OVIR-3D: Open-Vocabulary 3D Instance Retrieval Without Training on 3D Data

Anonymous Author(s)
Affiliation
Address
email

Abstract: This work presents OVIR-3D, a straightforward yet effective method for open-vocabulary 3D object instance retrieval without using any 3D data for training. Given a language query, the proposed method is able to return a ranked set of 3D object instance segments based on the feature similarity of the instance and the text query. This is achieved by a multi-view fusion of text-aligned 2D region proposals into 3D space, where the 2D region proposal network could leverage 2D datasets, which are more accessible and typically larger than 3D datasets. The proposed fusion process is efficient as it can be performed in real-time for most indoor 3D scenes and does not require additional training in 3D space. Experiments on public datasets and a real robot show the effectiveness of the method and its potential for applications in robot navigation and manipulation.

Keywords: Open Vocabulary, 3D Instance Retrieval

1 Introduction

There is recent progress in open-vocabulary 2D detection and segmentation methods [1, 2, 3], typically based on vision-language models [4, 5, 6] that are pre-trained on large data sets. In principle, this progress enables robots to detect and segment a wide variety of objects from 2D images. Open-vocabulary methods can detect and localize objects given an arbitrary text specification based on feature similarity. This is in contrast to traditional methods that can perform inference only over a closed set of categories, which are often the same as in the training set. The open-vocabulary aspect of such detectors has great benefits for robotics applications and reduces the need for retraining a model when novel categories are introduced to the robot's environment. For example, detecting not only a mug but also its handle is important for grasping the mug from its handle. Furthermore, open-vocabulary methods allow users to query an object not only by its category but also by its properties or affordance, such as "entertainment" and "graspable".

Despite the success of the 2D open-vocabulary models, their counterparts in the 3D space have not yet been widely explored. 3D models are important for many robotic tasks, such as navigation and manipulation and autonomous robots need to reason regarding 3D object instances in their surroundings. One possible reason is the lack of large 3D datasets with sufficient object diversity for training open-vocabulary models. The 3D dataset with the largest number of object categories to date is ScanNet200 [8], which contains 200 categories of objects. This, however, is 100x smaller than the 2D image dataset ImageNet21K [9] in terms of object categories. The annotation cost in the 3D space is also significantly larger than in 2D, which makes it unlikely for 3D datasets to match the size of 2D datasets. Early efforts of dense semantic mapping [10, 11, 12, 13] address this problem by projecting multi-view 2D detections to 3D using a closed-set detector, which cannot handle arbitrary language queries. More recently, OpenScene [14] achieved open-vocabulary 3D semantic segmentation by projecting text-aligned pixel features extracted from a 2D open-vocabulary segmentation model to 3D and further distilling 3D features from the projected 2D features. Once point features are ensembled, a heatmap of the point cloud can be generated based on the feature similarity between points and the input text query. Nevertheless, manual thresholding is required during

Submitted to the 7th Conference on Robot Learning (CoRL 2023). Do not distribute.

