Open-Ended 3D Point Cloud Instance Segmentation

Phuc Nguyen[1,*] Minh Luu[1,*] Anh Tran[1] Cuong Pham[1,2] Khoi Nguyen[1]
[1]Movian AI [2]Posts & Telecommunications Inst. of Tech
{phucnda, hoangminh291101, anstar1111, ducminhkhoi}@gmail.com cuongpv@ptit.edu.vn

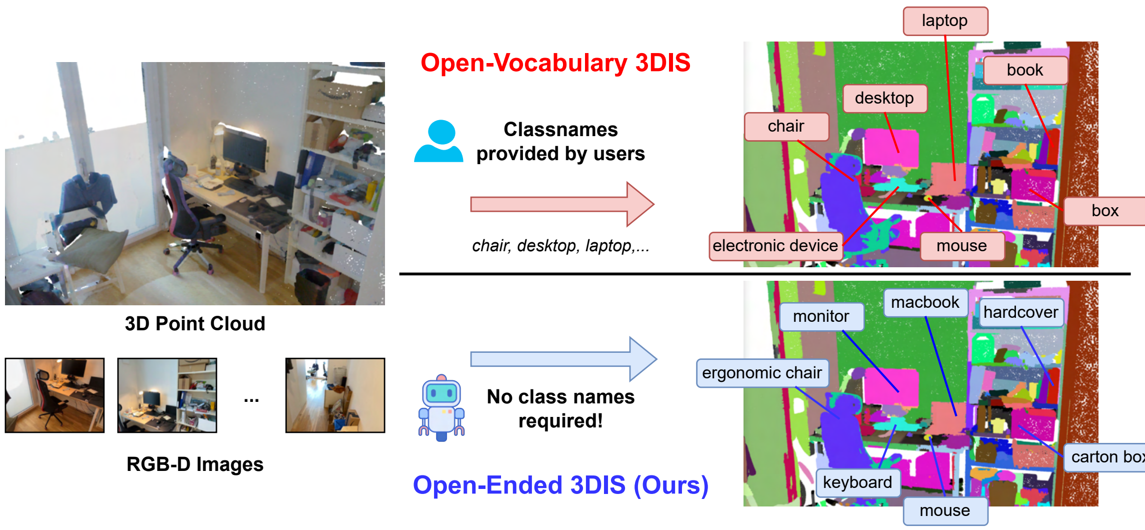

Figure 1. During testing, Open-Vocabulary 3D Instance Segmentation (OV-3DIS) utilizes CLIP to generate confidence scores for provided text prompts. However, its reliance on a predefined vocabulary during inference necessitates human intervention, restricting autonomy in intelligent agents. In contrast, our proposed Open-Ended 3D Instance Segmentation (OE-3DIS) leverages Large Language Models to enable open interaction with humans, recognizing fine-grained objects without being constrained by predefined class labels.

Abstract

Open-vocabulary 3D Instance Segmentation methods (OV-3DIS) have recently demonstrated their generalization ability to unseen objects. However, these methods still depend on predefined class names during inference, restricting agents' autonomy. To mitigate this constraint, we propose a novel problem termed ***Open-Ended 3D Instance Segmentation (OE-3DIS)****, which eliminates the necessity for predefined class names during testing. We present a comprehensive set of strong baselines inspired by OV-3DIS methodologies, utilizing 2D Multimodal Large Language Models. In addition, we introduce a novel token aggregation strategy that effectively fuses information from multi-view images. To evaluate the performance of our OE-3DIS system, we benchmark both the proposed baselines and our method on two widely used indoor datasets: ScanNet200 and ScanNet++. Our approach achieves substantial performance gains over the baselines on both datasets. Notably, even without access to ground-truth object class names during inference, our method outperforms Open3DIS, the current state-of-the-art in OV-3DIS. Source code available at:* https://github.com/PhucNDA/OE-3DIS.

1. Introduction

3D point cloud instance segmentation (3DIS) [5, 25, 30, 37, 50, 52], also known as closed-vocabulary 3D instance segmentation, aims to segment all points in a point cloud into instances of classes predefined in the training set. However, this approach is less practical for scenarios where the test classes are unknown or different from the training classes. This limitation has led to the development of open-vocabulary 3D instance segmentation (OV-3DIS) [28, 35, 39, 51, 55, 57]. Despite these advancements, OV-3DIS methods face several practical challenges. One major challenge is that class names must be predefined during inference, necessitating human intervention for scene under-

[*]These authors contributed equally to this work

standing. This requirement significantly impedes the perception of truly autonomous agents. A potential solution is to predefined large and comprehensive vocabularies; however, this can lead to inaccuracies and significantly degrade the performance of OV-3DIS when an excessive number of vocabularies are used.

To overcome these limitations, we introduce a novel task called **Open-Ended 3D Point Cloud Instance Segmentation (OE-3DIS)** see Fig. 1. Unlike traditional methods, OE-3DIS does not require predefined class names during testing. Given a 3D point cloud and RGBD sequence, the system automatically generates a set of 3D masks along with their class names. To evaluate the performance of OE-3DIS methods, we follow [46] to leverage an adaptive protocol of standard AP score calculation. To tackle this new and challenging task, **we introduce a novel method along with our proposed three established baselines** that leverage OV-3DIS techniques and Multimodal Large Language Models (MLLMs). In the first two baselines, class names are either predefined from extensive vocabularies or extracted using image taggers. The third baseline employs 2D visual tokens, obtained from a pretrained 2D CLIP encoder, which are then processed through a Visual Token Aggregation mechanism before being fed into an LLM to predict class names. Finally, our proposed method takes this further by consistently aggregating 2D visual tokens from multi-view images, transforming them into dense 3D point cloud tokens that enable real-time querying. Notably, our point-wise visual token lifting approach, which feeds these tokens into an LLM, achieves the best performance across multiple benchmarks, matching state-of-the-art OV-3DIS methods that rely on ground-truth class labels. Furthermore, both our proposed method and designed baselines are entirely training-free, utilizing only pretrained 2D vision encoders (e.g., CLIP [43]) and pretrained LLMs (e.g., Vicuna [10]). The training-free approach effectively mitigates the prevalent issue of insufficient training data that hampers many existing 3D-LLM techniques.

