FOLK: Fast Open-Vocabulary 3D Instance Segmentation via Label-guided Knowledge Distillation

Hongrui Wu[1]*, Zhicheng Gao[1]*, Jin Cao[2], Kelu Yao[3], Wen Shen[1], Zhihua Wei[1]†

[1]Tongji University, Shanghai, China
[2]Xi'an Jiaotong University, Xi'an, China
[3]Zhejiang Laboratory, Zhejiang University, Hangzhou, China
2152131@tongji.edu.cn, 2891379914@qq.com, xjincaox@gmail.com,
yaokelu@zhejianglab.com, wenshen@tongji.edu.cn, zhihua_wei@tongji.edu.cn

Abstract

Open-vocabulary 3D instance segmentation seeks to segment and classify instances beyond the annotated label space. Existing methods typically map 3D instances to 2D RGB-D images, and then employ vision-language models (VLMs) for classification. However, such a mapping strategy usually introduces noise from 2D occlusions and incurs substantial computational and memory costs during inference, slowing down the inference speed. To address the above problems, we propose a Fast Open-vocabulary 3D instance segmentation method via Label-guided Knowledge distillation (FOLK). Our core idea is to design a teacher model that extracts high-quality instance embeddings and distills its open-vocabulary knowledge into a 3D student model. In this way, during inference, the distilled 3D model can directly classify instances from the 3D point cloud, avoiding noise caused by occlusions and significantly accelerating the inference process. Specifically, we first design a teacher model to generate a 2D CLIP embedding for each 3D instance, incorporating both visibility and viewpoint diversity, which serves as the learning target for distillation. We then develop a 3D student model that directly produces a 3D embedding for each 3D instance. During training, we propose a label-guided distillation algorithm to distill open-vocabulary knowledge from label-consistent 2D embeddings into the student model. FOLK conducted experiments on the ScanNet200 and Replica datasets, achieving state-of-the-art performance on the ScanNet200 dataset with an AP50 score of 35.7, while running approximately 6.0× to 152.2× faster than previous methods. All codes will be released after the paper is accepted.

1 Introduction

Open-vocabulary 3D instance segmentation seeks to segment and classify unseen categories beyond the annotated label space. Recent advances in 2D open-vocabulary perception (Zhong et al. 2022; Gu et al. 2022; Kuo et al. 2022; Li et al. 2022, 2023; Wu et al. 2023b) have been driven by 2D vision-language models (VLMs), such as CLIP (Radford et al. 2021) and ALIGN (Jia et al. 2021), which learn extensive open-vocabulary knowledge during pre-training. However, in 3D scenes, the lack of large-scale 3D-text paired data hinders pretraining a 3D VLM comparable to CLIP.

To enable open-vocabulary instance segmentation in the 3D domain, most methods (Takmaz et al. 2023; Nguyen et al. 2024; Lu et al. 2023b; Boudjoghra et al. 2025; Huang et al. 2024) first project 3D instances onto multiple 2D RGB-D images. These images are then processed by 2D vision-language models (VLMs) to obtain semantic embeddings, which are used to classify the original 3D instances. However, these methods exhibit notable limitations. Specifically, relying solely on 2D images for the instance classification introduces noise due to occlusions. To mitigate these issues, existing methods often aggregate information from multiple 2D frames for each 3D instance, but this leads to considerable computational overhead and increased memory consumption, ultimately resulting in slow inference speeds.

To address the above limitations, we propose a Fast Open-vocabulary 3D instance segmentation method via Label-guided Knowledge distillation (FOLK). Our core idea is to distill the open-vocabulary knowledge of the 2D CLIP model into a 3D student model, enabling direct classification of 3D instances. Specifically, the proposed FOLK method comprises three main components: the teacher model, which generates 2D CLIP embeddings as the learning target for distillation; the student model, which produces a 3D embedding for each 3D instance; and the label-guided distillation algorithm, which distills the knowledge from 2D CLIP embeddings into the 3D embedding.

Teacher model. To ensure the quality of knowledge distillation, we aim to develop a teacher model that provides high-quality 2D embeddings. Previous methods typically employ a category-agnostic 3D backbone (e.g., Mask3D (Schult et al. 2023)) to generate 3D instance proposals from the input point cloud and select a set of 2D RGB-D images with the highest number of visible points for each 3D instance. These methods employ bounding boxes to crop instance regions from 2D images and utilize CLIP to extract 2D embeddings, representing the 3D instances. However, the 2D embeddings extracted by these methods may not fully capture the semantic information of 3D instances. First, images with the most visible points often share similar camera poses, limiting viewpoint diversity. Second, cropping instance regions frequently includes substantial background in the 2D embeddings, introducing irrelevant semantic information that degrades the quality of knowledge distillation.

*These authors contributed equally.
†Corresponding Author.

Therefore, to obtain 2D embeddings with high-quality semantic information across diverse perspectives, we introduce a multi-view selection algorithm that selects a set of RGB-D images with both the highest number of visible points and maximal viewpoint diversity for each 3D instance. Moreover, to minimize background noise during feature extraction, we follow MaskCLIP++ (Zeng et al. 2025) to extract only feature values within the instance mask region of the CLIP feature map for each image to derive the 2D embedding. Specifically, for each 3D instance, we first map its sparse point cloud to each image, obtaining a sparse 2D mask. Then, we propose a density-guided mask completion algorithm to generate dense and accurate 2D masks from sparse projected masks. Finally, we use these 2D masks to guide CLIP in extracting precise 2D embeddings.

Student model. To directly generate a 3D embedding for each 3D instance, the student model first employs a 3D backbone (Mask3D) to extract point-wise features from the input 3D point cloud. Subsequently, we propose a Vision-Language adapter module (VL-adapter) to extract 3D instance embeddings from these point-wise features and map them to the same embedding space as the teacher model. We then propose a label-guided distillation algorithm to transfer the open-vocabulary knowledge of the teacher model into the student model.

Label-guided distillation. During knowledge distillation, we use the multi-view 2D embeddings from the teacher model as the learning target of the student model. However, some 2D embeddings may be semantically inconsistent with the target 3D instance, reducing distillation quality. To address this, we propose a label-guided distillation algorithm to determine the most dominant label among the 2D embeddings of each 3D instance, and remove those with inconsistent labels that do not match this majority. Then, we optimize the contrastive loss between the semantically consistent 2D embeddings of each 3D instance and the 3D embedding of the student model, thereby distilling the open-vocabulary knowledge of the teacher model into the 3D student model. Simultaneously, we use the consistency label of each 3D instance as the pseudo-label to directly supervise the classification results of the student model, thereby enhancing its classification accuracy.

Experimental results on the ScanNet200 (Rozenberszki, Litany, and Dai 2022) dataset demonstrate the effectiveness of our method. We achieve state-of-the-art performance on metrics including AP, AP_{50}, and AP_{25}, surpassing the second-best method by 1.9%, 4%, and 5.2%, respectively. Additionally, the distilled model eliminates the need to select 2D images for classifying 3D instances during inference, achieving approximately 6.0 $\times$ faster inference speed than the second-best method.