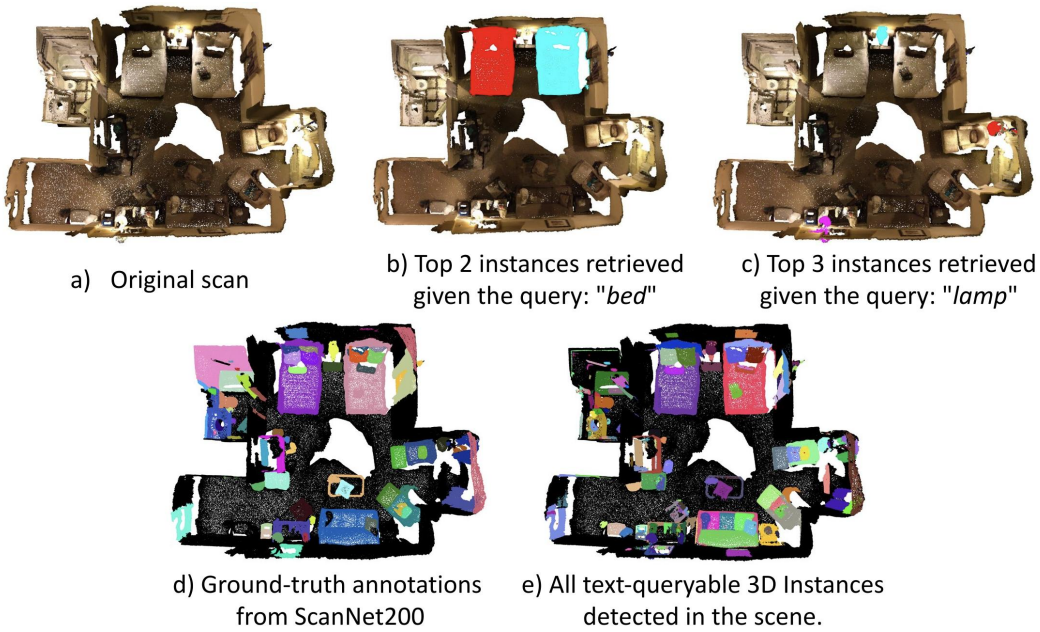

Figure 1: **Examples of open-vocabulary 3D instance retrieval using the proposed system.** (a-c) Given a 3D scan reconstructed from an RGB-D video (e.g., scene0645 from ScanNet [7]) and a text query (e.g., bed, lamp), the proposed method retrieves a set of 3D instances ranked based on their semantic similarity to the text query. (d-e) Instances that are not even in the ground-truth annotations can also be detected and queried by the proposed method, such as the cushions on the sofa.

test time to generate a binary mask, which would be needed for robot manipulation or navigation purposes. OpenScene also lacks the ability to separate instances that belong to the same category, which also limits its use in robotics applications.

This work aims to address open-vocabulary 3D instance retrieval as shown in Figure 1, i.e., given a text query during inference, return a set of 3D instance segments ranked by their semantic similarities to the query. The 2 main contributions are: (1) an efficient 2D-to-3D instance fusion module given text-aligned region proposals, and (2) an open-vocabulary 3D instance retrieval method that, given a text query, returns a ranked set of 3D instances based on semantic similarity.

In particular, this work considers a scenario, where a mobile robot navigates in an indoor scene and automatically reconstructs the 3D environment using its RGB-D sensor and an off-the-shelf SLAM module. The method fuses semantic instance information to a reconstructed 3D scene, so that given a text query such as “fetch a mug from the kitchen”, the robot can locate the relevant object instances in the reconstructed 3D scene and perform the task. Similar to the aforementioned 2D open-vocabulary methods, the proposed method also leverages the power of 2D models trained on large image datasets. It first generates text-aligned 2D object region proposals by querying a 2D open-vocabulary detector with a very large vocabulary. Those region proposals and corresponding features are then fed to a data association method for 3D instance fusion that is based on the overlap of projected 3D segments and the similarity of their semantic features. Periodic filtering and merging of 3D instances is performed to remove noisy detections and to improve the quality of instance masks. Finally, a post-processing step separates objects that are not connected in the 3D space but are incorrectly associated. No additional training on any 3D data is required during this process, which makes it a plug-and-play tool for real robotic applications. During inference, the fused instances in 3D are ranked based on their semantic similarity to the input text query, and users can further specify if the most relevant instance or the top k instances are to be retrieved depending on the specified task.

Extensive experiments and ablation studies were conducted on real scans of rooms from ScanNet [8] and reconstructed tabletop scenes from the YCB-Video [15] dataset to show the effectiveness of the proposed system and its components. Furthermore, real robot manipulation experiments highlight the benefits of instance segmentation compared to existing semantic segmentation methods.

2 Related Work

2.1 2D Open-Vocabulary Detection and Segmentation

With the advent of large vision-language pre-trained models, such as CLIP [4], ALIGN [5] and LiT [16], a number of 2D open-vocabulary object detection and segmentation methods have been proposed [17, 18, 19, 1, 2, 20, 21]. For 2D semantic segmentation, LSeg [17] encodes 2D images and aligns pixel features with segment label embeddings. OpenSeg [18] uses image-level supervision, such as caption text, which allows scaling up training data. GroupViT [19] performs bottom-up hierarchical spatial grouping of semantically-related visual regions for semantic segmentation. For 2D object detection, ViLD [1] achieves open-vocabulary detection by aligning the features of class-agnostic region proposals with text label features. Detic [2] attempts to address the long-tail detection problem by utilizing data with bounding box annotations and image-level annotations. OWL-ViT [20] proposes a pipeline for transferring image-text models to open-vocabulary object detection. Our proposed method adopts Detic [2] as a backbone detector to locate objects in 2D images since it can provide pixel-level instance segmentation and text-aligned features. Furthermore, it can be queried with a large vocabulary without sacrificing much speed.

2.2 3D Reconstruction and Closed-Vocabulary Semantic Mapping

Early works have addressed the 3D reconstruction problem either through online SLAM methods [22, 23, 24, 25, 26] or offline methods like structure-from-motion [27, 28] using a variety of 3D representations, such as TSDF [29], Surfel [30], and more recently NeRF [31, 32, 33, 34]. With the advancement of learning-based 2D object detection and segmentation methods, recent efforts have focused on point-wise dense semantic mapping of 3D scenes [10, 11, 13, 12, 35]. Despite being effective, these methods have not yet been designed to fit open-vocabulary detectors. They either assume mutually exclusive instances [10, 13] or utilize category labels for data association [11, 12, 35]. The proposed method in this paper adopts an off-the-shelf 3D reconstruction method and focuses on integrating 2D information with point-cloud information to achieve open-vocabulary 3D instance segmentation. A key contribution in this context is a method that associates open-vocabulary 2D instance detections and fuses them into a 3D point cloud while keeping them open-vocabulary.