To evaluate the performance of these methods, we conduct experiments on open-ended versions of two prominent 3D instance segmentation (3DIS) datasets: ScanNet200 [49] and ScanNet++ [56]. The results underscore the efficacy of our chosen approach over alternative baselines, demonstrating performance levels comparable to OV-3DIS methods that rely on ground-truth class names. Specifically, in ScanNet200, our approach attained an AP of 16.0, contrasting with the 22.2 AP achieved by Open3DIS, currently recognized as the state-of-the-art in OV-3DIS. However, our approach is superior in ScanNet++, where it outperforms Open3DIS by a significant margin (18.4 vs. 13.1 in AP).

In summary, the contributions of our work are:

1. We propose Open-Ended 3D Point Cloud Instance Segmentation (OE-3DIS), a task that segments 3D point clouds by instances and generates class names without predefined labels.
2. We establish solid baselines for OE-3DIS, including leveraging OV-3DIS methods and Multimodal Large Language Models (MLLMs).
3. We present a training-free OE-3DIS method that lifts 2D visual tokens to 3D and utilizes pretrained Multimodal LLMs to output the final object classes.

2. Related Work

3D instance segmentation (3DIS) methods such as Mask3D [50], ISBNet [37], PointGroup [30], and others [2, 34, 47, 48] cluster a point cloud scene into 3D instance masks of classes predefined in the training set. These methods utilize a 3D Convolutional backbone [11, 12, 15] to extract semantic information from the 3D scene. Subsequently, they employ either Dynamic Convolution-based [25] or Grouping-based [52] modules to generate 3D instance masks. Recently, some approaches have adopted techniques to back-project 2D information aggregated from multiple views onto the 3D point cloud to create an ensemble of 3D point cloud features [23, 28, 41, 42]. These 2D-derived features contain rich semantic information, while those derived from 3D capture the geometrical structure of 3D objects. Combined, they supervise a 3D instance decoder to refine segmentation masks. However, these methods are closed-vocab or cannot segment new classes in testing, limiting the capability to understand new 3D scenes.

Open-vocabulary 3D instance segmentation (OV-3DIS) aims at segmenting 3D objects of classes newly provided in testing. To provide 3D proposals for object recognition, OpenMask3D [51] and Lowis3D [21] employ 3DIS networks [37, 50] to generate class-agnostic 3D proposals, while SAI3D [57], MaskClustering [55], OVIR [35] and Any3DIS [38] utilize 2D segmenter for producing masks for each view and lift these masks to 3D. While 3DIS networks excel in capturing large geometrical structures, they often struggle to detect rare and small-shaped objects. Conversely, 2D segmenters are adept at focusing on small regions but face challenges in maintaining object consistency when lifting to the 3D point cloud. Open3DIS [39] addresses these limitations by combining both 3D and 2D branches, resulting in superior results. This approach effectively captures rare and small objects while preserving the 3D geometrical structures of large objects using superpoint-level masks. While OV-3DIS is useful in some scenarios, the constraint of a predefined vocabulary set in inference requires human intervention, hindering very autonomous agents.

3D scene understanding with Large Language Models (LLMs). Utilizing LLMs for 3D scene understanding focuses on how objects are aligned, their directions, and their locations based on textual questions within 3D environments. This approach emphasizes the spatial aspects of language understanding in three dimensions of data. Previous works [1, 3, 4, 26, 27] have contributed to providing 3D spatial data with language for various applications, including 3D instance and scene captioning [6–8, 26, 27, 53, 64], 3D visual answering questions [3, 6, 19, 26, 27, 36, 40, 64], 3D visual grounding [1, 4, 26, 27, 54, 64] and supporting embodied AI tasks like planning and reasoning [6, 26, 27, 64].

Open-ended 2D Image Understanding is an emerging task that addresses the need to recognize objects without predefined class names during training or testing. There is scant work on this task, with existing research primarily focusing on image classification [14, 29, 61], object detection [13], and instance segmentation [58–60]. Standard 2D Multimodal LLMs (2D MLLMs), such as LLAVA [33], consist of a frozen vision encoder, a projector, and an LLM module. These models typically finetune either (1) the linear projector and the LLM or (2) a complex Q-Former projector. However, applying these methods for 3D scene understanding (3D-LLMs) is challenging due to the lack of sufficient 3D data and text description pairs to effectively train 3D-LLMs. In this paper, we present a novel approach to leveraging pretrained 2D MLLMs for OE-3DIS.

OmniScient Model (OSM) [59] is a recently proposed pretrained 2D Multimodal LLM for Open-ended 2D Instance Segmentation, which serves as the foundation for our proposed method. OSM comprises three main modules: a visual encoder, a MaskQ-Former, and a Large Language Model (LLM). The visual encoder is a pretrained EVA-CLIP [22], a variant of the CLIP model [43], which extracts high-resolution visual features using a sliding-window scheme and incorporates global positional embeddings to preserve spatial information. The MaskQ-Former, a customized version of Q-Former [18] designed to focus on the mask region rather than the entire image, converts visual features into fixed-length visual tokens. These tokens are then input into the Vicuna LLM [10]. The LLM processes these tokens to answer the question, "What is in the segmentation mask?" by outputting the object name.

In OSM, only the MaskQ-Former is trained to align visual features with the visual tokens for the LLM, while the visual encoder and LLM remain unchanged. The training datasets are large, including COCO [32], LVIS [24], ADE20K [62], and Cityscapes [16]. This setup demonstrates a strong capability for recognizing objects without predefined class names in 2D images, inspiring us to extend this approach to 3D scene understanding.

3. Methodology

3.1. Problem Statement

Given a 3D point cloud scene $\mathbf{P} = \{\mathbf{p}_i\}_{i=1}^{N} \in \mathbb{R}^{N\times 6}$ consisting of N points with xyz coordinates and associated rgb colors, along with T RGB-D frames with color images $\{\mathbf{I}_t\}_{t=1}^{T}$ and depth ones $\{\mathbf{D}_t\}_{t=1}^{T}$, where $\mathbf{I}_t \in \mathbb{R}^{H\times W\times 3}, \mathbf{D}_t \in \mathbb{R}_{+}^{H\times W}$, we aim to segment all K object binary masks $\{\mathbf{m}_k\}_{k=1}^{K}, \mathbf{m}_k \in \{0,1\}^N$ and their associated class names $\{l_k\}_{k=1}^{K}$ without giving any predefined class names in inference. Of course, we do need GT class names during evaluation to assess our method's performance.

