classification (valid or not) and a heatmap over all points, just as in Sec. 3.2. To assess the quality of the predicted masks, we compute a confidence score based on the confidence map $\sigma(\mathbf{h})$, where $\mathbf{h}$ is the predicted heatmap and σ is the sigmoid function. We then compute the average

confidence of those points for which $\sigma(\mathbf{h}) > 0.5$ as the confidence score of the predicted mask, denoted as c_{mask}. We also consider the classification as a valid object and use the probability from the binary classification assigned to the "object" category as the confidence score, denoted as c_{obj}. The final confidence score for each predicted mask is then the product of the two scores, $c = c_{\text{mask}} \cdot c_{\text{obj}}$. For fine-tuning our 3D segmentation model in Stage 2, we select the most confidently predicted masks above a threshold τ_c.

Training with the High-Confidence Generated Masks. For fine-tuning, we follow the same procedure as before and use $\mathcal{L}_{\text{mask}}$ as defined in Eq. 3. In contrast to the pre-training stage, the binary classification loss $\mathcal{L}_{\text{obj}}$, responsible for categorizing queries into valid or invalid, is omitted. Since we only select masks with high confidence for supervision, it can happen that some objects in the scene have no assigned ground truth mask. In such instances, deeming a correctly predicted mask for those objects as invalid would be detrimental. Table 5 illustrates the efficacy of the self-supervised fine-tuning process in comparison to pre-training alone.

4 Experiments

We firstly evaluate Segment3D on indoor ScanNet++ [58], ScanNet200 [13, 44] and outdoor nuScenes [4] in a class-agnostic segmentation setting (Sec. 4.1). We then show the advantage of our method in segmenting small objects, and provide analysis to understand the importance of fine-tuning, the effectiveness of Mask Generation Module and the potential of training on more data (Sec. 4.2). Finally, we demonstrate its application for the task of open-set 3D instance segmentation as proposed in OpenMask3D [51] (Sec. 4.3).

4.1 Comparing with State-of-the-Art Methods

Datasets. We first evaluate on the recently released ScanNet++ [58] which includes high-resolution 3D scans captured at sub-millimeter precision, and fine-grained annotations covering objects of varying sizes. Due to its comprehensive data annotation, all of our analytical experiments in Sec. 4.2 are also based on it. Next, we adopt the ScanNet200 [13, 44], which builds upon the classic ScanNet [13] dataset and extends the semantic annotations to 200 classes. Then, we test our model on the nuScenes [4] outdoor dataset. For training our model, we employ ScanNet [13, 44], which is collected through a lightweight RGB-D scanning process. See the appendix for the implementation details of our method.

Methods in Comparison. We compare with a wide range of prior art methods from different categories. Mask3D [46] is a state-of-the-art fully-supervised method trained on manually annotated 3D segmentation masks. Segment3D has the same backbone as Mask3D but instead of training on manually annotated 3D masks, it learns from automatically generated masks. Felzenszwalb *et al.* [20] proposed a graph-based method for segmentation which operates directly on

Table 1: Segmentation Scores on ScanNet++ [58]. The metric is average precision (AP) on the validation split. We report scores with and without post-processing (more details in Sec. 4.1). For reference, we show results on Mask3D trained on manual ScanNet200 labels.

Model	Avg. Inference Time/s	*without post-processing*			*with post-processing*		
		AP	**AP$_{50}$**	**AP$_{25}$**	**AP**	**AP$_{50}$**	**AP$_{25}$**
Mask3D [46]	0.7	8.7	15.5	27.2	14.3	21.3	29.9
SAM3D [57]	386.7	3.9	9.3	22.1	8.4	16.1	30.0
Felzenszwalb *et al.* [20]	12.6	5.8	11.6	27.2	–	–	–
Segment3D (**Ours**)	0.7	**13.0**	**23.8**	**38.3**	**20.2**	**30.9**	**42.7**

the 3D geometry. UnScene3D [45] leverages self-supervised color and geometry features to obtain coarse segmentation and refines it through self-training. SAM3D [57] merges the segmentation masks of RGB-D images generated by SAM [28] to obtain the 3D segmentation result.

Metrics. We report average precision (AP) scores at IoU thresholds of 25%, 50%, and averaged over the range [0.5:0.95:05] between predicted and ground truth masks. Consistent with common practices in the field [47, 49, 54], we also report scores after post-processing. This involves smoothing the predicted masks through graph-based oversegmentation [20], and splitting distant parts of the same mask via connected component clustering DBSCAN [19]. We also report the average inference time for these methods on ScanNet++ tested on A100.