2.3 3D Open-Vocabulary Scene Understanding

More recently, research efforts aim for open-vocabulary 3D scene understanding [36, 14, 37, 38, 39, 40, 41]. Given that existing 3D datasets tend to be significantly smaller than 2D image datasets, this is mainly accomplished by fusing pretrained 2D image features into 3D reconstructions. OpenScene [14] projects pixel-wise features from 2D open-vocabulary segmentation models [17, 18] to a 3D reconstruction and distills 3D features for better semantic segmentation. ConceptFusion [39] fuses multi-modal features, such as sound, from off-the-shelf foundation models that can only produce image-level embeddings. LeRF [40] fuses multi-scale CLIP features to a neural radiance field for open-vocabulary query. These methods can generate a heatmap of a scene that corresponds to a query, but they do not provide instance-level segmentation, which limits their use in tasks that require a robot to interact with specific object instances. PLA [41] constructs hierarchical 3D-text pairs for 3D open-world learning and aims to perform not only 3D semantic segmentation but instance segmentation as well. Nevertheless, the method so far has been demonstrated only on certain furniture-scale objects, and performance in other categories is unclear. On the other hand, our method focuses on instance-level, open-vocabulary 3D segmentation without manual 3D annotation.

3 Problem Formulation

A 3D scan $\mathcal{X}^N$ represented by N points is reconstructed from an RGB-D video $\mathcal{V} = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_T\}$ given known camera intrinsics C and camera poses P_t, where $\mathcal{I}_t$ is the video frame at time t.

The objective in open-vocabulary 3D instance retrieval is to return a set of K ranked instances represented as binary 3D masks $\mathcal{M}^N = \{m_i | i \in [1, K]\}$ over the 3D scan $\mathcal{X}^N$, given a text query Q and the desired number of instances K to be retrieved. The ranking of instance masks is based on the semantic similarity between the 3D instance and the text query, where the most similar instance should be ranked first.