The information includes the intrinsic $\mathbf{\Gamma} \in \mathbb{R}^{3\times 3}$ and the extrinsic $[\mathbf{R}|\mathbf{v}]_t \in \mathbb{R}^{3\times 4}$. In this context, H and W represent the height and width of the image, respectively. The matrix $\mathbf{R}$ is a 3D rotation matrix, while $\mathbf{v}$ is a 3D translation vector. This composite matrix, which combines rotation and translation, converts coordinates from the global frame of the point cloud to the camera's frame at view t.

3.2. Evaluation Metrics

To evaluate open-ended object detection or instance segmentation, where predicted class names may be similar but not exactly the same as ground-truth (GT) class names, prior work [46] proposed a label reassignment technique. This method uses text encoders (e.g., CLIP [43], BERT [20], Sentence Transformer [44] to encode both the predicted and GT class names for each scene. It then matches each predicted class name to its closest GT class name based on cosine similarity. After this matching, the standard **AP score** is used to evaluate performance.

3.3. Proposed Baselines

Since OE-3DIS is very new and challenging, we focus our efforts on investigating prominent baselines. These baselines are illustrated in Fig. 2. They require a list of class-agnostic 3D mask proposals pre-extracted from Open3DIS [39] with the DETIC 2D segmenter.

Large-vocab approach (Fig. 2 - Left): We start with a simple OE-3DIS baseline by using a large vocabulary of 21K common classes introduced by DETIC [63] as predefined class names for OV-3DIS methods like Open3DIS [39] and OpenMask3D [51]. However, this approach fails to achieve robust class prediction. This is because the fixed large vocabulary set contains multiple synonyms, resulting in uninformative class predictions after the Softmax operation.

Image Tagging approach (Fig. 2 - Middle): To reduce the number of classes, we leverage image-tagging techniques such as RAM++ [29] to obtain only relevant class names per scene. Specifically, for each input view, a set of image tags is generated and then combined across all processed input views. The resulting unified tag set serves as the vocabulary for OV-3DIS. However, these methods often produce

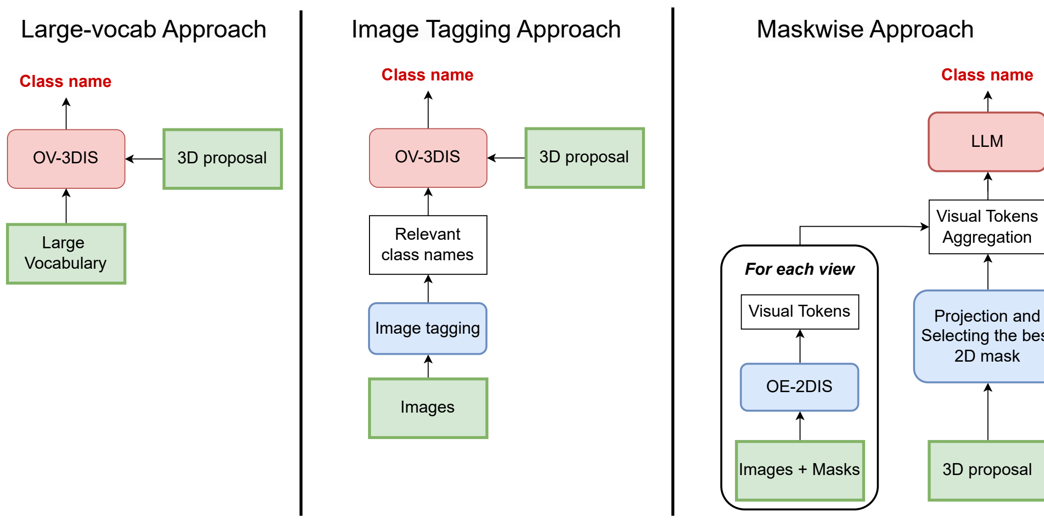

Figure 2. Our baselines: Large-vocab (Left), Image tagging (Middle), and Maskwise (Right).

inconsistent class names across views, leading to redundant and similar class names.

Maskwise approach (Fig. 2 - Right): To tackle the inconsistency in class names across views, we first apply an OE-2DIS method, such as OSM [59], to each view to obtain a list of 2D masks and their predicted fixed-length visual tokens. For each 3D mask proposal, we project it onto a view and associate it with the best-matched 2D mask based on IoU to obtain its 2D fixed-length visual tokens. The 3D visual tokens for the 3D mask are then aggregated by averaging the 2D visual tokens across views, which are subsequently input into a pretrained LLM to obtain the final class names. However, this approach relies on matching 3D proposals with 2D masks, which is often misaligned due to segmentation and depth map imperfections. A 3D mask can project onto multiple 2D masks, and selecting only the best match may discard valuable information.

3.4. Our Approach

To address the above limitations, we propose a method for producing pointwise 3D visual tokens, as illustrated in Fig. 3. First, we generate class-agnostic 2D instance segmentation masks for all views using class-agnostic 2D segmenters such as DETIC [63] and SAM [31]. Next, we lift 2D masks into 3D masks using Open3DIS [39]. Simultaneously, these 2D masks, along with their corresponding RGB images, are used to extract 2D visual tokens from an MLLM like OSM [59]. We then lift the resulting 2D visual tokens $\mathbf{F}^{2D}$ into 3D visual tokens to obtain pointwise 3D visual tokens $\mathbf{F}^{3D}$. Finally, for each 3D proposal mask, we query the 3D visual tokens associated with this proposal by aggregating the pointwise lifted 3D visual tokens, forming the final tokens $\mathbf{f}^{3D}$ for input to the LLM. This approach takes into account the depth and geometric structure of 3D objects via lifting, resulting in more robust visual tokens and a unified, densely-featured point cloud that can be queried instantly at test time. Subsequently, we will focus on our pointwise visual tokens lifting and aggregation.