2 Related work

Closed-vocabulary 3D instance segmentation. This task aims to segment each target instance in a 3D point cloud and classify them. Several clustering-based methods (Chen et al. 2021; Jiang et al. 2020; Han et al. 2020; Liang et al. 2021) segment instances by aggregating points with similar semantics. Other approaches (Engelmann et al. 2020; Hou, Dai, and Nießner 2019; Liu et al. 2020; Yang et al. 2019) first generate candidate instances before performing segmentation and classification. Additionally, methods like Mask3D (Schult et al. 2023) utilize a CNN backbone to extract point features, followed by a transformer-based instance mask decoder to produce 3D mask proposals with corresponding semantic labels. However, closed-set methods are limited and coupled to the datasets used during training, which restricts their development in real-world application scenarios.

Open-vocabulary 2D recognition. Recent advances in vision-language pretrained models (VLMs) have greatly impacted 2D open-vocabulary tasks. Models such as CLIP (Radford et al. 2021) and ALIGN (Jia et al. 2021), pretrained on large-scale image-text datasets (e.g., over 400 million pairs for CLIP), have acquired rich open-vocabulary knowledge, enabling recognition of a wide range of object categories. As a result, VLMs are widely adopted for 2D open-vocabulary recognition. Some methods directly use CLIP as a classifier to identify unseen categories (Bansal et al. 2018; Kuo et al. 2022; Wu et al. 2023c), while others apply knowledge distillation to transfer CLIP's capabilities into 2D detectors (Gu et al. 2022; Wu et al. 2023a), improving their generalization to unseen classes.

Open-vocabulary 3D instance segmentation. This task aims to identify unseen categories not encountered during training, offering practical value in real-world applications. Recent advances in 2D vision-language models (VLMs) have inspired several 3D methods to leverage models like CLIP (Radford et al. 2021) for classifying unseen instances. For example, OpenMask3D (Takmaz et al. 2023) uses Mask3D to generate instance proposals and applies CLIP for classification, whereas Open-YOLO 3D (Boudjoghra et al. 2025) leverages 2D object detection from multi-view RGB images to enable open-vocabulary prediction. Other methods leverage SAM (Kirillov et al. 2023) for open-vocabulary prediction. For example, SAI3D (Yin et al. 2024) and Open3DIS (Nguyen et al. 2024) propose integrating 3D superpoints (Felzenszwalb and Huttenlocher 2004) guided by predictions from SAM. However, these methods depend on 2D VLMs to extract embeddings from images, leading to the loss of 3D geometric information and slow inference due to multi-view sampling. In this paper, we propose FOLK: a fast open-vocabulary 3D instance segmentation method via label-guided knowledge distillation. FOLK transfers CLIP's knowledge into a 3D model, enabling direct embedding extraction from point clouds without requiring 2D views, thus improving efficiency.

3 Method

Given a 3D point cloud $\vec{P} \in \mathbb{R}^{P\times 3}$ with P points and a set of corresponding RGB images $\vec{I}$, [1] where each image $I_j \in \vec{I}$ has a shape of $\mathbb{R}^{3\times H\times W}$, the goal of the open-vocabulary 3D instance segmentation framework is to predict labels for all instances in the input point cloud $\vec{P}$, using information derived from the associated RGB images. In this paper, we em-

[1] While the input consists of RGB-D images, we define notation only for RGB images, as the method section does not involve depth formulations. Depth images are still used in the actual experiment.

ploy the Mask3D to process the input point cloud $\vec{P}$ to obtain a set of class-agnostic 3D instance proposals $\{\Omega_i^{\text{3D}}\}_{i=1}^{N}$. The i-th 3D instance proposal Ω_i^{3D} contains M 3D points $\{p_m^{(i)}\}_{m=1}^{M}$. We also use the Mask3D model to extract point-wise features $\{f_m^{(i)}\}_{m=1}^{M}$ of the 3D instance proposal Ω_i^{3D}.

Our main contribution lies in developing algorithms to predict high-quality labels for 3D instances. To achieve this, we introduce a knowledge distillation-based method, including a teacher model, a student model, and a label-guided distillation algorithm, as shown in Fig. 1. For the i-th instance, the proposed teacher model (Sec. 3.1) generates 2D CLIP embeddings $\vec{F}^{\text{2D},i}$ as training targets during the distillation phase, while the 3D student model (Sec. 3.2) is designed to produce the corresponding 3D instance embeddings $\vec{F}^{\text{3D},i}$. The label-guided distillation algorithm (Sec. 3.3) further distills information from both 2D CLIP embeddings $\vec{F}^{\text{2D},i}$ and the pseudo-label $l^{(i)}$ into the student model, enabling open-vocabulary 3D instance segmentation.

3.1 Teacher model design

Given the i-th 3D instance proposal Ω_i^{3D}, we follow Takmaz et al. to project 3D points from Ω_i^{3D} onto the j-th image $I_j \in \vec{I}$, generating a set of pixel coordinates $U^{i\to j} = \{(u_1, v_1), (u_2, v_2), \dots\}$ that correspond to the projections of the 3D points to the 2D pixels. These pixels collectively form a sparse binary mask $\vec{M}^{i\to j} \in \{0,1\}^{H\times W}$, meaning that for all $(u, v) \in U^{i\to j}$, $\vec{M}^{i\to j}(u, v) = 1$. Each active pixel in this mask exclusively corresponds to a visible 3D surface point in the i-th 3D instance, while a value of 0 at location (u, v) in $\vec{M}^{i\to j}$ indicates that no 3D surface point from the i-th instance was projected to that pixel. The number of projected pixels $||\vec{M}^{i\to j}||_{\text{L1-norm}}$ reflects the visibility of the 3D instance Ω_i^{3D} in the image I_j: fewer pixels indicate stronger occlusion, while more pixels suggest better visibility. To this end, Takmaz et al. proposed selecting the top-K images with the highest number of projected pixels to generate 2D CLIP embeddings. However, their method neglects the viewpoint diversity and may potentially select images from similar camera poses. To address this, we propose a *multi-view selection algorithm* that considers both visibility and viewpoint diversity.

Multi-view selection algorithm. The multi-view selection algorithm first preselects the top-K_{pre} images with the highest number of projected pixels $||\vec{M}^{i\to j}||_{\text{L1-norm}}$. The algorithm then further selects a subset of images with sufficiently diverse camera poses, thereby enhancing the diversity of 2D CLIP embeddings, as Fig. 2 shows. We first rank the images $\vec{I}$ in descending order according to the number of projected pixels, and then select the top-K_{pre} images as a preselected subset $\vec{I^{\text{pre}}} \subseteq \vec{I}$ to ensure all candidate images have sufficient visibility of the target instance. Next, for each image $I_k \in \vec{I^{\text{pre}}}$, we remove images with similar camera poses. Specifically, given each image I_k and its camera rotation matrix $R_k \in \mathbb{R}^{3\times 3}$, we compute the angular difference (Huynh 2009) with respect to the subsequent images $I_{k+1}, I_{k+2}, \dots, I_{K_{\text{pre}}}$, as follows:

$$\begin{aligned} &\forall s \in \{k+1, \dots, K_{\text{pre}}\}, \\ &\theta(R_k, R_s) = \cos^{-1}\left(\frac{\text{tr}(R_k R_s^{\top}) - 1}{2}\right) \end{aligned} \tag{1}$$

We define a threshold θ_{th} to control the allowed angular difference. If $\theta(R_k, R_s) < \theta_{\text{th}}$, then we consider that the image I_k and the image I_s have similar camera poses, thereby removing the image I_s from $\vec{I^{\text{pre}}}$. Otherwise, we retain the image I_s. Let $\vec{I}^{(i)} = \{I_k\}_{k=1}^{K}$ denote the set of images retained after applying the multi-view selection algorithm. Thus, $\vec{I}^{(i)}$ is the set of images with the highest visibility and viewpoint diversity for the i-th 3D instance. In the subsequent steps, we extract 2D embeddings for the 3D instance from this selected set of images.