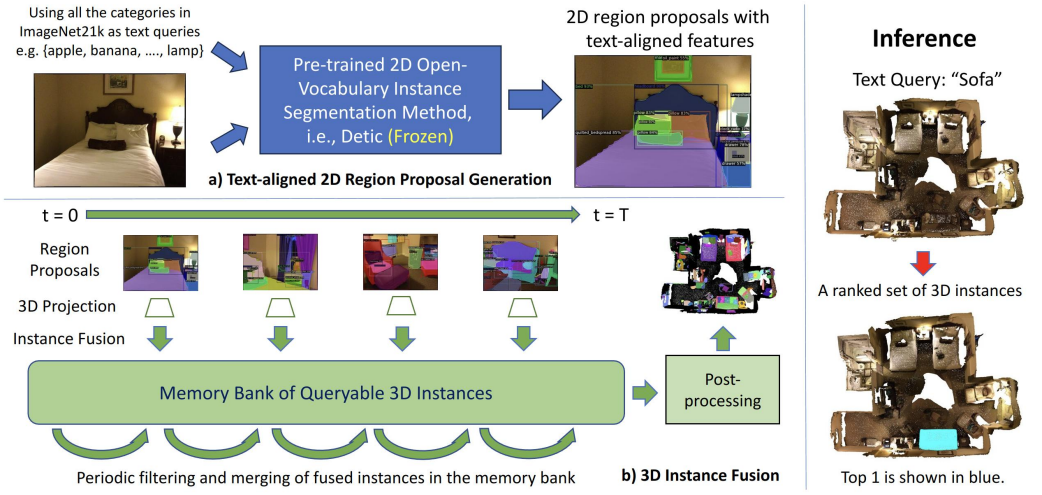

Figure 2: **Pipeline of the proposed method.**

4 Method

The overall pipeline of the proposed method is illustrated in figure 2. To summarize, given a video frame, the method first generates 2D region proposals $\mathcal{R}^{2D} = \{r_1, .., r_k\}$ with text-aligned features $F^{2D} = \{f_1^{2D}, .., f_k^{2D}\}$ using an off-the-shelf 2D open-vocabulary method trained on large 2D datasets. The 2D region proposals $\mathcal{R}^{2D}$ of each frame $\mathcal{I}_t$ are then projected to the reconstructed 3D point cloud given the camera intrinsics C and poses P_t. The projected 3D regions $\mathcal{R}^{3D}$ are either matched to existing 3D object instances $O = \{o_1, .., o_b\}$ with 3D features $F^{3D} = \{f_1^{3D}, .., f_b^{3D}\}$ stored in the memory bank $\mathcal{B}$, or added as a new instance if not matched with anything. The 2D region to 3D instance matching is based on feature similarity $s_{ij} = cos(f_i^{2D}, f_j^{3D})$ and region overlapping $IoU(r_i^{3D}, o_j)$ in the 3D space. Matched regions are integrated into the 3D instance. To remove less reliable detections and improve segmentation quality, periodic filtering and merging of 3D instances in the memory bank $\mathcal{B}$ is performed every T frames. A final post-processing step removes 3D instances that are too small and separates object instances that are not connected in 3D space but are incorrectly merged. During inference time, the text query q will be used to match with a set of representative features of each 3D instance, and the instances O will be ranked based on the similarity and returned. Details of the proposed method are presented below.

4.1 Text-aligned 2D Region Proposal

Since the advent of Faster-RCNN [42], learning-based region proposal networks have served as a critical module for a large number of object detection methods. Directly generating 3D region proposals for open-vocabulary instance retrieval, however, is hard due to the lack of annotated 3D data with enough category varieties, and most existing region proposal networks cannot provide text-aligned features. Instead, this work leverages the power of an off-the-shelf open-vocabulary 2D detector trained with large 2D datasets and uses it as a text-aligned region proposal network by querying it with a huge vocabulary, i.e., the 21k categories from ImageNet21k [9] dataset, which essentially asks the model to detect all the possible instances that it can find in a single frame I_t. The ablation studies of section 6.1 show that region proposals generated in this way have certain generalizability, such that a region of an object is likely to be generated even if the object does not belong to any of the input categories.

In particular, this work adopts Detic [2] to generate 2D region proposals $\mathcal{R}^{2D}$ with text-aligned features F^{2D} by querying 21k categories and setting the confidence threshold to 0.3. Detic achieves open-vocabulary 2D object detection with a standard two-stage object detection architecture, similar to MaskRCNN [43], but modifies its classification head so that features of detected instances are not mapped to specific category IDs but aligned to the text features of annotated categories using the pre-trained language-image model CLIP [4]. This strategy has also been used in other open-vocabulary

detectors, such as ViLD [1], which could be alternatively used for the proposed method to generate region proposals. The reason that this work adopts Detic is that it can output instance segment masks instead of just bounding boxes, and it has a fast inference speed even when queried with a huge vocabulary from ImageNet21k ($\sim$ 10fps on an NVIDIA RTX3090).

While Detic [2] can predict a category label for each detected instance given the input vocabulary, those predicted labels are ignored and not used in 3D fusion. This is because the classification is rather noisy when the vocabulary size is large and instance labels across video frames are often inconsistent, which makes them barely useable for instance matching. Instead, the text-aligned feature f^{2D} of each detected instance r^{2D} is extracted before the classification layer.

4.2 2D-to-3D Instance Fusion

Given a new RGB-D frame $\mathcal{I}_t$ at time t, along with k region proposals $R^{2D} = \{r_1^{2D}, .., r_k^{2D}\}$ with text-aligned features $F^{2D} = \{f_1^{2D}, .., f_k^{2D}\}$, the 2D regions are first projected to the reconstructed 3D scan using camera intrinsics C and pose P_t. The projected 3D regions $\mathcal{R}^{3D} = \{r_1^{3D}, .., r_k^{3D}\}$ are either matched to 3D object instances $O = \{o_1, .., o_b\}$ with 3D features $F^{3D} = \{f_1^{3D}, .., f_b^{3D}\}$, where b is the number of 3D instances already stored in the memory bank $\mathcal{B}$, or added as a new instance if it is not matched with anything. The memory bank is initialized to empty at the beginning.

The matching of 2D region r_i to 3D instance o_j is based on cosine similarity $s_{ij} = cos(f_i^{2D}, f_j^{3D})$ and 3D intersection over union between the projected region r_i^{3D} and visible part of the 3D instance $\hat{o_j}$ in the current frame, i.e., $IoU(r_i^{3D}, \hat{o_j})$. If s_{ij} is greater than a predefined threshold $\theta_s = 0.75$ and the overlapping $IoU(r_i^{3D}, \hat{o_j})$ is also greater than predefined threshold $\theta_{iou} = 0.25$, then they are considered as a match. Matched regions will be aggregated to the 3D instance, i.e., $o_j := o_j \cup r_i^{3D}$ and $f_j^{3D} := f_j^{3D} + f_i^{2D}$. Since multiple 2D region proposals in a single image can belong to the same 3D instance, the matching is not restricted to one-to-one.