Concretely, first, the correspondence of a 3D point $\mathbf{p}_i(x, y, z) \in \mathbf{P}$ with its 2D projection (u, v) in view t is:

$$d_{i,t} \cdot \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}_t = \mathbf{\Gamma} \cdot [\mathbf{R}|\mathbf{c}]_t \cdot \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \tag{1}$$

where $d_{i,t}$ is the projected depth of point i to frame t.

Next, for each view t, we extract the 2D visual tokens $\mathbf{F}_t^{k,\text{2D}} \in \mathbb{R}^{E \times C}$, where E, C are the number of visual tokens and feature dimensions, for each 2D mask k from the MaskQ-Former module of OSM [59]. Furthermore, we denote $\lambda_{t,i}^k = \{0, 1\}$ as the visibility value indicating a point i is visible in mask k of view t, $\lambda_t^k = \{0, 1\}^N$. We set the visibility value $\lambda_{t,i}^k$ of any points whose pixel projections fall outside the k-th 2D mask's boundaries or the disparity between projected depth d and the collected depth $\mathbf{D}$ exceeds a defined depth threshold τ_{depth}, or $|d_{i,t} - \mathbf{D}_t[\lfloor u_i \rfloor, \lfloor v_i \rfloor]| > \tau_{depth}$, to 0. Then, we accumulate 3D visual tokens $\mathbf{F}^{\text{3D}} \in \mathbb{R}^{N \times E \times C}$, from every 2D mask k and compute for the frequency $\mathbf{r}^{\text{3D}} \in \mathbb{N}^N$ of every view as follows:

$$\mathbf{F}^{\text{3D}} = \sum_{t,k} \lambda_t^k * \mathbf{F}_t^{k,\text{2D}}, \qquad \mathbf{r}^{\text{3D}} = \sum_{t,k} \lambda_t^k, \tag{2}$$

where $*$ is the outer product operation. The normalized pointwise 3D visual tokens are then obtained as follows:

$$\bar{\mathbf{F}}_i^{\text{3D}} = \begin{cases} \mathbf{F}_i^{\text{3D}} / \mathbf{r}_i^{\text{3D}} & \text{if } \mathbf{r}_i^{\text{3D}} > 0 \\ \mathbf{0} & \text{otherwise} \end{cases}. \tag{3}$$

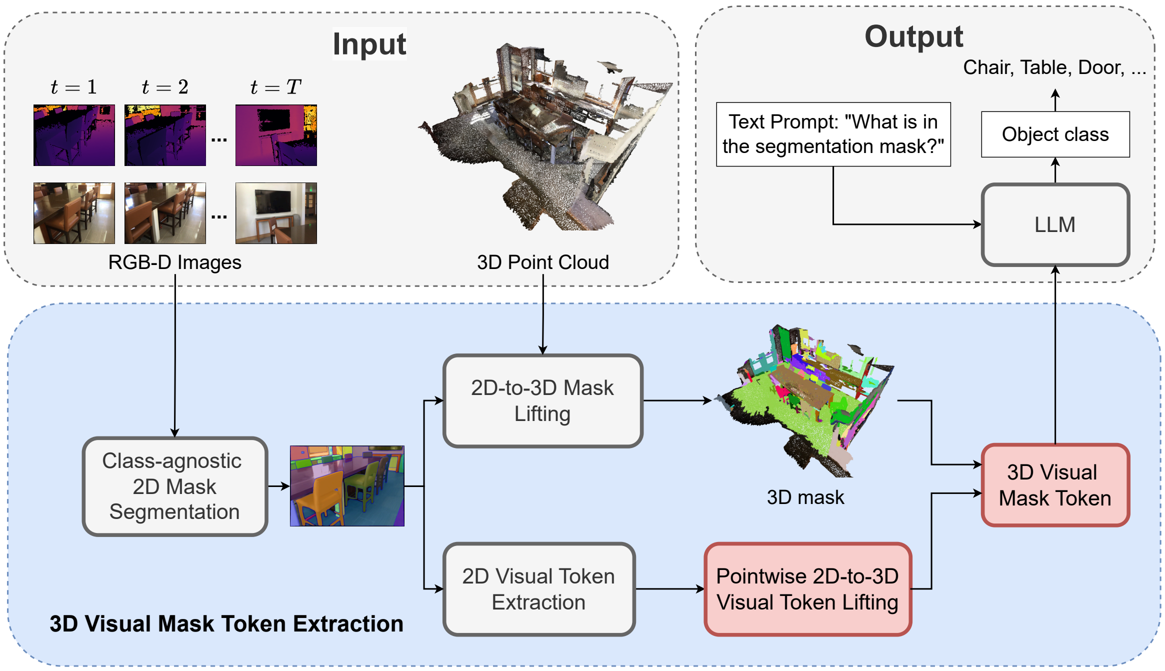

Figure 3. **Overview of our approach**. First, we generate class-agnostic 2D instance segmentation masks for all views using segmenters like DETIC [63] and SAM [31], and lift these 2D masks into 3D masks using Open3DIS [39]. Simultaneously, the 2D masks and their corresponding RGB images are used to extract 2D visual tokens from an MLLM like OSM [59], which are then lifted into pointwise 3D visual tokens. Finally, for each 3D proposal mask, we aggregate the pointwise 3D visual tokens to form the final tokens for input to the LLM to predict final class names.

Finally, for each 3D mask $\mathbf{m}_k$, we weighted average the visual tokens of all its points by their frequency to obtain the 3D visual tokens $\mathbf{f}_k^{3D}$ for that mask, which are then used as input to the LLM to predict the final class, as follows.

$$\mathbf{f}_k^{3D} = \frac{\sum_{i \in \mathbf{m}_k} \bar{\mathbf{F}}_i^{3D} \cdot \mathbf{r}_i^{3D}}{\sum_{i \in \mathbf{m}_k} \mathbf{r}_i^{3D}}. \tag{4}$$

4. Experimental Results

Datasets: We conducted experiments to assess the performance of the baselines and our proposed method using two common 3DIS datasets: ScanNet200 [49] and ScanNet++ [56]. The *ScanNet200* dataset builds on the original ScanNet [17] by expanding its semantic categories from 20 to 200. Its instance segmentation benchmark includes 1,201 training scenes and 312 validation scenes with 198 object categories, significantly enriching the vocabulary and enhancing its capability for detailed 3D semantic and instance segmentation. The *ScanNet++* dataset was recently introduced, featuring up to 1,659 semantic categories, with 360 training scenes and 50 validation scenes. Given the large number of classes, we follow the standard 3DIS evaluation protocol on ScanNet++ and evaluate only the most common 100 object categories. This dataset offers a much denser 3D point cloud scene representation, making it the most challenging dataset for 3D understanding.