After obtaining the selected subset of images $\vec{I}^{(i)}$, we cannot directly input them into the CLIP visual encoder to generate 2D CLIP embeddings for each instance, because each image may contain multiple irrelevant instances and background. To eliminate information from these irrelevant regions, previous methods such as (Takmaz et al. 2023; Nguyen et al. 2024) used a bounding box to crop the relevant region. However, rectangular bounding boxes fail to capture diverse instance shapes, still leaving irrelevant content in the cropped regions. To address this, inspired by MaskCLIP++ (Zeng et al. 2025), we use 2D instance masks to remove irrelevant information directly from the feature maps of each image extracted by the CLIP visual encoder, preserving only the target instance's information. To this end, for each image $I_k \in \vec{I}^{(i)}$, we design a *density-guided mask completion algorithm* to generate a dense 2D instance mask $\widetilde{\vec{M}}^{i\to k}$. This mask is then used to extract 2D CLIP embeddings from the image I_k related to the i-th 3D instance.

Density-guided mask completion algorithm. Given the sparse binary mask $\vec{M}^{i\to k} \in \{0,1\}^{H\times W}$, the density-guided mask completion algorithm aims to expand from each sparsely distributed pixel to its adjacent pixels, thereby forming a 2D dense mask that accurately captures the shape of the target instance. The core idea is to estimate the local pixel density around each pixel and iteratively expand the mask along directions where the density increases most rapidly. To this end, we first perform a coarse and rapid mask completion over the sparse binary mask $\vec{M}^{i\to k}$ to obtain an intermediate dense mask $\widehat{\vec{M}}^{i\to k}$. Specifically, given a set of projected pixels in $\vec{M}^{i\to k}$, we uniformly expand around each pixel (u, v) s.t. $\vec{M}_{u,v}^{i\to k} = 1$ to its neighboring region within a square window of a radius $r \in \mathbb{R}^{+}$, following an isotropic directional manner (Cantrell 2000) as follows:

$$\widehat{\vec{M}}_{u',v'}^{i\to k} = \begin{cases} 1, & \text{if } \max(|u'-u|, |v'-v|) \le r \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Next, we perform a finer mask completion to improve the coverage of pixels and obtain the final dense mask $\widetilde{\vec{M}}^{i\to k}$. Specifically, we introduce a Gaussian kernel

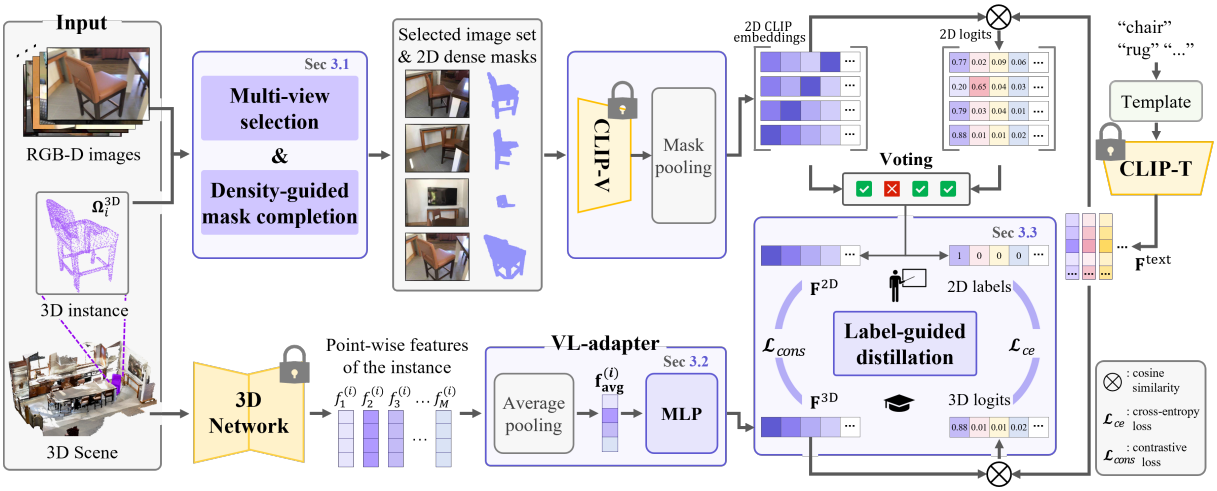

Figure 1: An overview of our method. The FOLK method consists of three main components: (1) The teacher model uses the multi-view selection algorithm and the density-guided mask completion algorithm to obtain pairs of representative images and dense masks, which are then passed into the mask-guided CLIP visual encoder to get high-quality 2D CLIP embeddings. (2) The student model uses a VL-adapter to derive 3D instance embeddings from the point-wise features. (3) The label-guided distillation algorithm transfers the knowledge from 2D CLIP embeddings into the 3D instance embeddings.

$G((u,v),(u',v'))$ with the kernel size k_s to compute the density around each pixel (u,v) in $\widetilde{\vec{M}}^{i\to k}$, as follows:

$$\rho_{u,v} = \sum_{\substack{\|(u-u',v-v')\|_2 \\ \le k_s}} G\left((u,v),(u',v')\right) \tag{3}$$

We then select pixels satisfying $\rho_{u,v} > \rho_{\text{th}}$ for the next phase of expansion. In the next phase, the mask is expanded adaptively based on local density increments to progressively fill the uncovered regions in $\widetilde{\vec{M}}^{i\to k}$. For each selected pixel (u,v) in $\widetilde{\vec{M}}^{i\to k}$, we evaluate the change in density across 8 discrete directions $\mathcal{D} = \{(-1,-1),(-1,0),\ldots,(1,1)\}$, as follows:

$$\Delta_{u,v}^{(dy,dx)} = \rho_{u+dy,v+dx} - \rho_{u,v}, \quad (dy,dx) \in \mathcal{D} \tag{4}$$

The expansion is then performed toward the top-S directions in $\mathcal{D}^* \subset \mathcal{D}$ with the highest positive increments, and if fewer than S directions exist, we expand along all of them:

$$\mathcal{D}^*\big|_{u,v} = \underset{\mathcal{A}\subseteq\mathcal{D},|\mathcal{A}|=S}{\arg\max} \sum_{(dy,dx)\in\mathcal{A}} \max\left(\Delta_{u,v}^{(dy,dx)}, 0\right) \tag{5}$$

For each direction $(dy,dx) \in \mathcal{D}^*\big|_{u,v}$, we fill the neighboring region of the pixel (u,v) within a square window of radius r. Through several iterations of this density-guided expansion, we obtain a 2D dense mask $\widetilde{\vec{M}}^{i\to k}$ that closely conforms to the distribution of pixels in the sparse binary mask $\vec{M}^{i\to k}$.