In the accompanying implementation, 3D segment and feature information of instances O are represented by $M_p \in \mathbb{N}^{N \times O_{max}}$ and $M_f^{3D} \in \mathbb{R}^{D \times O_{max}}$ respectively, where N is the number of points in the 3D scan, $O_{max} = 3000$ is a predefined constant that represents the maximum number of possible instances in a scene, and D is the dimension of text-aligned features. $M_p[i, j]$ stores the frequency of a point j being assigned to each 3D instance i, and $M_f^{3D}[i]$ stores the sum of features from all regions that are associated with instance i, which is used as a representative feature after normalizing. Meanwhile, for a processed video frame $\mathcal{I}_t$ with k region proposals, the projected regions $\mathcal{R}^{3D}$ and their corresponding features F^{2D} are also represented with two similar matrices $M_r \in \mathbb{N}^{N \times k}$ and $M_f^{2D} \in \mathbb{R}^{D \times k}$. In this way, the pairwise feature similarity S between all region proposals R^{3D} and all 3D instances in the memory bank $\mathcal{B}$ can be computed via the dot product, i.e., $S = M_f^{2D} \cdot normalized(M_f^{3D}[: b])^T$. Similarly, the pairwise intersection I can be computed by first masking M_p based on current visibility, converting it to a binary matrix M_p', and then taking the dot product $I = M_r \cdot M_p'[: b]^T$. The fusion process can be efficiently executed on GPU.

4.3 Periodic 3D Instance Filtering and Merging

The 2D-to-3D instance fusion approach mentioned above is effective but it can still generate a lot more 3D instances than in the real world, as a new 3D instance will be created whenever a region proposal is not properly matched. Furthermore, it can result in low-quality segmentation due to the aggressive integration of noisy 2D segments, which can further worsen the data association accuracy. In order to remove less reliable detections and improve segmentation quality, periodic filtering and merging of 3D instances stored in the memory bank $\mathcal{B}$ is performed every $T = 300$ frames. An ablation study over values of T is available in section 6.3.

Whether a point p from an instance o_i needs to be filtered is based on the detection rate r_p^{vis} of point p, i.e., the frequency of the 3D point p being associated with this instance $c_p^{o_i}$ over the frequency of point p being visible in the video c_p^{vis}. In other words, $r_p^{vis} = c_p^{o_i} / c_p^{vis}$. If the detection rate r_p^{vis} falls below a predefined threshold $\theta_{vis} = 0.2$, then point p will be removed from the instance o_i by setting $c_p^{o_i} := 0$. An ablation study regarding the visibility threshold θ_{vis} is available in section 6.3. Instance o_i as a whole is removed from the memory bank $\mathcal{B}$ if not enough points are contained after point filtering, i.e., $|o_i| < 50$.

Whether to merge 3D instances $\{o_p, o_q\}$ is again based on feature similarity $s_{pq} = cos(f_p^{3D}, f_q^{3D})$ and 3D intersection over union $IoU(o_p, o_p)$ as in the instance fusion process with the same default thresholds $\theta_s = 0.75$ and $\theta_{iou} = 0.25$. While the only difference is that the periodic merging process will consider the case when $recall(o_p, o_q) = |o_p \cup o_q|/|o_q| \geq \theta_{recall}$ and $s_{pq} \geq \theta_s$. This is to merge the small segment o_p with o_q if o_q is mostly contained in o_p and both instances have similar features.

4.4 Post-processing

A simple post-processing step is executed to separate object instances that are not connected in 3D space. This is achieved by using DBSCAN to find 3D point clusters in each instance, where the distance parameter eps is set to $10cm$. If an instance o_i has segments not connected in 3D space, DBSCAN will return more than one point cluster and o_i will be separated in multiple instances. Small clusters with less than 50 points are filtered out.

4.5 Inference

During inference time, a text query q is converted to a feature vector f_q using the language-image pre-trained model CLIP [4]. Instead of representing each 3D instance with the average feature of associated 2D regions, the K clustering centers by K-Means of associated features are used. The clustering centers can be viewed as representative features from a set of viewpoints. The largest cosine similarity s between the text query q and K representative features of an instance is used as the final similarity between the query and the instance. An ablation study on different ways of feature ensemble is presented in section 6.2. Instances are sorted by the final similarity in descending order and returned.

5 Experiments

5.1 Datasets

The first dataset used for the experiment is ScanNet200 [8], which contains a validation set of 312 indoor scans with 200 categories of objects. Uncountable categories ”floor”, ”wall”, and ”ceiling” and their subcategories are not evaluated. The second dataset is YCB-Video [15], which contains a validation set of 12 videos. It’s a tabletop dataset that was originally designed for object 6DoF pose estimation for robot manipulation. The 3D scans of the tabletop scene are reconstructed by KinectFusion [22], and the 3D labels are generated given the mesh models and annotated poses of target objects. Hyper-parameters presented in the method section 4 are the same for both datasets.