Evaluation metrics: We assess OE-3DIS using the AP score of reassignment of the label (detailed in Sec. 3.2). For ScanNet200, we also report AP_h (head), AP_c (common), and AP_t (tail). For ScanNet++, we include the recall rate (RC) and the average recall rate (AR). We note that the OV-3DIS methods adopt a specific AP score calculation protocol by assigning a confidence score of 1.0 to each 3D proposal. Similarly, we follow the same evaluation protocol as the Fully-sup 3DIS by ranking 3D proposals according to their confidence scores. In the context of OE-3DIS, our approach utilizes the confidence scores generated by the LLM, while potential baselines employ the CLIP score.

Implementation details: Following Open3DIS [39], we generate class-agnostic 3D proposals from ISBNet [37] pretrained on ScanNet200 or by lifting 2D masks to 3D using Detic [63]. For RAM++ [29], we employ the Swin-L model, trained on a 14-million image dataset with an image size of 384px and a tagging threshold of 0.68. For LLM, we leverage Vicuna-7B [9], fine-tuned for open-ended 2D Instance Segmentation [59].

4.1. Comparison with Baselines

We compare our approach to the proposed baselines using the OE-3DIS setting on the ScanNet200 and ScanNet++ datasets in Tab. 1. For reference, we also present results from OV-3DIS methods, including OpenMask3D [51] and

Setting	Method	ScanNet200						ScanNet++					
		AP	AP_{50}	AP_{25}	AP_h	AP_c	AP_t	AP	AP_{50}	AP_{25}	AR	RC_{50}	RC_{25}
Fully-sup 3DIS	Mask3D [50]	26.9	36.2	41.4	39.8	21.7	17.9	8.9	14.6	38.9	-	-	-
	ISBNet [37]	24.5	32.7	37.6	38.6	20.5	12.5	16.7	29.7	21.0	-	-	-
OV-3DIS	OpenMask3D [51]	15.4	19.9	23.1	17.1	14.1	14.9	2.0	2.7	3.4	4.6	8.3	12.4
	Open3DIS [39]	23.7	29.4	32.8	27.8	21.2	21.8	13.1	20.8	24.6	22.1	33.9	39.1
Large-Vocab	21K DETIC classes (Ours)	8.5	11.7	13.1	9.9	7.2	8.3	7.3	11.9	15.2	13.3	20.3	23.6
Image-Tagging	RAM++ [29] (Ours)	10.7	14.3	16.0	11.6	11.0	9.3	9.1	15.5	19.1	16.0	24.8	28.7
Maskwise	OSM [59] (Ours)	14.4	19.8	23.9	18.9	13.5	10.2	16.3	24.8	29.0	22.2	32.0	36.0
Pointwise	Ours	**16.0**	**22.0**	**24.7**	**20.0**	**14.3**	**13.2**	**18.4**	**29.4**	**33.6**	**23.3**	**35.2**	**39.3**

Table 1. Comparative results on the ScanNet200 and ScanNet++ datasets. Shaded text indicates a reference method, not a direct comparison. '-' indicates results are not provided. The best results are in **bold**.

Open3DIS [39], as well as fully-supervised methods like ISBNet [37] and Mask3D [50] for ScanNet200; and PointGroup [30] and SoftGroup [52] for ScanNet++.

For Scannet200: We obtain class-agnostic 3D proposals from two sources: 3D masks from a 3DIS network such as ISBNet [37], and 2D Lift 3D masks from Open3DIS [39]. Our proposed approach outperforms other baselines in the AP score. Furthermore, the performance progression of the baselines, in the specified order, clearly justifies the motivation behind each baseline compared to its predecessor, as discussed in Sec. 3.3. Interestingly, our approach also surpasses OpenMask3D [51] (16.0 vs. 15.4 in AP), even though OpenMask3D utilizes provided class names. This indicates that, in some cases, we can achieve OV-3DIS without relying on provided class names.

For ScanNet++: Due to the extensive scale and vast array of classes in ScanNet++, the performance of 3D mask results from ISBNet is inadequate. Consequently, we solely rely on the utilization of 2D Lift 3D masks from Open3DIS [39]. We notice a consistent trend akin to the results observed in ScanNet200. Particularly noteworthy is the significant outperformance of our approach compared to OV-3DIS methods or even fully supervised 3DIS, as evidenced by higher AP scores. This underscores the promising application of OE-3DIS in navigating complex 3D scene.

Qualitative comparison: In Fig. 4, both our proposed approach and baseline methods effectively assign class names to 3D proposals under OE-3DIS settings. The first column of qualitative results on ScanNet200 demonstrates that our architectures correctly predict class labels for all 3D proposals. The second column highlights notable differences, with our method accurately identifying the 'painting' object as a 'wall painting', whereas the baselines produce less precise labels. In the final column, our detailed 3D mask proposals surpass the granularity of the ground-truth annotations, enabling accurate recognition of classes absent from ScanNet++'s vocabulary, such as 'photocopier'. This illustrates how OE-3DIS allows the model to comprehensively understand a 3D scene without restricting to fixed categories.

Technique	AP	AP_{50}	AP_{25}	AP_h	AP_c	AP_t
L2 Norm (Open3DIS)	5.7	8.2	9.8	6.8	4.9	5.2
Memory Fusion (OVIR-3D)	8.4	11.6	14.4	7.2	7.0	11.6
Max	6.3	8.4	10.2	8.5	4.7	5.6
Random	13.2	18.9	21.6	17.7	12.3	10.0
Mean	14.5	20.1	22.6	18.2	13.1	11.2
Weighted Average (Ours)	**16.0**	**22.0**	**24.7**	**20.0**	**14.3**	**13.2**

Table 2. Ablation on point aggregation techniques

3D Proposals	2D Masks	AP	AP_{50}	AP_h	AP_c	AP_t
3D masks	DETIC	11.7	16.0	16.3	10.3	8.1
2D Lift 3D masks	DETIC	11.7	18.6	11.3	12.1	11.8
3D + 2D Lift 3D masks	DETIC	**16.0**	**22.0**	**20.0**	14.3	**13.2**
3D + 2D Lift 3D masks	SAM	15.4	20.5	19.2	**14.8**	11.7

Table 3. Study on different types 3D proposals.