Results on ScanNet++. Scores are reported in Table 1. Segment3D outperforms all previous methods by at least +**4.3** AP and up to +**12.7** AP$_{25}$. Notably, we achieve such improvements without any ground-truth mask annotation contrary to Mask3D trained on ScanNet200. In general, the performance of Mask3D (and other fully-supervised methods) depends on the quality of the annotated training dataset; often the manual annotation of small objects (pens, cellphones) or other fine-details is challenging. Instead, Segment3D relies on automatically generated high-quality masks from MGM using the 2D foundation model, which can capture fine-grained details without human annotation effort. Table 4 highlights that Segment3D excels particularly in predicting the more challenging small object masks. Furthermore, our method significantly outperforms SAM3D, indicating that Segment3D can circumvent the noise that arises during the merging process of SAM3D. Additionally, As shown in Table 1, Segment3D is three-order-of-magnitude faster than SAM3D in inference speed.

Fig. 6 shows segmentation results of several representative examples on ScanNet++. As can be seen, the scenes are quite diverse, presenting multiple challenges such as clutter and a wide range of mask sizes. Despite these challenges, our model predicts quite accurate and well-localized segmentation masks. For example, compared to Mask3D, our method is able to segment the finer-grained objects on top of the bed and the shelf. The masks of the computer screen and the chair in front of it are also less fragmented than predictions of the baselines.

Table 2: Segmentation Scores on ScanNet200 [13, 44]. The evaluation metric is average precision on the val split.

Model	AP	AP_{50}
Mask3D [46]	34.1	43.1
Felzenszwalb *et al.* [20]	6.1	12.1
UnScene3D [45]	15.9	32.2
SAM3D [57]	19.0	32.5
Segment3D (**Ours**)	**27.7**	**39.8**

Table 3: Segmentation Scores on nuScenes [4]. The evaluation metric is average precision on the val split.

Model	GT Labels	AP_{50}	AP_{25}
zero-shot test			
Mask3D	×	15.3	25.5
Segment3D	×	37.8	48.0
fine-tuned on nuScenes			
Mask3D	✓	46.5	57.8
Segment3D	×	42.5	55.5

Results on ScanNet200. As illustrated in Table 2, Segment3D outperforms all baselines that require no manual 3D labels, and even shows competitive results with Mask3D, which is trained on ScanNet200. Compared to Unscene3D, which leverages self-supervised color and geometry features to obtain coarse segmentation, we achieve superior results by using the Mask Generation Module to generate high-quality complete masks.

Results on Outdoor Data. The performance on nuScenes is demonstrated in Table 3. We observe that even though Segment3D is trained using purely indoor data, applying it directly to outdoor data produces surprisingly good results. This indicates the stronger generalizability of Segment3D over the fully-supervised method Mask3D when transferring from indoor to outdoor scenes. Furthermore, we report scores with fine-tuning on nuScenes which leads to overall improved results. Similar to Table 2, Mask3D performs best, however, Segment3D achieves very competitive performance *without any hand-labeled training data*. Refer to the appendix for additional results on STPLS3D [7] and Paris-Lille-3D [43] datasets.

4.2 Analysis Experiments

Performance on Different Mask Sizes. We proceed with an analysis of the performance of our Segment3D and Mask3D (the best-performing baseline) on object masks of various sizes. We categorize the size of masks based on the number of points they contain (small: [0, 2k], medium: [2k, 15k] large: [15k, ∞]). The results are reported in Table 4. Our method yields significantly improved

Table 4: Segmentation Scores on Different Mask Sizes. Segmented3D improves over Mask3D, especially on small object masks. Details are in Sec. 4.2.

Mask Size	Large		Medium		Small	
ScanNet++	AP	AP_{50}	AP	AP_{50}	AP	AP_{50}
Mask3D	13.8	25.1	13.6	23.3	1.8	4.3
Segment3D	**18.3**	**31.9**	**16.6**	**29.8**	**8.4**	**17.3**
	(+4.5)	(+6.8)	(+3.0)	(+6.5)	(+6.6)	(+13.0)

Fig. 6: Qualitative Results on ScanNet++. From top to bottom, we show the colored input 3D scenes, the segmentation masks predicted by SAM3D [57], Mask3D [46], our Segment3D and the ground truth 3D mask annotations.

segmentation results on objects of various sizes, especially on small objects. This confirms our intuition that Mask3D performs poorly on small-sized object masks as those are typically harder to manually annotate. In contrast, Segment3D utilizes masks generated by MGM with the foundation model as supervision, capturing fine-grained scene details. This showcases the usefulness of foundation models and raises the question if manually labeled large-scale 3D datasets are necessary for training 3D point cloud segmentation models.