5.2 Metrics

Standard mean average precision (mAP) metric for instance retrieval at different IoU thresholds is adopted for the evaluation purpose. In particular, mAP_{25} and mAP_{50} at the IoU threshold $\theta = 0.25$ and $\theta = 0.5$ respectively, as well as the overall mAP, which is the averaged mAP at different IoU thresholds $\theta = [0.5 : 0.05 : 0.95]$ are reported. The proposed method can often find more instances than those annotated in a 3D scene. but only annotated object categories in a 3D scene are used as text queries for evaluation. The results were computed for each 3D scene and then averaged for the whole dataset.

5.3 Baselines

The most relevant work to date is OpenScene [14], which can search for objects given a text query. It returns a heatmap of the input point cloud based on the similarity between point features and the query feature. To use it as a baseline, a threshold needs to be set to convert the heatmap into a binary mask and then cluster foreground points into 3D instances using DBSCAN, similar to the post-processing step in section 4.4. A set of thresholds uniformly sampled from 0.5 to 0.9 with a step of 0.03 is tested for each category and the one with the best overall performance is reported for OpenScene. Furthermore, a series of prior research has focused on semantic mapping using closed-vocabulary detectors. Among them, two representative works that don’t use closed-set category labels for data association are Fusion++ [12] and PanopticFusion [13]. Instead of using their whole SLAM system, this work assumes the ground truth 3D reconstruction and camera poses are given, and only tested their data association and instance mapping algorithms as baselines. Their backbone detector MaskRCNN [43] is replaced with Detic [2] for open-vocabulary detection, and the mean feature of associated 2D detections for each instance is used to match text queries.

5.4 Results

Quantitative results on ScanNet200 and YCB-Video dataset are shown in Table 1. Furthermore, results on different sets of categories with different frequencies in ScanNet200 are shown in Table 2. The proposed method outperforms all other baselines by a large margin in terms of instance retrieval mAP. It also performs well on tail categories that do not frequently appear in the 3D dataset thanks to the use of 2D region proposals. It's surprising that OpenScene is not performing well on this task even with an automatically tuned threshold for each category. This is probably due to the fused features not being distinguishable enough to generate good boundaries of object instances.

	ScanNet200 [8]			YCB-Video [15]		
Method	mAP_{25}	mAP_{50}	mAP	mAP_{25}	mAP_{50}	mAP
OpenScene [14]	0.268	0.190	0.089	0.421	0.333	0.116
*Fusion++ [12]	0.414	0.253	0.094	0.817	0.464	0.120
*PanopticFusion [13]	0.539	0.370	0.150	0.549	0.378	0.136
Ours	**0.564**	**0.443**	**0.211**	**0.831**	**0.801**	**0.427**

Table 1: Results on ScanNet200 [8] and YCB-Video [15] dataset

	mAP_{50}			mAP		
Method	head	common	tail	head	common	tail
OpenScene [14]	0.308	0.178	0.067	0.150	0.076	0.033
*Fusion++ [12]	0.235	0.243	0.288	0.094	0.090	0.098
*PanopticFusion [13]	0.335	0.360	0.424	0.145	0.146	0.162
Ours	**0.417**	**0.433**	**0.469**	**0.224**	**0.214**	**0.193**

Table 2: Results on three sets of categories with different frequencies in ScanNet200 [8]

6 Ablation Studies

6.1 Input queries to the 2D region proposal method

The proposed method utilizes an open-vocabulary 2D detector as a region proposal method by querying it with a large vocabulary. One concern is whether the input query of the 2D method would affect the performance of the 3D instance retrieval. This ablation study uses different categories from multiple datasets as input queries and displays the instance retrieval performance on the ScanNet200 dataset. In addition to ScanNet200 and ImageNet21K, COCO contains 80 categories, LVIS contains 1203 categories. More aggressively, a vocabulary with ImageNet21k categories but without ScanNet200 categories is tested. Results of 3D instance retrieval on the ScanNet200 dataset are shown in Table 3. It is shown that a large vocabulary is helpful and the region proposal network has certain generalizability, such that even when ScanNet200 categories are completely removed from the ImageNet21k categories, it can still find most regions based on similar categories in the vocabulary, and the final performance on ScanNet200 only slightly dropped.