Then we pass the image $I_k \in \vec{I}^{(i)}$ of the i-th 3D instance to the CLIP visual encoder to extract the global feature map $\vec{F}_k^{\text{global},i} \in \mathbb{R}^{C\times H'\times W'}$. Meanwhile, to perform the mask pooling, the corresponding 2D dense mask $\widetilde{\vec{M}}^{i\to k} \in \{0,1\}^{H\times W}$ is downsampled to $\mathbb{R}^{H'\times W'}$ and then broadcast along the channel dimension to $\widetilde{\vec{M}}_{\text{align}}^{i\to k} \in \mathbb{R}^{C\times H'\times W'}$ to align with the shape of the feature map $\vec{F}_k^{\text{global},i}$. The 2D CLIP embedding $\vec{F}_k^{\text{2D},i} \in \mathbb{R}^{1\times D}$ is obtained through:

$$\begin{aligned} \vec{F}_k^{\text{2D},i} &= \text{MLP}\left(\text{MaskPool}(\widetilde{\vec{M}}_{\text{align}}^{i\to k}, \vec{F}_k^{\text{global},i})\right), \\ \text{s.t.}\quad \text{MaskPool}(\vec{M},\vec{F}) &= \frac{\vec{M}\odot\vec{F}}{\|\vec{M}\|_{\text{L1-norm}}} \end{aligned} \tag{6}$$

Here, $\odot$ denotes element-wise multiplication, and $\text{MLP}(\cdot)\colon \mathbb{R}^C \to \mathbb{R}^D$ is a one-layer perceptron projecting mask-pooled features into the CLIP space. Finally, we obtain the teacher model's output, CLIP embeddings $\{\vec{F}_k^{\text{2D},i}\}_{k=1}^K$, as learning targets for the student model.

3.2 Student model design

Given each i-th 3D instance Ω_i^{3D} and its corresponding point-wise features $\{f_m^{(i)}\}_{m=1}^M$ generated by the Mask3D, we propose a vision-language adapter (VL-adapter) module to derive a 3D embedding for each instance. Specifically, we perform average pooling on the corresponding M point features $\{f_m^{(i)}\}_{m=1}^M$, yielding an average embedding. Then, we design a two-layer MLP to map the average embedding into CLIP's embedding space, producing a 3D instance embedding $\vec{F}^{\text{3D},i} \in \mathbb{R}^D$ as the output of the student model.

$$\vec{F}^{\text{3D},i} = \text{MLP}\left(\frac{1}{M}\sum_{m=1}^{M} f_m^{(i)}\right) \tag{7}$$

Subsequently, we propose a label-guided distillation algorithm to distill the open-vocabulary knowledge of the teacher model into the VL-adapter of the student model.

3.3 Label-guided distillation algorithm

For the i-th instance proposal, the teacher model generates K multi-view 2D embeddings $\{\vec{F}_k^{\text{2D},i}\}_{k=1}^K$ as the learning

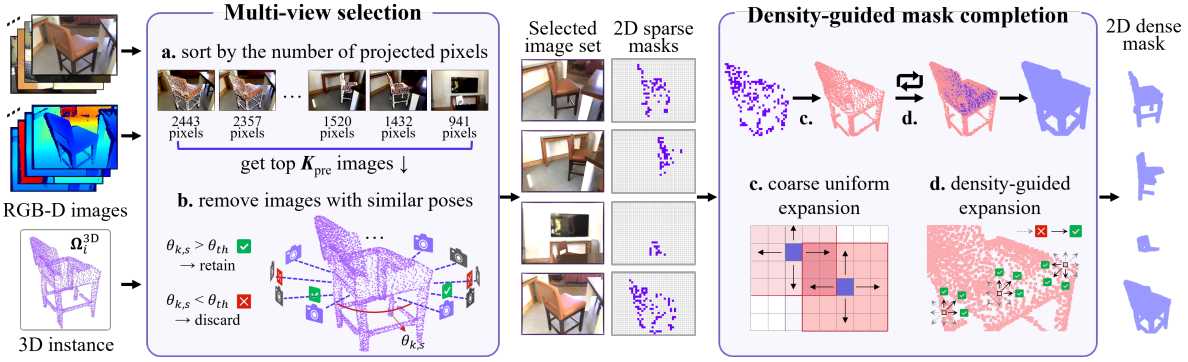

Figure 2: Multi-view selection algorithm and density-guided densification algorithm. For each instance, the multi-view selection algorithm first projects 3D points on 2D images to get a set of pixels which could form sparse 2D masks and sorts images by the number of projected pixels to get the top-K_{pre} images (a), and then removes images with similar poses (b). Given the corresponding 2D sparse masks of the selected images, the density-guided mask completion algorithm firstly performs a coarse uniform expansion (c), and then iteratively conducts the density-guided expansion to get 2D dense masks (d).

target for the student model, where each $\vec{F}_k^{\text{2D},i} \in \mathbb{R}^D$ represents a view-specific embedding. However, some 2D embeddings may be semantically inconsistent with the i-th instance, which can degrade distillation quality. To address this, we propose a label-guided distillation algorithm to filter out a small subset of 2D embeddings with conflicting semantic information. Specifically, we utilize the CLIP text encoder to generate text embeddings $\{\vec{T}_c\}_{c=1}^{C}$ for C candidate categories, where each text embedding $\vec{T}_c \in \mathbb{R}^D$ represents a category. Then, we compute the cosine similarity between each 2D embedding and the CLIP text embeddings, selecting the category corresponding to the text embedding with the highest similarity as the classification label $l^{(i)}$.

$$l_k^{(i)} = \underset{c\in\{1,2,\ldots,C\}}{\arg\max} \left(\frac{\vec{F}_k^{\text{2D},i} \cdot \vec{T}_c}{\|\vec{F}_k^{\text{2D},i}\| \|\vec{T}_c\|} \right) \tag{8}$$

Then, we employ a multi-view voting mechanism to filter out the minority of 2D embeddings with inconsistent labels, obtaining the most frequent label $l^{(i)}$.

$$l^{(i)} = \underset{c\in\{1,2,\ldots,C\}}{\arg\max} \sum_{k=1}^{K} \mathbb{1}(l_k^{(i)} = c) \tag{9}$$

Subsequently, we compute the average embedding of the 2D embeddings with classification labels equal to $l^{(i)}$, as the final CLIP embedding $\vec{F}^{\text{2D},i}$ for the i-th instance proposal.

$$\vec{F}^{\text{2D},i} = \frac{\sum_{k=1}^{K} \mathbb{1}(l_k^{(i)} = l^{(i)}) \vec{F}_k^{\text{2D},i}}{\sum_{k=1}^{K} \mathbb{1}(l_k^{(i)} = l^{(i)})} \tag{10}$$

To transfer the open-vocabulary knowledge from the teacher model to the student model, we employ the contrastive loss between the 2D CLIP embeddings $\{\vec{F}^{\text{2D},i}\}_{i=1}^{N}$ of the teacher model and the 3D instance embeddings $\{\vec{F}^{\text{3D},i}\}_{i=1}^{N}$ produced by the student model as the primary optimization objective for knowledge distillation. This encourages the student model to generate embeddings that align with those of the teacher model in the same embedding space. The contrastive loss is defined as:

$$\mathcal{L}_{\text{contrastive}} = \frac{1}{N} \sum_{i=1}^{N} \left(-\log \frac{\exp(\vec{F}^{\text{2D},i} \cdot \vec{F}^{\text{3D},i}/\tau)}{\sum_{t=1}^{N} \exp(\vec{F}^{\text{2D},i} \cdot \vec{F}^{\text{3D},t}/\tau)} \right) \tag{11}$$

Here, N represents the total number of instance proposals, and τ is the temperature parameter.