The Importance of Two-Stage Training. Next, we compare the performance of Segment3D pre-trained solely on partial RGB-D point clouds (Stage 1) and with additional fine-tuning on full point clouds (Stage 2). Scores are reported in Table 5. The additional fine-tuning stage almost doubles the segmentation performance of our model on the most challenging AP metric. By training with the predicted high-confidence masks, Segment3D effectively reduces the inherent

Table 5: The Importance of Two-Stage Training. Fine-tuning on full point clouds supervised by confident predicted 3D masks significantly surpasses pre-training on projected 2D SAM masks alone.

Training Stages	**AP**	**AP$_{50}$**	**AP$_{25}$**
Pre-Training (Stage 1)	6.8	14.2	30.6
+ Fine-Tuning (Stage 2)	**13.0** (+6.2)	**23.8** (+9.6)	**38.3** (+7.7)

Table 6: Effectiveness of Mask Generation Module. Compared to the model trained with masks generated by Automatic-SAM, our model shows improvement across all object mask sizes, particularly for medium-sized and large objects.

Mask Size	Large		Medium		Small	
Masks Generated by	AP	AP$_{50}$	AP	AP$_{50}$	AP	AP$_{50}$
Automatic-SAM	16.8	30.6	14.2	27.2	7.5	16.2
MGM	**18.3** (+1.5)	**31.9** (+1.3)	**16.6** (+2.4)	**29.8** (+2.6)	**8.4** (+0.9)	**17.3** (+1.1)

domain gap between the partial point clouds derived from RGB-D images and full 3D point clouds.

The Effectiveness of Mask Generation Module. To verify whether the MGM enhances the model's performance by reducing the over-segmentation issue and providing more complete masks, we conducted evaluations on objects of different sizes. As shown in Table 6, we can see that compared to the model trained with masks generated by Automatic-SAM, our model notably improves the performance of large and medium-sized objects which demonstrates the effectiveness of MGM.

Pre-Training with Additional Data. Since RGB-D data is available in abundance and masks can be automatically generated, it is natural to ask if pre-training on additional masks will further improve the overall performance. To that end, we perform a first experiment where we select additional frames from the training set of the ScanNet++ dataset (to increase the variety of training data). Table 7 shows an impressive performance boost of **+4.1** AP. For future work, considering that this improvement is obtained simply by adding more automatically generated masks to the pre-training, our approach seems promising to train on even more data. It could even be plausible to train on internet images combined with monocular depth estimation, such as ZoeDepth [3], to compute the partial point clouds.

4.3 Application: Open-Set Scene Understanding

A real-world application of our class-agnostic 3D segmentation method is open-vocabulary 3D scene understanding, as implemented recently by OpenMask3D [51]. Given a 3D scene, a user can search for arbitrary objects via text prompts (see Fig. 7). A core component of OpenMask3D is Mask3D, which segments the

Table 7: Results with Additional Training Data. The performance is further boosted with more training data.

Pre-Training Data	AP	AP_{50}
ScanNet	13.0	23.8
ScanNet, ScanNet++	**17.1** (+4.1)	**26.7** (+2.9)

Table 8: Open-Set 3D Scene Understanding. The evaluation metric is average precision with an IoU threshold of 50%.

Segmentor	ScanNet++	Replica
Mask3D [46]	15.0	18.0
Segment3D (Ours)	**18.5** (+3.5)	**18.4** (+0.4)

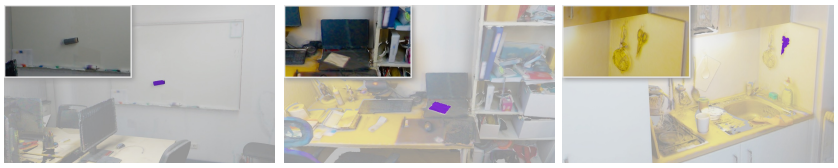

Fig. 7: Open-Set 3D Object Retrieval Results. Given a text prompt (*bottom*), OpenMask3D [51] finds the corresponding object masks ■ in a given 3D scene (*top*). We adapt OpenMask3D and use fine-grained masks from our Segment3D method. We show 3D reconstructions and RGB images for better visualization (*top left corner*).

scene into a set of object masks. Since Mask3D is trained on the closed-set of labeled annotations from ScanNet, its masks are not truly class-agnostic or open-vocabulary. We therefore replace Mask3D with our class-agnostic Segment3D. We evaluate on the closed-set labels of ScanNet++ and Replica in Table 8 and report an improvement of up to **+3.5** AP_{50}. Since Segment3D can accurately generate fine-grained masks in a class-agnostic manner, we are able to retrieve objects such as "eraser", "paper" and "scissors", which are small in scale.

