Sparse Multiview Open-Vocabulary 3D Detection

Olivier Moliner[1,2], Viktor Larsson[1] and Kalle Åström[1]
[1] Centre for Mathematical Sciences, Lund University [2] Sony Corporation, Lund Laboratory, Sweden
{olivier.moliner, viktor.larsson, karl.astrom}@math.lth.se

Abstract

The ability to interpret and comprehend a 3D scene is essential for many vision and robotics systems. In numerous applications, this involves 3D object detection, i.e. identifying the location and dimensions of objects belonging to a specific category, typically represented as bounding boxes. This has traditionally been solved by training to detect a fixed set of categories, which limits its use. In this work, we investigate open-vocabulary 3D object detection in the challenging yet practical sparse-view setting, where only a limited number of posed RGB images are available as input. Our approach is training-free, relying on pre-trained, off-the-shelf 2D foundation models instead of employing computationally expensive 3D feature fusion or requiring 3D-specific learning. By lifting 2D detections and directly optimizing 3D proposals for featuremetric consistency across views, we fully leverage the extensive training data available in 2D compared to 3D. Through standard benchmarks, we demonstrate that this simple pipeline establishes a powerful baseline, performing competitively with state-of-the-art techniques in densely sampled scenarios while significantly outperforming them in the sparse-view setting.

1. Introduction

The ability to parse and understand a 3D scene is a prerequisite for many vision or robotic systems. In many applications, this takes the form of 3D object detection, i.e. determining the location and dimension of all objects of a particular category, e.g. as a bounding box. Object detection is a classical problem in computer vision, and is traditionally solved by selecting a discrete set of object categories, which the method is trained to detect. Having a fixed set of labels limits the applicability of the methods and prevents them from generalizing to new problem domains not represented in the label-set without expensive re-training.

Recent advances in 2D visual-language models have allowed for so-called *open-vocabulary* object detection, where the system can be queried with arbitrary labels. The expressive power coming from incorporating the language model also allows for more complex queries, e.g. a lengthier description of an object beyond single labels or even via object affordances (*Where can I sit?*). While this development originally happened in 2D, several methods have been proposed for open-vocabulary 3D object detection using these rich feature representations.

Figure 1. **Open-vocabulary 3D Object Detection.** Our method takes as input a sparse collection of posed RGB images together with a query text prompt. The outputs are 3D bounding boxes corresponding to the prompt. In the figure we include the ground-truth mesh for visualization purposes only.

Existing methods either rely on dense 3D geometry obtained by scanning and reconstructing the scene offline, or perform monocular detection in RGB-D images. To detect new or moved objects, the scene needs to be rescanned, hence many applications requiring continuous monitoring of a scene are not feasible in this setting. Most methods use trained 3D proposal networks to localize objects in the 3D data. They leverage vision-language models either when training the 3D backbone or at inference time by back-projecting language-augmented visual features and matching open-vocabulary 2D detections to class-agnostic 3D masks. However, current 3D datasets are orders of mag-