Furthermore, to enhance the student model's classification accuracy, we introduce label supervision into the distillation process. Specifically, we first utilize CLIP text embeddings $\{\vec{T}_c\}_{c=1}^{C}$ to compute classification labels for the 3D embeddings generated by the student model. Subsequently, we calculate the cross-entropy loss $\mathcal{L}_{\text{CE}}$ between the labels of the student model and the teacher model, serving as the second optimization objective. During the label-guided distillation, we define the total loss as a weighted sum of the contrastive loss and the label loss, as follows:

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{contrastive}} + \beta \mathcal{L}_{\text{CE}} \tag{12}$$

where α and β are two positive hyperparameters, representing the weights of the contrastive loss and the cross-entropy loss in the total loss, respectively.

4 Experiments

Dataset and metrics. We conduct experiments on the ScanNet200 and Replica (Straub et al. 2019) datasets and our analysis on ScanNet200 is based on 312 validation scenes, covering 198 object categories, divided into head (66), common (68), and tail (66) subsets based on their frequency in training scenes. This partition enables comprehensive evaluation under a long-tail distribution, making ScanNet200 an ideal benchmark for open-vocabulary 3D instance segmentation. For evaluation metrics, we report standard average precision (AP) at IoU thresholds of 50% (AP_{50}) and 25% (AP_{25}), as well as mean AP (mAP) across IoU thresholds from 50% to 95% in 5% increments (AP). Additionally, we assess AP on the head, common, and tail subsets, denoted as AP_{head}, AP_{com}, and AP_{tail}, respectively.

Model	Proposal source	AP	AP_{50}	AP_{25}	AP_{head}	AP_{com}	AP_{tail}	time/scene (s)
Mask3D (2023)		26.9	36.2	41.4	39.8	21.7	17.9	13.41
SAM3D (2023)	2D	6.1	14.2	21.3	7.0	6.2	4.6	482.60
OpenScene (2023)	Mask3D	11.7	15.2	17.8	13.4	11.6	9.9	46.45
OVIR-3D (2023)	2D	13.0	24.9	32.3	14.4	12.7	11.7	466.80
OpenMask3D (2023)	Mask3D	15.4	19.9	23.1	17.1	14.1	14.9	553.87
SAI3D (2024)	2D	12.7	18.8	24.1	12.1	10.4	16.2	163.85
Open3DIS (2024)	ISBNet	18.6	23.1	27.3	24.7	16.9	13.3	57.68
Open3DIS (2024)	ISBNet +2D	23.7	29.4	32.8	27.8	21.2	21.8	360.12
Open-YOLO 3D (2025)	Mask3D	24.7	31.7	36.2	27.8	24.3	21.6	21.80
FOLK (Ours)	Mask3D	**26.6**	**35.7**	**41.4**	**30.2**	**25.0**	**24.0**	**3.64**

Table 1: Open vocabulary 3D instance segmentation results on the ScanNet200 validation set.

Model	Proposal source	AP	AP_{50}	AP_{25}	time/scene (s)
OVIR-3D	2D	11.1	20.5	27.5	52.74
OpenScene	Mask3D	8.2	10.5	12.6	4.29
OpenMask3D	Mask3D	13.1	18.4	24.2	547.32
Open3DIS	ISBNet +2D	18.5	24.5	28.2	187.97
Open-YOLO 3D	Mask3D	**23.7**	**28.6**	**34.8**	16.60
FOLK (Ours)	Mask3D	20.9	27.9	34.7	**2.91**

Table 2: Open vocabulary 3D instance segmentation results on Replica dataset. We use Mask3D trained on ScanNet200 training set to generate class-agnostic mask proposals.

Multi view	Dense mask	Mask pooling	Distill	AP	Time/ scene (s)
✗	✗	✗	✗	15.4	553.87
✓	✗	✗	✗	16.4	545.82
✓	✗	✓	✗	22.9	553.83
✗	✓	✓	✗	26.6	593.95
✓	✓	✓	✗	**27.0**	585.95
✓	✓	✓	✓	26.6	**3.64**

Table 3: Ablation study on the FOLK method conducted on the ScanNet200 dataset. "Multi view" refers to the multi-view selection algorithm. "Dense mask" denotes the density-guided mask completion algorithm. "Mask pooling" refers to using mask pooling method to extract 2D CLIP embeddings. "Distill" denotes the label-guided knowledge distillation algorithm and an ✗ means inference is performed by the teacher model via 3D-to-2D mapping in this column, while a ✓ denotes using the distilled student model to directly infer from 3D data.

Implementation Details. Since each scene in the ScanNet200 dataset contains thousands of image frames, we follow Open3DIS (Nguyen et al. 2024) by selecting one image every 10 frames to minimize redundant perspectives. Instance proposals and point-wise features are obtained using the Mask3D model pretrained on the ScanNet200 training set. For generating text embeddings to assign classification labels, category names are embedded into the fixed text template "a [category] in the scene" and passed through the CLIP text encoder to obtain corresponding text embeddings. For the teacher model's hyperparameters, in the multi-view selection algorithm, we use $K_{\text{pre}} = 6$ in Equation (1) and an angular difference threshold $\theta_{\text{th}} = 5^\circ$. In the density-guided mask completion algorithm, we expand each pixel uniformly within a square window of radius $r = 7$ in Equation (2) and apply a Gaussian kernel of size $k_s = 10$ in Equation (3) with a density threshold $\rho_{\text{th}} = 0.02$. We also set $S = 3$ in Equation (5) and iterate for 2 rounds to obtain dense masks. For knowledge distillation, we set the hyperparameter τ to 0.01 in Equation (11). Additionally, the loss weight factors $\alpha = 0.4$ and $\beta = 0.6$ are used in Equation (12) to optimize distillation performance. All experiments are conducted on a single NVIDIA RTX 3090 24GB GPU.

Comparison to prior methods. We compare the proposed FOLK method with six open-vocabulary 3D instance segmentation models (Yang et al. 2023; Peng et al. 2023; Lu et al. 2023a; Takmaz et al. 2023; Nguyen et al. 2024; Boudjoghra et al. 2025) and a representative closed-vocabulary model, Mask3D, on the ScanNet200 and Replica dataset. As shown in Table 1, our method achieves state-of-the-art performance across all metrics among open-vocabulary methods, relying solely on class-agnostic proposals generated by Mask3D. Notably, our performance is comparable to that of the closed-vocabulary Mask3D model. Furthermore, by eliminating the need for 3D-to-2D mapping during inference, our method achieves the fastest inference speed, which is approximately between 6.0× and 152.2× faster than previous methods (see Appendix A). In Figure 3, we also present qualitative results of our method on the ScanNet200 dataset (see in Appendix C). Our method achieves clear segmentation and successfully identifies most instances, closely aligning with the ground truth. Compared to OpenMask3D and OpenYOLO-3D, our method detects more candidate instances and classifies them accurately, exhibiting a higher matching rate (see Appendix B). As shown in Table 2, we also evaluate our method on the Replica dataset, where it achieves performance comparable to current state-of-the-art open-vocabulary methods. Notably, our approach significantly improves inference efficiency, achieving speedups ranging from 1.47× to 188.1× over prior methods.