5 Conclusion

We have presented Segment3D, a simple, yet powerful class-agnostic 3D segmentation model. Segment3D employs a two-stage training framework that requires no manually annotated labels. We propose a Mask Generation Module to automatically generate high-quality, complete training masks with the 2D foundation model. The model is first pre-trained on RGB-D data with predicted 2D masks, then fine-tuned on full point clouds. Segment3D shows strong generalization and even outperforms existing 3D segmentation models that rely on hand-labeled 3D training scenes. Indeed, this raises the question of whether hand-labeled 3D training datasets are as essential as they were once thought to be.

Acknowledgement. Gao Huang is supported in part by the National Natural Science Foundation of China under grants (62321005 and 42327901). Francis Engelmann is partially supported by an ETH AI Center postdoctoral research fellowship and an ETH Zurich Career Seed Award. Songyou Peng is supported by an Innosuisse funding (100.567 IP-ICT), and Ayça Takmaz is supported by an Innosuisse grant (48727.1 IP-ICT).

References

1. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3D Semantic Parsing of Large-Scale Indoor Spaces. In: CVPR (2016) 4
2. Baruch, G., Chen, Z., Dehghan, A., Dimry, T., Feigin, Y., Fu, P., Gebauer, T., Joffe, B., Kurz, D., Schwartz, A., et al.: ARKitScenes: A Diverse Real-World Dataset for 3D Indoor Scene Understanding using Mobile RGB-D Data. arXiv preprint arXiv:2111.08897 (2021) 4
3. Bhat, S.F., Birkl, R., Wofk, D., Wonka, P., Müller, M.: ZoeDepth: Zero-Shot Transfer by Combining Relative and Metric Depth. arXiv preprint arXiv:2302.12288 (2023) 13
4. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuScenes: A Multimodal Dataset for Autonomous Driving. In: CVPR (2020) 3, 9, 11
5. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-End Object Detection with Transformers. In: ECCV (2020) 8
6. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging Properties in Self-Supervised Vision Transformers. In: ICCV (2021) 4
7. Chen, M., Hu, Q., Yu, Z., Thomas, H., Feng, A., Hou, Y., McCullough, K., Ren, F., Soibelman, L.: STPLS3D: A Large-Scale Synthetic and Real Aerial Photogrammetry 3D Point Cloud Dataset. arXiv preprint arXiv:2203.09065 (2022) 11
8. Chen, R., Liu, Y., Kong, L., Zhu, X., Ma, Y., Li, Y., Hou, Y., Qiao, Y., Wang, W.: CLIP2Scene: Towards Label-Efficient 3D Scene Understanding by CLIP. In: CVPR (2023) 4
9. Chen, S., Fang, J., Zhang, Q., Liu, W., Wang, X.: Hierarchical Aggregation for 3D Instance Segmentation. In: ICCV (2021) 2, 3
10. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-Attention Mask Transformer for Universal Image Segmentation. In: CVPR (2022) 7, 8
11. Cheng, B., Schwing, A., Kirillov, A.: Per-Pixel Classification is Not All You Need for Semantic Segmentation. In: NeurIPS (2021) 7
12. Choy, C., Gwak, J., Savarese, S.: 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In: CVPR (2019) 7
13. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In: CVPR (2017) 2, 3, 4, 6, 7, 9, 11
14. Dai, A., Nießner, M., Zollhöfer, M., Izadi, S., Theobalt, C.: BundleFusion: Real-time Globally Consistent 3D Reconstruction using On-the-fly Surface Reintegration. TOG (2017) 3, 8
15. Delitzas, A., Takmaz, A., Tombari, F., Sumner, R., Pollefeys, M., Engelmann, F.: SceneFun3D: Fine-Grained Functionality and Affordance Understanding in 3D Scenes. In: CVPR (2024) 2
16. Ding, R., Yang, J., Xue, C., Zhang, W., Bai, S., Qi, X.: PLA: Language-Driven Open-Vocabulary 3D Scene Understanding. In: CVPR (2023) 4
17. Engelmann, F., Bokeloh, M., Fathi, A., Leibe, B., Nießner, M.: 3D-MPA: Multi-Proposal Aggregation for 3D Semantic Instance Segmentation. In: CVPR (2020) 2
18. Engelmann, F., Manhardt, F., Niemeyer, M., Tateno, K., Tombari, F.: OpenNeRF: Open Set 3D Neural Scene Segmentation with Pixel-Wise Features and Rendered Novel Views. In: ICLR (2024) 2

19. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: KDD (1996) 10
20. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Graph-Based Image Segmentation. IJCV (2004) 9, 10, 11
21. Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-Vocabulary Object Detection via Vision and Language Knowledge Distillation. In: ICLR (2022) 4
22. Ha, H., Song, S.: Semantic Abstraction: Open-World 3D Scene Understanding from 2D Vision-Language Models. In: CoRL (2022) 4
23. Huang, T., Dong, B., Yang, Y., Huang, X., Lau, R.W., Ouyang, W., Zuo, W.: CLIP2Point: Transfer CLIP to Point Cloud Classification with Image-Depth Pre-Training. In: ICCV (2023) 4
24. Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A., et al.: KinectFusion: Real-time 3D Reconstruction and Interaction using a Moving Depth Camera. In: UIST (2011) 3, 8
25. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up Visual and Vision-Language Representation Learning with Noisy Text Supervision. In: ICML (2021) 2, 4
26. Jiang, L., Zhao, H., Shi, S., Liu, S., Fu, C.W., Jia, J.: PointGroup: Dual-Set Point Grouping for 3D Instance Segmentation. In: CVPR (2020) 3
27. Kerr, J., Kim, C.M., Goldberg, K., Kanazawa, A., Tancik, M.: LERF: Language Embedded Radiance Fields. In: ICCV (2023) 2, 4
28. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment Anything. In: ICCV (2023) 1, 2, 4, 5, 6, 10
29. Kobayashi, S., Matsumoto, E., Sitzmann, V.: Decomposing Nerf for Editing via Feature Field Distillation. In: NeurIPS (2022) 4
30. Lemke, O., Bauer, Z., Zurbrügg, R., Pollefeys, M., Engelmann, F., Blum, H.: Spotcompose: A framework for open-vocabulary object retrieval and drawer manipulation in point clouds. In: 2nd Workshop on Mobile Manipulation and Embodied Intelligence at ICRA 2024 (2024) 2
31. Liang, Z., Li, Z., Xu, S., Tan, M., Jia, K.: Instance Segmentation in 3D Scenes using Semantic Superpoint Tree Networks. In: CVPR (2021) 2, 3
32. Lu, J., Deng, J., Wang, C., He, J., Zhang, T.: Query Refinement Transformer for 3D Instance Segmentation. In: ICCV (2023) 2, 3
33. Lu, Y., Xu, C., Wei, X., Xie, X., Tomizuka, M., Keutzer, K., Zhang, S.: Open-Vocabulary Point-Cloud Object Detection without 3D Annotation. In: CVPR (2023) 2, 4
34. Milletari, F., Navab, N., Ahmadi, S.A.: V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In: 3DV (2016) 8
35. Nekrasov, A., Schult, J., Litany, O., Leibe, B., Engelmann, F.: Mix3D: Out-of-context data augmentation for 3D scenes. In: 3DV (2021) 2
36. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: DINOv2: Learning Robust Visual Features without Supervision. arXiv preprint arXiv:2304.07193 (2023) 4
37. Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., Lischinski, D.: StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. In: ICCV (2021) 4
38. Peng, S., Genova, K., Jiang, C., Tagliasacchi, A., Pollefeys, M., Funkhouser, T., et al.: OpenScene: 3D Scene Understanding with Open Vocabularies. In: CVPR (2023) 2, 4

39. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In: CVPR (2017) 2
40. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: NeurIPS (2017) 2
41. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning Transferable Visual Models from Natural Language Supervision. In: ICML (2021) 2, 4
42. Rao, Y., Zhao, W., Chen, G., Tang, Y., Zhu, Z., Huang, G., Zhou, J., Lu, J.: DenseCLIP: Language-Guided Dense Prediction with Context-Aware Prompting. In: CVPR (2022) 4
43. Roynard, X., Deschaud, J.E., Goulette, F.: Paris-Lille-3D: A Large and High-Quality Ground-Truth Urban Point Cloud Dataset for Automatic Segmentation and Classification. IJRR (2018) 11
44. Rozenberszki, D., Litany, O., Dai, A.: Language-Grounded Indoor 3D Semantic Segmentation in the Wild. In: ECCV (2022) 2, 3, 9, 11
45. Rozenberszki, D., Litany, O., Dai, A.: UnScene3D: Unsupervised 3D Instance Segmentation for Indoor Scenes. In: CVPR (2024) 10, 11
46. Schult, J., Engelmann, F., Hermans, A., Litany, O., Tang, S., Leibe, B.: Mask3D: Mask Transformer for 3D Semantic Instance Segmentation. In: ICRA (2023) 1, 2, 3, 7, 8, 9, 10, 11, 12, 14
47. Schult, J., Engelmann, F., Kontogianni, T., Leibe, B.: DualConvMesh-Net: Joint Geodesic and Euclidean Convolutions on 3D Meshes. In: CVPR (2020) 2, 10
48. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., et al.: The Replica Dataset: A Digital Replica of Indoor Spaces. arXiv preprint arXiv:1906.05797 (2019) 4
49. Sun, J., Qing, C., Tan, J., Xu, X.: Superpoint Transformer for 3D Scene Instance Segmentation. In: AAAI (2023) 10
50. Sun, T., Hao, Y., Huang, S., Savarese, S., Schindler, K., Pollefeys, M., Armeni, I.: Nothing stands still: A spatiotemporal benchmark on 3d point cloud registration under large geometric and temporal change. arXiv preprint arXiv:2311.09346 (2023) 2
51. Takmaz, A., Fedele, E., Sumner, R.W., Pollefeys, M., Tombari, F., Engelmann, F.: OpenMask3D: Open-Vocabulary 3D Instance Segmentation. In: NeurIPS (2023) 2, 3, 4, 9, 13, 14
52. Takmaz, A., Schult, J., Kaftan, I., Akçay, M., Leibe, B., Sumner, R., Engelmann, F., Tang, S.: 3D Segmentation of Humans in Point Clouds with Synthetic Data. In: ICCV (2023) 2
53. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: KPConv: Flexible and Deformable Convolution for Point Clouds. In: ICCV (2019) 2
54. Vu, T., Kim, K., Luu, T.M., Nguyen, T., Yoo, C.D.: SoftGroup for 3D Instance Segmentation on Point Clouds. In: CVPR (2022) 2, 3, 10
55. Weder, S., Blum, H., Engelmann, F., Pollefeys, M.: LabelMaker: Automatic Semantic Label Generation from RGB-D Trajectories. In: 3DV (2024) 3
56. Yang, B., Wang, J., Clark, R., Hu, Q., Wang, S., Markham, A., Trigoni, N.: Learning Object Bounding Boxes for 3D Instance Segmentation on Point Clouds. In: NeurIPS (2019) 3
57. Yang, Y., Wu, X., He, T., Zhao, H., Liu, X.: SAM3D: Segment Anything in 3D Scenes. In: ICCVW (2023) 2, 4, 5, 10, 11, 12
58. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: ScanNet++: A High-Fidelity Dataset of 3D Indoor Scenes. In: ICCV (2023) 3, 9, 10

59. Yi, L., Zhao, W., Wang, H., Sung, M., Guibas, L.J.: GSPN: Generative Shape Proposal Network for 3D Instance Segmentation in Point Cloud. In: CVPR (2019) 3
60. Yue, Y., Das, A., Engelmann, F., Tang, S., Lenssen, J.: Improving 2D Feature Representations by 3D-Aware Fine-Tuning. In: ECCV (2024) 2
61. Yue, Y., Kontogianni, T., Schindler, K., Engelmann, F.: Connecting the Dots: Floorplan Reconstruction Using Two-Level Queries. In: CVPR (2023) 2
62. Zeng, Y., Jiang, C., Mao, J., Han, J., Ye, C., Huang, Q., Yeung, D.Y., Yang, Z., Liang, X., Xu, H.: CLIP2: Contrastive Language-Image-Point Pretraining from Real-World Point Cloud Data. In: CVPR (2023) 4
63. Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., Li, H.: PointCLIP: Point Cloud Understanding by CLIP. In: CVPR (2022) 4
64. Zurbrügg, R., Liu, Y., Engelmann, F., Kumar, S., Hutter, M., Patil, V., Yu, F.: ICGNet: A Unified Approach for Instance-Centric Grasping. In: ICRA (2024) 2