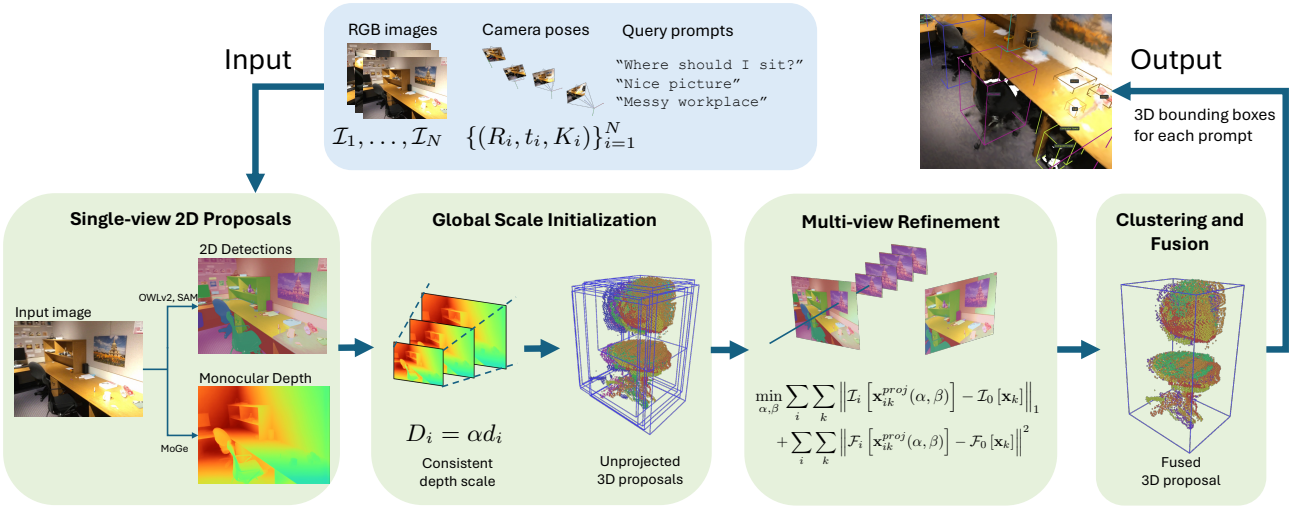

Figure 2. **Overview of SMOV3D.** Our method takes as input a sparse collection of posed RGB images together with a collection of text query prompts. The pipeline then conists of three steps. **i) Monocular 2D Proposals** For each prompt and image we perform 2D detection yielding a set of masks. These are then lifted to 3D using monocular depth. **ii) Multi-view Refinement** Lifted 3D point clouds are refined by optimizing a multi-view featuremetric loss that combines both photometric and CLIP consistency. **iii) 3D Clustering and Fusion.** The optimized 3D point clouds are aggregated in 3D and greedily fused using a simple heuristic. The output is a collection of 3D bounding boxes. For visualization we show them overlayed on the ground-truth mesh.

nitude smaller than the large-scale image datasets used to train 2D foundation models. As a result, the strong generalization capabilities of the 2D models may be lost during training.

In this work, we investigate how effectively 2D foundation models can be leveraged to tackle the task of open-vocabulary 3D object detection from only a sparse set of RGB images, without any 3D-specific training, and introduce a straightforward, training-free baseline. Our approach works by lifting detections from off-the-shelf 2D open-vocabulary and segmentation models into 3D via monocular depth, followed by a multi-view refinement that optimizes for photometric and semantic consistency. Our experiments show that this simple approach is surprisingly powerful, producing comparable results to state-of-the-art methods in densely-sampled scenarios while establishing a strong baseline for the sparse-view setting.

2. Related Work

3D object detection. 3D object detection aims at predicting three-dimensional bounding boxes and object classes from 3D or 2D input data. 3D point clouds are appealing for 3D object detection as they provide accurate geometric information, and point cloud-based methods leveraging deep Hough voting [5, 23, 35, 42] or Transformers [14, 21] have shown remarkable performance in both indoor and outdoor settings. However, these methods require depth sensors or dense image sequences for data acquisition, which may not be practical due to cost or power consumption constraints.

Early methods for 3D object detection from RGB images extend conventional 2D detectors to lift monocular detections to 3D [16, 24, 30, 31], but their performance suffers from the lack of explicit depth information. Recently, methods leveraging multiple views to better capture scene geometry have gained increasing attention. Extending DETR [3] to the 3D domain, Transformer-based methods [13, 33, 37, 38] predict bounding boxes by attending to multi-view features. Feature volume-based methods [29, 34] perform 3D object detection in a voxel-based 3D feature volume [22], while recent work leverages Neural Radiance Fields (NeRF) [19] to model geometry implicitly in the feature volume [39].

Despite their proven performance, current state-of-the-art 3D object detectors, whether 3D or 2D-based, are closed-set methods trained to detect a limited number of predefined object categories. Extending these models to new domains requires collecting and annotating new data and retraining the models, which is both costly and time-consuming.

Open-vocabulary 2D and 3D object detection. Open-vocabulary object detection is an emerging field in computer vision that aims to localize and identify arbitrary, previously unseen objects. The ideas of going from closed to open vocabulary tasks emerged with the advent of better foundation models, for example GPT for text, [2], CLIP for image-text [25], and Lidar-CLIP for image-text-lidar [10].

Such models have been shown to be important for a large variety of few- or single-shot learning and open vocabulary tasks. For example CLIP opened up for open-ended queries primarily for classification and action recognition, but later open-vocabulary 2D Object detection was explored and developed, e.g. in [4, 26, 43, 44].