Ablation study. We study the effectiveness of different components of the proposed FOLK method through systematic ablation experiments. As presented in Table 3, each component independently enhances performance compared to the baseline, and their combined integration yields synergistic improvements. Specifically, the mask pooling method improves classification accuracy by effectively mitigating the impact of background noise. The density-guided mask completion algorithm further enhances performance by generating precise 2D dense masks. When combined with the multi-view selection algorithm, the model leverages higher-

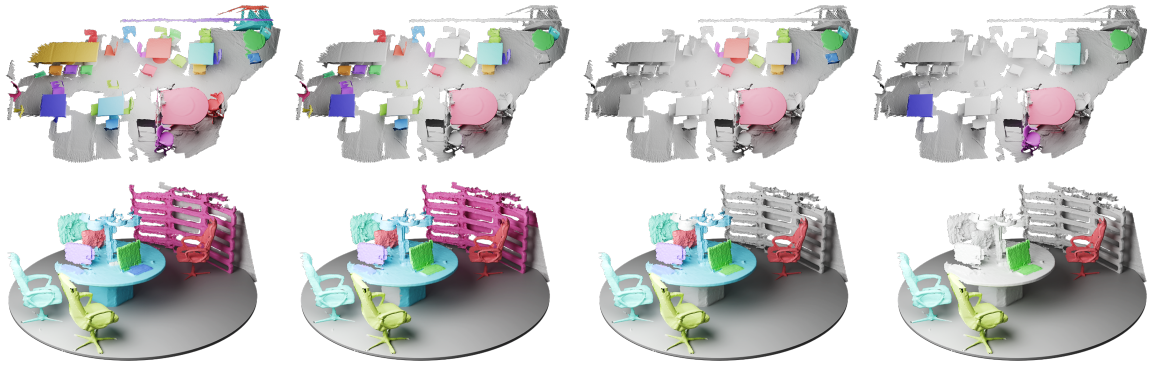

Figure 3: Qualitative results on the Scannet200 dataset. We show detection results from our method alongside two representative baselines: OpenMask3D and OpenYOLO-3D. Our method detects more candidate instances with higher classification accuracy, and demonstrates superior segmentation quality in terms of both precision and recall.

quality and more varied 2D CLIP embeddings, resulting in further gains in classification accuracy. Moreover, the label-guided knowledge distillation algorithm enables the training of a student model that achieves a compelling trade-off: while the teacher model exhibits slightly higher accuracy, the student model significantly reduces inference time per scene (approximately 161× faster) with only a marginal performance drop (0.4 decrease in AP), making it highly suitable for large-scale deployment.

K_{pre}	AP	AP_{50}	AP_{25}	AP_{head}	AP_{com}	AP_{tail}
5	26.8	35.9	42.0	30.8	25.0	24.1
6	**27.0**	**36.4**	**42.2**	**30.9**	**25.4**	**24.5**
7	26.8	36.2	42.2	30.9	25.3	24.0
8	26.8	36.1	42.1	30.9	25.2	24.0

Table 4: Exploring the effectiveness on the different values of K_{pre} in the multi-view selection algorithm. We evaluated the different values of K_{pre} in Equation (1) of detection accuracy on the ScanNet200 dataset before the distillation.

α	β	AP	AP_{50}	AP_{25}	AP_{head}	AP_{com}	AP_{tail}
1.0	0.0	16.4	21.8	25.5	18.2	13.8	17.5
0.8	0.2	22.5	30.0	35.2	26.2	19.8	21.4
0.4	**0.6**	**26.6**	**35.7**	**41.4**	**30.2**	**25.0**	**24.0**
0.2	0.8	26.2	35.3	40.8	29.9	24.7	23.5
0.0	1.0	25.9	34.5	39.4	29.6	25.0	22.6

Table 5: Exploring the effectiveness of varying ratios of contrastive loss and label loss on the ScanNet200 dataset. The α and β represent the weights assigned to two loss terms, and a larger value of β indicates stronger guidance from the label loss. When $\beta = 0$ and $\alpha = 1$, our method performs pure knowledge distillation without any label guidance.

Hyperparameters Exploration. We explore key hyperparameters in the proposed FOLK method. First, we explore the effect of the number of selected images K_{pre} in the multi-view selection algorithm (Equation (1)) by setting $K_{\text{pre}} = 1, 5, 6, 7, 8$. As shown in Table 4, the proposed FOLK method consistently outperforms the current state-of-the-art across all settings. Notably, the best performance is achieved when $K_{\text{pre}} = 6$. Second, we investigate how varying α and β affect distillation performance. As shown in Table 5, the results indicate that without label guidance ($\beta = 0$), the distilled model achieves low instance segmentation accuracy, with an AP of 16.4. In contrast, with stronger guidance ($\beta = 0.6, 0.8, 1.0$), it consistently surpasses the state of the art. The best result (AP = 26.6) is achieved with $\alpha = 0.4$ and $\beta = 0.6$, highlighting the importance of label supervision in knowledge distillation.

5 Conclusion and limitations

Conclusion. In this paper, we introduced FOLK, a novel method for open-vocabulary 3D instance segmentation that addresses limitations of existing 2D mapping-based methods. By distilling open-vocabulary knowledge from the 2D CLIP model into a 3D student model via a label-guided distillation algorithm, FOLK eliminates the need for 3D-to-2D mapping during inference. This approach preserves geometric information of 3D point clouds, mitigates noise from 2D occlusions, and significantly boosts inference speed. Moreover, our results demonstrate the effectiveness of each proposed algorithm and the robustness of hyperparameter settings, paving the way for scalable open-vocabulary 3D instance segmentation in real-world applications.

Limitations. Our method uses the Mask3D model to generate 3D proposals for segmentation. However, Mask3D occasionally produces low-quality instance proposals, compromising segmentation precision and classification accuracy. Thus, there is room to improve proposal quality, which we plan to explore in future work. Additionally, our experiments show that while knowledge distillation significantly accelerates inference—achieving a 161× speedup over the teacher model (Table 1)—the student model exhibits a slight drop in classification accuracy, with AP decreasing by 0.4. Nevertheless, it still outperforms the state-of-the-art. This trade-off may result from limited distillation data, constraining the student model's ability to fully capture the teacher's features. We will further investigate this in future work to enhance distillation effectiveness.

Appendix

A Runtime breakdown.