Text Encoder	AP	AP_{50}	AP_{25}	AP_h	AP_c	AP_t
BERT [20]	13.5	19.0	21.2	17.6	11.6	11.0
CLIP [43]	15.9	**22.0**	24.5	**20.3**	13.7	**13.3**
Sentence Transformer [44]	**16.0**	**22.0**	**24.7**	20.0	**14.3**	13.2

Table 4. Study on different text encoders in evaluation metrics

4.2. Ablation Study

To study many design choices of our pointwise approach, we intensively carry out ablation study on the ScanNet200 [17] dataset.

Study on different point feature aggregation techniques (Eq. (4)) is shown in Tab. 2. We evaluate four techniques for combining point features from a given 3D mask: L2-norm, max, random (randomly selecting one point), mean, and weighted average (our proposed operation). The weighted

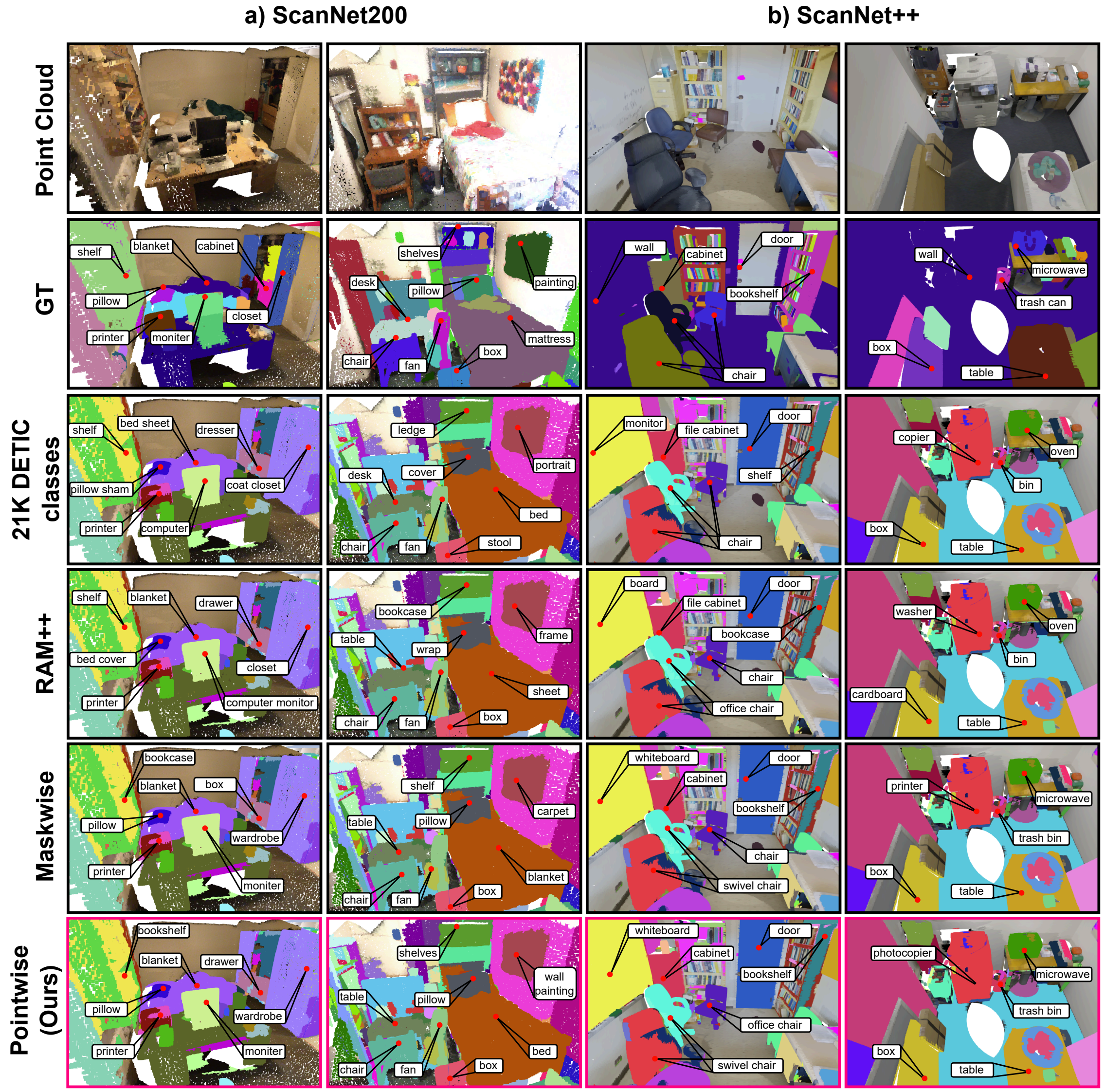

Figure 4. Qualitative examples are provided for ScanNet200 [17] (first two columns) and ScanNet++ [56] (last two columns). Both our baselines and approach yield notably good results, particularly in terms of accurately identifying class names even though they do not match exactly the GT classes.

Modules	# Params(M)	FLOPs (G)	Time (s)
Pw. CLIP extract (CLIP)	427.94	191.11	97
QFormer	185.66	2.74	250
LLM (Vicuna 7B)	6,738.42	850.01	50
RAM++	329.49	104.43	40
Generate 3D proposals	N/A	357.99	152
Pw. Visual Token Lift (PVTL)	N/A	4.915	314

Table 5. Expected latency, FLOPs, and runtime of each module for a 3D scene. Pw. denotes 'Pointwise'. Generating 3D proposals from 2D and Pw. visual token lifting are non-parametric modules

average technique achieved the highest performance with an AP score of 16.0, outperforming the other methods. Consequently, we utilized the weighted average technique in all our experiments.