Open-vocabulary 3D object detection is still in its infancy. Existing open-vocabulary 3D detection models usually operate on point-cloud or RGB-D data. OV-3DETIC [17] expands a 3D object detector's vocabulary using ImageNet1K [7] and uses contrastive learning to transfer knowledge between image and point cloud modalities. OV-3DET [18] generates pseudo-annotations using a pre-trained 2D open-vocabulary detector [45] to train a 3D detector to localize objects. Object2Scene [46] proposes a point-cloud object detector (L3Det) trained on a 3D dataset augmented by inserting 3D objects and corresponding text descriptions. It leverages cross-domain contrastive learning to mitigate the domain gap between scene and inserted objects. FM-OV3D [41] blends knowledge from multiple pre-trained foundation models to improve the open-vocabulary localization and recognition abilities of its 3D detection model. OpenIns3D [11] casts the problem as an extension of open-vocabulary semantic segmentation. Its "Snap" module generates synthetic images from point clouds and uses 2D vision-language models to detect objects in 2D based on text prompts, while its "Lookup" module matches the 2D detections to class-agnosic 3D point clounds predicted by the "Mask" module. ImOV3D [40] addresses the scarcity of annotated 3D datasets by generating pseudo 3D point clouds and annotations from 2D datasets to train an open-vocabulary point cloud detector.

The reliance on point cloud data, whether obtained by depth sensors or dense image sequences, limits the applicability of open-vocabulary 3D detection, and the need for trained 3D proposal networks raises the question of the ability of existing methods to generalize to new domains. In this work, we leverage pre-trained foundation models to introduce a straightforward, training-free method for open-vocabulary 3D object detection using only sparse, multi-view RGB images as input.

3. Method

We now present our method for open-vocabulary 3D object detection which we call **S**parse **M**ulti-view **O**pen-**V**ocabulary **3**D **D**etection (SMOV3D). Our method takes as input a collection of RGB images $\{\mathcal{I}_i\}_{i=1}^m$ together with poses and intrinsics $\{(R_i, t_i, K_i)\}_{i=1}^m$, as well as query text prompts. For each of the prompts, the method then consists of three steps:

- For each image we generate a collection 2D proposals which are lifted to a camera-centric point cloud using monocular depth estimation. (Sec. 3.1)
- Each proposal is then refined by considering a multi-view featuremetric consistency. (Sec 3.2)
- The proposals are then robustly clustered in 3D and fused to generate the final result. (Sec. 3.3)

The method returns a collection of 3D bounding boxes. For each prompt, multiple bounding boxes can be returned if there are multiple instances present in the scene. Figure 2 shows an overview of our method and in the next sections we detail each of the three steps.

3.1. Single-view Proposal Generation

In a first step, we generate initial 3D object proposals for each 2D view.

In each image, we generate a collection of 2D object bounding boxes using a state-of-the-art 2D open-vocabulary detector (OWLv2 [20]) queried with each prompt. These bounding boxes are used as input to an image segmentation model (Segment Anything [12]) to produce accurate 2D masks. Finally, we lift each 2D mask to 3D using an affine-invariant monocular depth estimator (MoGe [36]).

As monocular depth estimators tend to oversmooth edges, we first filter out depth values with high gradients (i.e. where normals are close to orthogonal with the viewing direction). Due to image noise, or to the presence of several similar objects in front of each other, the 2D masks generated by SAM sometimes contain parts of different objects. To separate erroneously merged objects and remove background points, we cluster the 3D proposals with DBSCAN [9]. We treat large clusters as new proposals, and remove small clusters and outlier points.

3.2. Multi-view Proposal Refinement

The monocular depth estimator predicts relative depth maps that are invariant to affine (scale and shift) transformations. Moreover, the depth maps are often only locally consistent, thus even if the optimal global shift and scale parameters are estimated, the depths of individual objects may be inaccurate. To retrieve accurate 3D positions and sizes for the 3D proposals, we first find an initial global scale factor for each input image using multi-view semantic consistency. We then proceed to refine each 3D mask by optimizing individual scale and shift parameters.

3.2.1. Multi-view Consistency of 3D Proposals

To accurately estimate the 3D positions of the detected objects we require a multi-view consistency loss, measuring how much the backprojected proposals from each image are supported by the other images.

Let $\{\mathbf{x}_k\}_{k=1}^N$ be the pixels belonging to a proposal in the image $\mathcal{I}_0$. These are lifted to 3D in the camera coordinate system by backprojecting using the depth as

$$\mathbf{X}_k^{cam} = (\alpha d_k + \beta) K_0^{-1} \mathbf{x}_k, \tag{1}$$

where d_k is the original mono-depth depth value, and α, β are the shift-scale parameters which we aim to recover. Using the known camera poses and intrinsics, the lifted 3D point can be projected into the ith view $\mathcal{I}_i$ as

$$\mathbf{x}_{ik}^{proj} = \Pi\left(K_i(R_i R_0^T(\mathbf{X}_k^{cam} - t_0) + t_i)\right), \quad (2)$$

where $\Pi : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ is the pinhole projection function. Note that $\mathbf{x}_{ik}^{proj}$ is now a function of scale α and shift β.