We explore how long each step of a full open-vocabulary 3D instance-segmentation task takes, comparing our method, shown for both the teacher and student networks, with two representative methods: OpenMask3D and OpenYOLO-3D. In Figure 4, "3D Proposal Generation" refers to the process of generating class-agnostic 3D instance proposals. All methods shown in the figure utilize the Mask3D model for proposal generation. To ensure a fair comparison, we standardize the time consumption for this step across all methods. "2D Data Processing & Extraction" involves different procedures depending on the specific method. In general, this step includes the 3D-to-2D projection, computing the visibility of each mask proposal within all images, and extracting relevant 2D information and features. For our method, this step corresponds to two key algorithms: the multi-view selection algorithm and the density-guided mask completion algorithm. "Prediction Label Inference" denotes the step where 2D model outputs are used to prompt the class-agnostic 3D masks with text embeddings, ultimately resulting in the final predictions. It is noteworthy that our student model does not require the "2D Data Processing & Extraction" step, and the runtime of its "Prediction Label Inference" step is below 0.1 seconds per scene, rendering its time consumption negligible in practice.

B Matching rate.

As shown in Table 6, we report the average number of instance proposals employed per scene and the corresponding matching rate with the ground truth under the AP50 evaluation metric with two representative methods OpenMask3D and OpenYOLO-3D. We apply Non-Maximum Suppression (NMS), a common post-processing technique that removes redundant instance proposals by suppressing lower-confidence predictions with high overlap, to the instance proposals generated by the Mask3D. The IoU threshold is set to 0.9. Similarly, we evaluate OpenMask3D using the same NMS-processed proposals for fair comparison. As OpenYOLO-3D includes its own built-in NMS module with the same IoU threshold, all methods are evaluated under consistent conditions. The results demonstrate that our method not only relies on the fewest proposals but also attains the highest matching rate among all compared methods.

Method	Proposals/scene	Matching rate (%)
FOLK (Ours)	**64**	**17.59**
OpenMask3D+NMS	64	7.81
OpenMask3D	152	7.33
Open-YOLO 3D	600	1.54

Table 6: Average number of proposals per scene and success rate (AP50) for the three evaluated methods on the ScanNet200 validation set.

C Additional qualitative results.

We provide qualitative result visualizations of the open-vocabulary 3D instance segmentation on the ScanNet200 dataset, as shown in Figure 5 and Figure 6. Following the evaluation rules, the wall and floor categories are ignored in the visualization. We visualize the instances that are correctly segmented according to the AP50 evaluation metric.

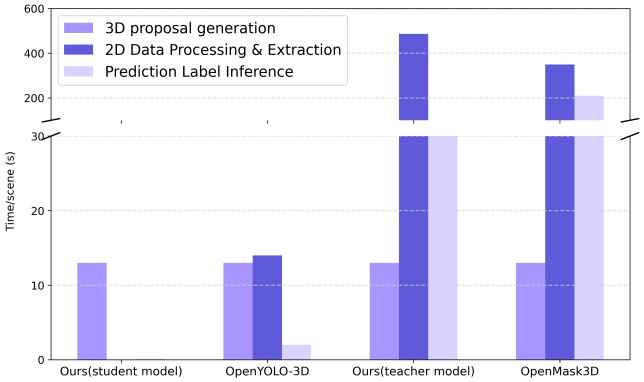

Figure 4: Time breakdown of the average runtime for open-vocabulary 3D instance segmentation on the ScanNet200 dataset.

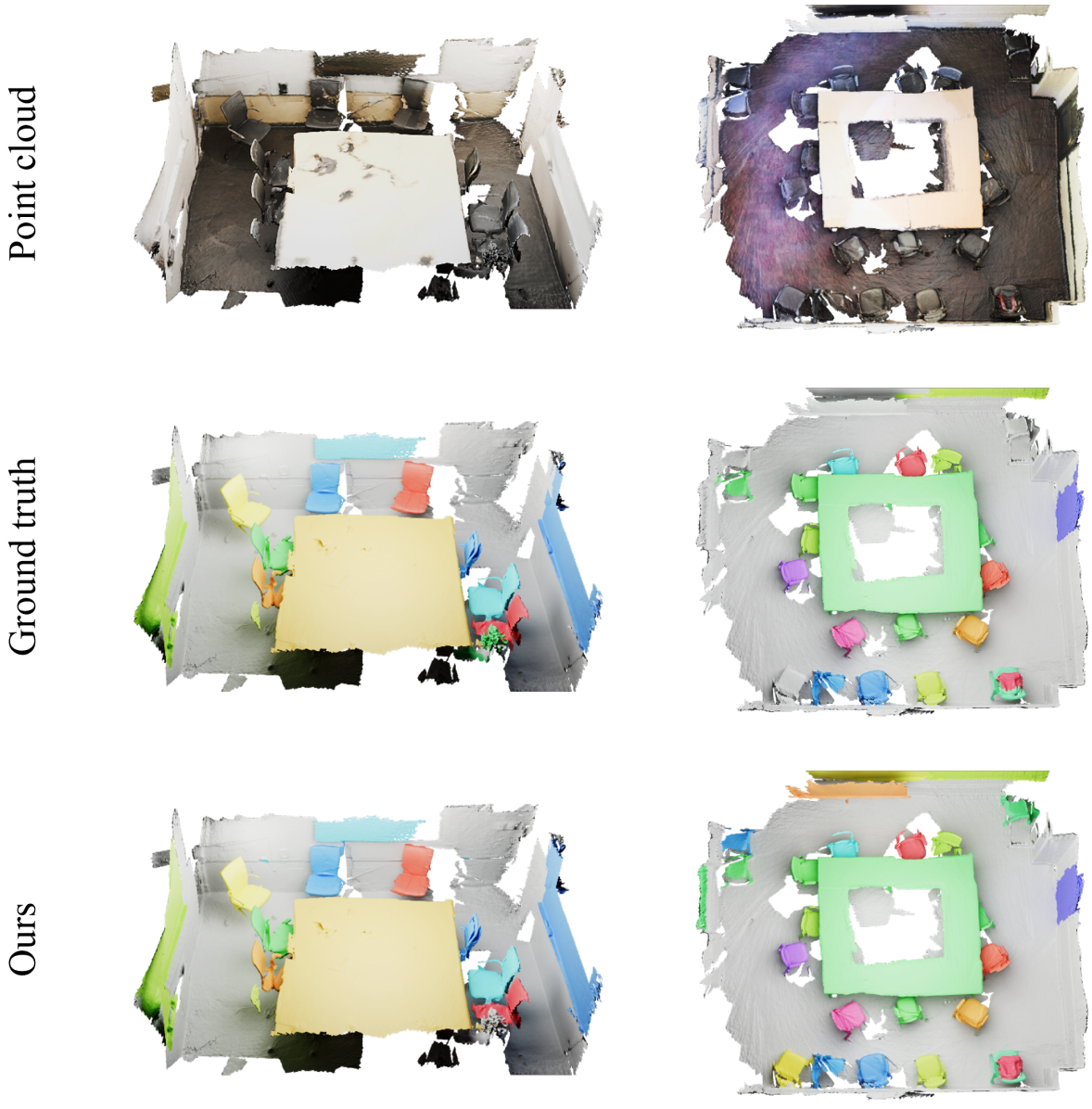

Figure 5: Additional qualitative results on ScanNet200.

Figure 6: Additional qualitative results on ScanNet200.

References

Bansal, A.; Sikka, K.; Sharma, G.; Chellappa, R.; and Divakaran, A. 2018. Zero-shot object detection. In *Proceedings of the European conference on computer vision (ECCV)*, 384–400.