Study on different types 3D proposals. As reported in Tab. 3, combining 3D proposals from both 3D masks and 2D Lift 3D masks yielded the most favorable results. This outcome validates our choice of 3D proposals for the ScanNet200 dataset. Additionally, the 2D Lift 3D proposals obtained from DETIC demonstrated slightly superior performance compared to those from SAM. This observation diverges from the methodology adopted in Open3DIS [39], where provided class names are utilized to employ Grounded-SAM [45] instead of SAM [31], as in our OE-

Method	AP	Total Time (s)	RAM++	Generate 3D proposals	CLIP	QFormer	PVTL	LLM
Open3DIS	N/A	249		✓	✓			
Large Vocab	8.5	249		✓	✓			
Image Tagging	10.7	289	✓	✓	✓			
Maskwise	14.4	452		✓		✓		✓
Pointwise	16.0	766		✓		✓	✓	✓

Table 6. Expected runtime for our proposed baselines and methods with the Open3DIS for a sample 3D scene

Text Prompt	AP
“What is in the segmentation mask? Assistant:”	**16.0**
“Can you describe what is in the segmentation mask region? Assistant:”	15.3
“What can you see in the segmentation mask region? Assistant:”	15.9
“Could you use a few words to describe what is in the segmentation mask? Assistant:”	15.2
“What is this segmentation mask? Assistant:”	15.8

Table 7. Study on different text prompts used to query the LLM.

3DIS scenario.

Study on different text encoders for evaluation metrics is described in Tab. 4.We evaluate three text decoders: BERT [20], CLIP [43], and Sentence Transformer [44]. Among these, Sentence Transformer embeddings achieve the highest AP scores, outperforming CLIP and BERT, which yield comparatively lower scores. We also observe that CLIP embeddings exhibit greater similarity among themselves compared to those from BERT. Moreover, Sentence Transformer embeddings are better suited for capturing complex sentence-level descriptions rather than focusing solely on individual class names. Therefore, we recommend Sentence Transformer as our preferred text encoder.

Latency analysis of our baselines and proposed method is presented in Tab. 5 (excluding the runtime for generating 2D proposals). Additionally, Tab. 6 compares the total runtime across our baselines, proposed methods, and the state-of-the-art 3DIS approach, Open3DIS. As shown, our method prioritizes accuracy at the cost of increased runtime, meaning higher accuracy comes with slower execution.

Study on different text prompts used to query the LLM: We experiment with various input text prompts to query the LLM for class names. Tab. 7 demonstrates that altering the prompt subtly affects the model’s accuracy. We select the prompt “What is in the segmentation mask? Assistant:” from the variants for our approach.

5. Discussion and Conclusion

Limitations: The motivation behind our proposed Open-ended 3D Instance Segmentation (OE-3DIS) arises from limitations in current Open-Vocabulary 3D Point Cloud Instance Segmentation (OV-3DIS) methods, which still depend on a predefined set of class names during testing. This constraint is impractical in scenarios lacking prior knowledge of class names, such as a robot navigating unfamiliar environments. Although our method advances open-ended 3D scene understanding, it still faces notable limitations. Firstly, our approach heavily depends on 2D visual tokens extracted from a pretrained OSM, which itself is trained on instance segmentation datasets containing only a limited number of classes. Consequently, this restricts the model’s capability to recognize an extensive range of classes in truly open-world contexts. Secondly, the performance of OE-3DIS relies significantly on class-agnostic 3DIS methods, whose effectiveness in turn depends on the accuracy of 3D representations and the quality of 2D-to-3D mapping, including factors like camera calibration and depth image quality.

Conclusion: We have introduced Open-Ended 3D Point Cloud Instance Segmentation (OE-3DIS), which generates 3D masks and object class names without predefined labels during testing. We have explored baselines using OV-3DIS methods and MLLMs, and introduced a pointwise training-free approach leveraging OSM. Experiments on ScanNet200 and ScanNet++ show our approach’s superior performance, notably outperforming Open3DIS (SOTA on OV-3DIS) on ScanNet++ without ground-truth class names.

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas J. Guibas. ReferIt3D: Neural listeners for fine-grained 3d object identification in real-world scenes. In *16th European Conference on Computer Vision (ECCV)*, 2020. 3

[2] Salwa Al Khatib, Mohamed El Amine Boudjoghra, Jean La-

houd, and Fahad Shahbaz Khan. 3d instance segmentation via enhanced spatial and semantic supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 541–550, 2023. 2

[3] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[4] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. *16th European Conference on Computer Vision (ECCV)*, 2020. 3

[5] Shaoyu Chen, Jiemin Fang, Qian Zhang, Wenyu Liu, and Xinggang Wang. Hierarchical aggregation for 3d instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15467–15476, 2021. 1

[6] Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding, reasoning, and planning, 2023. 3

[7] Sijin Chen, Hongyuan Zhu, Xin Chen, Yinjie Lei, Gang Yu, and Tao Chen. End-to-end 3d dense captioning with vote2cap-detr. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11124–11133, 2023.

[8] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2cap: Context-aware dense captioning in rgb-d scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3193–3203, 2021. 3

[9] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 5

[10] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6, 2023. 2, 3

[11] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8958–8966, 2019. 2

[12] Christopher Choy, Junha Lee, Rene Ranftl, Jaesik Park, and Vladlen Koltun. High-dimensional convolutional networks for geometric pattern recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2

[13] Lin Chuang, Jiang Yi, Qu Lizhen, Yuan Zehuan, and Cai Jianfei. Generative region-language pretraining for open-ended object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[14] Alessandro Conti, Enrico Fini, Massimiliano Mancini, Paolo Rota, Yiming Wang, and Elisa Ricci. Vocabulary-free image classification, 2023. 3

[15] Spconv Contributors. Spconv: Spatially sparse convolution library. https://github.com/traveller59/spconv, 2022. 2

[16] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 3

[17] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 5, 6, 7

[18] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 3

[19] Alexandros Delitzas, Maria Parelli, Nikolas Hars, Georgios Vlassis, Sotirios Anagnostidis, Gregor Bachmann, and Thomas Hofmann. Multi-clip: Contrastive vision-language pre-training for question answering tasks in 3d scenes. *arXiv preprint arXiv:2306.02329*, 2023. 3