	COCO	ScanNet200	LVIS	ImageNet21k	ImageNet21k - ScanNet200
mAP_{50}	0.228	0.419	0.429	**0.443**	0.410

Table 3: Results on ScanNet200 [8] with different input queries to the 2D region proposal network

6.2 Feature ensemble strategies

Given the data association between 2D region proposals and 3D instances, three different strategies are tested to ensemble 2D features for each 3D instance. The first one is to simply average all the features from corresponding 2D regions. The second one is to cluster 2D features from different viewpoints using the K-Means algorithm and use the clustering centers to represent each instance. During instance retrieval, the feature similarity is defined as the maximum similarity between the query feature and clustering centers. The third one is to use the feature from the largest associated 2D region. The 3D instance retrieval results on the ScanNet200 dataset are shown in Table 4. Using multiple features by clustering to represent a 3D instance edges over simple averaging while using the feature from the largest associated 2D region performs the worst.

	average	clustered (K=16)	clustered (K=64)	feature from largest 2D detection
mAP_{50}	0.428	0.429	**0.443**	0.380

Table 4: Results on ScanNet200 [8] with different feature ensemble strategies

6.3 Time intervals and visibility threshold for periodic instance filtering and merging

The proposed method filters instances based on visibility and merges 3D instances based on their similarity and overlaps periodically every T frame. This ablation study tested different time intervals T and visibility threshold θ_{vis} for filtering. Results of 3D instance retrieval on the ScanNet200 dataset are shown in Table 5 and Table 6 respectively.

	$T = 1$	$T = 100$	$T = 300$	$T = 500$	$T = 1000$
mAP_{50}	0.340	0.417	**0.443**	0.410	0.412

Table 5: Results on ScanNet200 [8] with different time intervals of periodic filtering and merging

	$\theta_{vis} = 0$	$\theta_{vis} = 0.1$	$\theta_{vis} = 0.15$	$\theta_{vis} = 0.2$	$\theta_{vis} = 0.25$	$\theta_{vis} = 0.3$
mAP_{50}	0.256	0.386	0.407	**0.443**	0.418	0.408

Table 6: mAP_{50} on ScanNet200 [8] dataset

7 Robotics Experiments

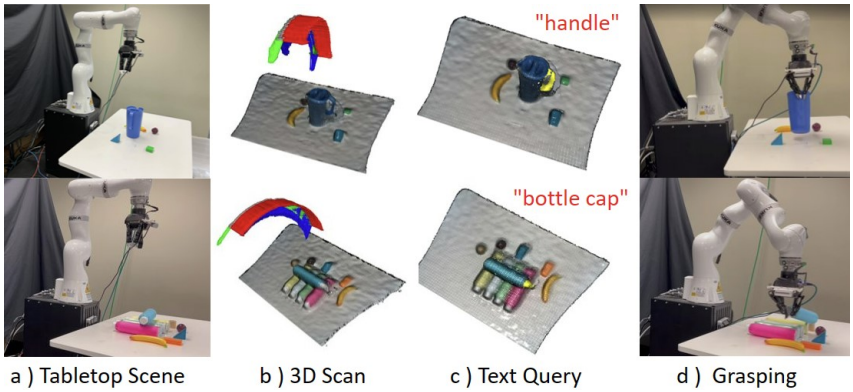

Figure 3: **Robot Experiments.**

Contrasting to conventional closed-set semantic methods, the superiority of open-vocabulary detectors for manipulation lies in their ability to pinpoint the grasp region with a language specification. For instance, a robot can be instructed to seize the pitcher by the handle or grasp the cap of a bottle. This nuanced specification can be exceptionally valuable in affordance-related manipulation tasks. A part-based grasping experiment was devised given this inspiration. The robot is asked to grasp the ”bottle cap” and ”handle” respectively in two sets of experiments with five distinct table setups in each set. OVIR-3D is compared against OpenScene to segment parts given the text query and located segments are used to guide the robot's grasp. For OVIR-3D, the 5/5 graspable bottle caps and 4/5 ”handle” of the pitcher were detected and the robot grasping success rate was 90%. The detection rate for OpenScene was low at 0/5 and 3/5 respectively, and the overall grasping success rate was 30%. An object is considered detected if a reasonable visual segment is found.

8 Conclusion and Limitations

This paper presents OVIR-3D, a rather straightforward but effective method for open-vocabulary 3D instance retrieval. By utilizing an off-the-shelf open-vocabulary 2D instance segmentation method for region proposal and fusing its output 2D regions and text-aligned features in 3D space, the proposed method can achieve much better performance than other baselines without using any 3D instance annotation, additional training, or manual heatmap thresholding during inference. This method can also be used for 3D instance pseudo-label generation for self-supervised learning.

A limitation of the proposed method is that it misses very small objects, typically those with less than 50 points, as they are likely to be treated as noise and filtered out during fusion. Furthermore, while the proposed method can improve segmentation quality due to multi-view noisy filtering, it still relies on the 2D segments and features to be accurate enough, since it does not use any 3D data for fine-tuning. A promising direction is to integrate this method with a 3D learning-based method to utilize the scarcer but clean 3D annotations.

References

[1] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *International Conference on Learning Representations*.

[2] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra. Detecting twenty-thousand classes using image-level supervision. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IX*, pages 350–368. Springer, 2022.