Boudjoghra, M. E. A.; Dai, A.; Lahoud, J.; Cholakkal, H.; Anwer, R. M.; Khan, S.; and Khan, F. S. 2025. Open-YOLO 3D: Towards Fast and Accurate Open-Vocabulary 3D Instance Segmentation. In *The Thirteenth International Conference on Learning Representations*.

Cantrell, C. D. 2000. *Modern mathematical methods for physicists and engineers*. Cambridge University Press.

Chen, S.; Fang, J.; Zhang, Q.; Liu, W.; and Wang, X. 2021. Hierarchical Aggregation for 3D Instance Segmentation. In *International Conference on Computer Vision (ICCV)*.

Engelmann, F.; Bokeloh, M.; Fathi, A.; Leibe, B.; and Nießner, M. 2020. 3D-MPA: Multi Proposal Aggregation for 3D Semantic Instance Segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Felzenszwalb, P. F.; and Huttenlocher, D. P. 2004. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2): 167–181.

Gu, X.; Lin, T.-Y.; Kuo, W.; and Cui, Y. 2022. Open-vocabulary Object Detection via Vision and Language Knowledge Distillation. In *International Conference on Learning Representations (ICLR)*.

Han, L.; Zheng, T.; Xu, L.; and Fang, L. 2020. OccuSeg: Occupancy-aware 3D Instance Segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hou, J.; Dai, A.; and Nießner, M. 2019. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4421–4430.

Huang, Z.; Wu, X.; Chen, X.; Zhao, H.; Zhu, L.; and Lasenby, J. 2024. Openins3d: Snap and lookup for 3d open-vocabulary instance segmentation. In *European Conference on Computer Vision*, 169–185. Springer.

Huynh, D. Q. 2009. Metrics for 3D Rotations: Comparison and Analysis. *Journal of Mathematical Imaging and Vision*, 35(2): 155–164.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q. V.; Sung, Y.-H.; Li, Z.; and Duerig, T. 2021. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *International Conference on Machine Learning (ICML)*.

Jiang, L.; Zhao, H.; Shi, S.; Liu, S.; Fu, C.-W.; and Jia, J. 2020. PointGroup: Dual-Set Point Grouping for 3D Instance Segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, 4015–4026.

Kuo, W.; Cui, Y.; Gu, X.; Piergiovanni, A.; and Angelova, A. 2022. F-vlm: Open-vocabulary object detection upon frozen vision and language models. *arXiv preprint arXiv:2209.15639*.

Li, B.; Weinberger, K. Q.; Belongie, S.; Koltun, V.; and Ranftl, R. 2022. Language-driven Semantic Segmentation. In *International Conference on Learning Representations (ICLR)*.

Li, Z.; Zhou, Q.; Zhang, X.; Zhang, Y.; Wang, Y.; and Xie, W. 2023. Open-vocabulary object segmentation with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7667–7676.

Liang, Z.; Li, Z.; Xu, S.; Tan, M.; and Jia, K. 2021. Instance segmentation in 3d scenes using semantic superpoint tree networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2783–2792.

Liu, S.-H.; Yu, S.-Y.; Wu, S.-C.; Chen, H.-T.; and Liu, T.-L. 2020. Learning gaussian instance segmentation in point clouds. *arXiv preprint arXiv:2007.09860*.

Lu, S.; Chang, H.; Jing, E. P.; Boularias, A.; and Bekris, K. 2023a. Ovir-3d: Open-vocabulary 3d instance retrieval without training on 3d data. In *Conference on Robot Learning*, 1610–1620. PMLR.

Lu, Y.; Xu, C.; Wei, X.; Xie, X.; Tomizuka, M.; Keutzer, K.; and Zhang, S. 2023b. Open-vocabulary point-cloud object detection without 3d annotation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1190–1199.

Nguyen, P. D.; Ngo, T. D.; Gan, C.; Kalogerakis, E.; Tran, A.; Pham, C.; and Nguyen, K. 2024. Open3dis: Open-vocabulary 3d instance segmentation with 2d mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Peng, S.; Genova, K.; Jiang, C. M.; Tagliasacchi, A.; Pollefeys, M.; and Funkhouser, T. 2023. OpenScene: 3D Scene Understanding with Open Vocabularies. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning (ICML)*.

Rozenberszki, D.; Litany, O.; and Dai, A. 2022. Language-Grounded Indoor 3D Semantic Segmentation in the Wild. In *European Conference on Computer Vision (ECCV)*.

Schult, J.; Engelmann, F.; Hermans, A.; Litany, O.; Tang, S.; and Leibe, B. 2023. Mask3D: Mask Transformer for 3D Semantic Instance Segmentation. In *International Conference on Robotics and Automation (ICRA)*.

Straub, J.; Whelan, T.; Ma, L.; Chen, Y.; Wijmans, E.; Green, S.; Engel, J. J.; Mur-Artal, R.; Ren, C.; Verma, S.; Clarkson, A.; Yan, M.; Budge, B.; Yan, Y.; Pan, X.; Yon, J.; Zou, Y.; Leon, K.; Carter, N.; Briales, J.; Gillingham, T.; Mueggler, E.; Pesqueira, L.; Savva, M.; Batra, D.; Strasdat, H. M.; Nardi, R. D.; Goesele, M.; Lovegrove, S.; and Newcombe, R. 2019. The Replica Dataset: A Digital Replica of Indoor Spaces. arXiv:1906.05797.

Takmaz, A.; Fedele, E.; Sumner, R. W.; Pollefeys, M.; Tombari, F.; and Engelmann, F. 2023. OpenMask3D: Open-Vocabulary 3D Instance Segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Wu, S.; Zhang, W.; Xu, L.; Jin, S.; Li, X.; Liu, W.; and Loy, C. C. 2023a. Clipself: Vision transformer distills itself for open-vocabulary dense prediction. *arXiv preprint arXiv:2310.01403*.

Wu, W.; Zhao, Y.; Shou, M. Z.; Zhou, H.; and Shen, C. 2023b. Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1206–1217.

Wu, X.; Zhu, F.; Zhao, R.; and Li, H. 2023c. CORA: Adapting CLIP for Open-Vocabulary Detection with Region Prompting and Anchor Pre-Matching. arXiv:2303.13076.

Yang, B.; Wang, J.; Clark, R.; Hu, Q.; Wang, S.; Markham, A.; and Trigoni, N. 2019. Learning Object Bounding Boxes for 3D Instance Segmentation on Point Clouds. In *Neural Information Processing Systems (NeurIPS)*.

Yang, Y.; Wu, X.; He, T.; Zhao, H.; and Liu, X. 2023. Sam3d: Segment anything in 3d scenes. *arXiv preprint arXiv:2306.03908*.

Yin, Y.; Liu, Y.; Xiao, Y.; Cohen-Or, D.; Huang, J.; and Chen, B. 2024. Sai3d: Segment any instance in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3292–3302.

Zeng, Q.-S.; Li, Y.; Zhou, D.; Li, G.; Hou, Q.; and Cheng, M.-M. 2025. High-Quality Mask Tuning Matters for Open-Vocabulary Segmentation. In *arXiv preprint arXiv:2412.11464*.

Zhong, Y.; Yang, J.; Zhang, P.; Li, C.; Codella, N.; Li, L. H.; Zhou, L.; Dai, X.; Yuan, L.; Li, Y.; et al. 2022. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16793–16803.