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3, 6, 8

[21] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Lowis3d: Language-driven open-world instance-level 3d scene understanding, 2023. 2

[22] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369, 2023. 3

[23] Haoyu Guo, He Zhu, Sida Peng, Yuang Wang, Yujun Shen, Ruizhen Hu, and Xiaowei Zhou. Sam-guided graph cut for 3d instance segmentation. *arXiv preprint arXiv:2312.08372*, 2023. 2

[24] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019. 3

[25] Tong He, Chunhua Shen, and Anton van den Hengel. Dyco3d: Robust instance segmentation of 3d point clouds through dynamic convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 354–363, 2021. 1, 2

[26] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *NeurIPS*, 2023. 3

[27] Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024. 3

[28] Rui Huang, Songyou Peng, Ayca Takmaz, Federico Tombari, Marc Pollefeys, Shiji Song, Gao Huang, and Francis Engelmann. Segment3d: Learning fine-grained class-agnostic 3d segmentation without manual labels. *arXiv*, 2023. 1, 2

[29] Xinyu Huang, Yi-Jie Huang, Youcai Zhang, Weiwei Tian, Rui Feng, Yuejie Zhang, Yanchun Xie, Yaqian Li, and Lei Zhang. Open-set image tagging with multi-grained text supervision, 2023. 3, 5, 6

[30] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4867–4876, 2020. 1, 2, 6

[31] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 4, 5, 7

[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 3

[33] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 3

[34] Jiahao Lu, Jiacheng Deng, Chuxin Wang, Jianfeng He, and Tianzhu Zhang. Query refinement transformer for 3d instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 18516–18526, 2023. 2

[35] Shiyang Lu, Haonan Chang, Eric Pu Jing, Abdeslam Boularias, and Kostas Bekris. Ovir-3d: Open-vocabulary 3d instance retrieval without training on 3d data. In *7th Annual Conference on Robot Learning*, 2023. 1, 2

[36] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. In *International Conference on Learning Representations*, 2023. 3

[37] Tuan Duc Ngo, Binh-Son Hua, and Khoi Nguyen. Isbnet: a 3d point cloud instance segmentation network with instance-aware sampling and box-aware dynamic convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13550–13559, 2023. 1, 2, 5, 6

[38] Phuc Nguyen, Minh Luu, Anh Tran, Cuong Pham, and Khoi Nguyen. Any3dis: Class-agnostic 3d instance segmentation by 2d mask tracking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3636–3645, 2025. 2

[39] Phuc D. A. Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3dis: Open-vocabulary 3d instance segmentation with 2d mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 2, 3, 4, 5, 6, 7

[40] Maria Parelli, Alexandros Delitzas, Nikolas Hars, Georgios Vlassis, Sotirios Anagnostidis, Gregor Bachmann, and Thomas Hofmann. Clip-guided vision-language pre-training for question answering in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5606–5611, 2023. 3

[41] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[42] Gilles Puy, Spyros Gidaris, Alexandre Boulch, Oriane Siméoni, Corentin Sautier, Patrick Pérez, Andrei Bursuc, and Renaud Marlet. Three pillars improving vision foundation model distillation for lidar. In *CVPR*, 2024. 2

[43] 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. 2, 3, 6, 8

[44] Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2020. 3, 6, 8

[45] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 7

[46] Pitchaporn Rewatbowornwong, Nattanat Chatthee, Ekapol Chuangsuwanich, and Supasorn Suwajanakorn. Zero-guidance segmentation using zero segment labels. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 2, 3

[47] Damien Robert, Hugo Raguet, and Loic Landrieu. Efficient 3d semantic segmentation with superpoint transformer. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 2

[48] Damien Robert, Hugo Raguet, and Loic Landrieu. Scalable 3d panoptic segmentation as superpoint graph clustering. *Proceedings of the IEEE International Conference on 3D Vision*, 2024. 2

[49] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2, 5

[50] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d for 3d semantic instance segmentation. In *International Conference on Robotics and Automation (ICRA)*, 2023. 1, 2, 6

[51] Aycca Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann.

OpenMask3D: Open-Vocabulary 3D Instance Segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 1, 2, 3, 5, 6

[52] Thang Vu, Kookhoi Kim, Tung M. Luu, Xuan Thanh Nguyen, and Chang D. Yoo. Softgroup for 3d instance segmentation on 3d point clouds. In *CVPR*, 2022. 1, 2, 6

[53] Heng Wang, Chaoyi Zhang, Jianhui Yu, and Weidong Cai. Spatiality-guided transformer for 3D dense captioning on point clouds. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2022. 3

[54] Yanmin Wu, Xinhua Cheng, Renrui Zhang, Zesen Cheng, and Jian Zhang. Eda: Explicit text-decoupling and dense alignment for 3d visual grounding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[55] Mi Yan, Jiazhao Zhang, Yan Zhu, and He Wang. Maskclustering: View consensus based mask graph clustering for open-vocabulary 3d instance segmentation. *arXiv preprint arXiv:2401.07745*, 2024. 1, 2

[56] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023. 2, 5, 7

[57] Yingda Yin, Yuzheng Liu, Yang Xiao, Daniel Cohen-Or, Jingwei Huang, and Baoquan Chen. Sai3d: Segment any instance in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 2

[58] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023. 3

[59] Qihang Yu, Xiaohui Shen, and Liang-Chieh Chen. Towards open-ended visual recognition with large language model. In *arxiv: 2311.08400*, 2023. 3, 4, 5, 6

[60] Yuqian Yuan, Wentong Li, Jian Liu, Dongqi Tang, Xinjie Luo, Chi Qin, Lei Zhang, and Jianke Zhu. Osprey: Pixel understanding with visual instruction tuning, 2023. 3

[61] Kaiyu Yue, Bor-Chun Chen, Jonas Geiping, Hengduo Li, Tom Goldstein, and Ser-Nam Lim. Object Recognition as Next Token Prediction. In *Computer Vision and Pattern Recognition Conference (CVPR)*, 2024. 3

[62] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 3

[63] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022. 3, 4, 5

[64] Zhu Ziyu, Ma Xiaojian, Chen Yixin, Deng Zhidong, Huang Siyuan, and Li Qing. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *ICCV*, 2023. 3