[3] F. Liang, B. Wu, X. Dai, K. Li, Y. Zhao, H. Zhang, P. Zhang, P. Vajda, and D. Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. *arXiv preprint arXiv:2210.04150*, 2022.

[4] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[5] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021.

[6] L. Yuan, D. Chen, Y.-L. Chen, N. Codella, X. Dai, J. Gao, H. Hu, X. Huang, B. Li, C. Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

[7] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

[8] D. Rozenberszki, O. Litany, and A. Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[10] J. McCormac, A. Handa, A. Davison, and S. Leutenegger. Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. In *2017 IEEE International Conference on Robotics and automation (ICRA)*, pages 4628–4635. IEEE, 2017.

[11] M. Runz, M. Buffier, and L. Agapito. Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 10–20. IEEE, 2018.

[12] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger. Fusion++: Volumetric object-level slam. In *2018 international conference on 3D vision (3DV)*, pages 32–41. IEEE, 2018.

[13] G. Narita, T. Seno, T. Ishikawa, and Y. Kaji. Panopticfusion: Online volumetric semantic mapping at the level of stuff and things. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4205–4212. IEEE, 2019.

[14] S. Peng, K. Genova, C. M. Jiang, A. Tagliasacchi, M. Pollefeys, and T. Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *CVPR*, 2023.

[15] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. 2018.

[16] X. Zhai, X. Wang, B. Mustafa, A. Steiner, D. Keysers, A. Kolesnikov, and L. Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18123–18133, 2022.

[17] B. Li, K. Q. Weinberger, S. Belongie, V. Koltun, and R. Ranftl. Language-driven semantic segmentation. *arXiv preprint arXiv:2201.03546*, 2022.

[18] G. Ghiasi, X. Gu, Y. Cui, and T.-Y. Lin. Scaling open-vocabulary image segmentation with image-level labels. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI*, pages 540–557. Springer, 2022.

[19] J. Xu, S. De Mello, S. Liu, W. Byeon, T. Breuel, J. Kautz, and X. Wang. Groupvit: Semantic segmentation emerges from text supervision. *arXiv preprint arXiv:2202.11094*, 2022.

[20] A. S. M. N. D. W. A. D. A. M. A. A. M. D. Z. S. X. W. X. Z. T. K. N. H. Matthias Minderer, Alexey Gritsenko. Simple open-vocabulary object detection with vision transformers. *ECCV*, 2022.

[21] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022.

[22] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136. Ieee, 2011.

[23] T. Whelan, S. Leutenegger, R. Salas-Moreno, B. Glocker, and A. Davison. Elasticfusion: Dense slam without a pose graph. Robotics: Science and Systems, 2015.

[24] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics (ToG)*, 36(4):1, 2017.

[25] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison. imap: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6229–6238, 2021.

[26] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[27] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[28] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

[29] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996.

[30] H. Pfister, M. Zwicker, J. Van Baar, and M. Gross. Surfels: Surface elements as rendering primitives. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 335–342, 2000.

[31] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[32] T. Müller, A. Evans, C. Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. doi: 10.1145/3528223.3530127. URL https://doi.org/10.1145/3528223.3530127.

[33] Sara Fridovich-Keil and Alex Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa. Plenoxels: Radiance fields without neural networks. In *CVPR*, 2022.

[34] C. Sun, M. Sun, and H.-T. Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5459–5469, 2022.

[35] X. Kong, S. Liu, M. Taher, and A. J. Davison. vmap: Vectorised object mapping for neural field slam. *arXiv preprint arXiv:2302.01838*, 2023.

[36] N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. *arXiv preprint arXiv:2210.05663*, 2022.

[37] J. Zhang, R. Dong, and K. Ma. Clip-fo3d: Learning free open-world 3d scene representations from 2d dense clip. *arXiv preprint arXiv:2303.04748*, 2023.

[38] D. Hegde, J. M. J. Valanarasu, and V. M. Patel. Clip goes 3d: Leveraging prompt tuning for language grounded 3d recognition. *arXiv preprint arXiv:*, 2022.

[39] K. Jatavallabhula, A. Kuwajerwala, Q. Gu, M. Omama, T. Chen, S. Li, G. Iyer, S. Saryazdi, N. Keetha, A. Tewari, J. Tenenbaum, C. de Melo, M. Krishna, L. Paull, F. Shkurti, and A. Torralba. Conceptfusion: Open-set multimodal 3d mapping. *arXiv*, 2023.

[40] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik. Lerf: Language embedded radiance fields. *arXiv preprint arXiv:2303.09553*, 2023.

[41] R. Ding, J. Yang, C. Xue, W. Zhang, S. Bai, and X. Qi. Language-driven open-vocabulary 3d scene understanding. *arXiv preprint arXiv:2211.16312*, 2022.

[42] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[43] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.