To measure consistency across images we introduce a photo-consistency loss as

$$\mathcal{L}_{rgb}(\alpha, \beta) = \sum_i \sum_k \left\| \mathcal{I}_i\left[\mathbf{x}_{ik}^{proj}(\alpha, \beta)\right] - \mathcal{I}_0\left[\mathbf{x}_k\right] \right\|_1, \quad (3)$$

where $\mathcal{I}_i[\mathbf{x}] \in \mathbb{R}^3$ denotes the image $\mathcal{I}_i$ (bi-linearly) interpolated at the pixel position $\mathbf{x}$. The loss is normalized over the reprojected points visible in $\mathcal{I}_i$. To improve robustness to viewpoint changes and other effects causing photometric inconsistencies across images we also include a CLIP-based consistency term in the optimization. For each image we compute a dense CLIP feature map,

$$\mathcal{F}_i = \text{CLIP}(\mathcal{I}_i) \quad (4)$$

and then define the reprojected CLIP loss as

$$\mathcal{L}_{sim}(\alpha, \beta) = \sum_i \sum_k \left\| \mathcal{F}_i\left[\mathbf{x}_{ik}^{proj}(\alpha, \beta)\right] - \mathcal{F}_0\left[\mathbf{x}_k\right] \right\|^2. \quad (5)$$

This term will ensure alignment to similar semantic image content in cases where the photometric loss is insufficient.

For each 3D proposal, the full consistency measure is then a weighted combination of these two,

$$\mathcal{L}(\alpha, \beta) = \mathcal{L}_{rgb}(\alpha, \beta) + \lambda \mathcal{L}_{sim}(\alpha, \beta), \quad (6)$$

where λ is the trade-off hyperparameter.

3.2.2. Global Scale Initialization

In the first step of the shift-and-scale estimation we estimate a global scaling parameter α_{global} which is used to initialize the scale α for each proposal in the second step.

To do this we jointly consider all object proposals and select α_{global} by sampling. Let $\alpha_1, \ldots, \alpha_m$ be m scales uniformly sampled in the range $[\alpha_{min}, \alpha_{max}]$, and let $\mathcal{L}_k$ denote the loss (6) for the kth proposal. We then take

$$\alpha_{global} = \arg\min \left\{ \sum_k \mathcal{L}_k(\alpha, 0) \;\middle|\; \alpha \in \{\alpha_1, \ldots, \alpha_k\} \right\}. \quad (7)$$

As we are only interested in a coarse scale estimate, we only consider at most M pixels across all proposals to speed up the process. The motivation for sampling within the detected 2D masks instead of anywhere in the image is to avoid sampling in background areas, e.g. walls, ceiling, sky, which are not as helpful for estimating multi-view consistency.

3.2.3. Per-mask Refinement

The previous step yields an initial estimate of the scale parameter α, but to allow for errors in the depth map, we now optimize an independent α and β for each proposal. This allows us to handle cases where the depth map is not globally consistent across different scene elements, which is supported by our ablations (Section 4.5). Each proposal is then refined by minimizing

$$\min_{\alpha,\beta} \; \mathcal{L}(\alpha, \beta), \quad (8)$$

initialized with α_{global} and zero. The optimization is performed using gradient descent. To speed up the optimization, we randomly sample a subset of the mask pixels at each iteration. Note that as each proposal is optimized independently, this can be done in parallel.

3.3. Clustering and Fusion

For each image and prompt, we now have a collection of 3D proposals (point clouds). The last step of the pipeline is now to combine these into the final output bounding boxes.

For this, we use a simple sequential clustering approach. First, we compute axis-aligned 3D bounding boxes for all proposals. These are then greedily merged based on their intersection-over-union (IoU). Finally, for each merged cluster we compute a bounding box for the union of the point-clouds.

4. Experiments

4.1. Experimental setup

Datasets. The ScanNet dataset [6] comprises 1,201 training and 312 validation scenes. We evaluate our method on the ScanNet10 and ScanNet20 categories defines by Lu *et al*. [17, 18]. To demonstrate the open-vocabulary capabilities of our method, we show results on the ScanNet200 benchmark [28], in which the 200 most represented object classes of ScanNet are split into 3 subsets based on the frequency of the number of labeled surface points in the training set: *head* (66 classes), *common* (68 classes) and *tail* (66 classes). We also experiment with Replica [32], a dataset of photo-realistic 3D indoor scenes reconstructed from RGB-D scans, which contains 48 object classes. To demonstrate the ability of our method to handle arbitrary queries, we present qualitative results on data from the OpenSUN3D challenge [8], featuring ARKitScenes scans [1] with long-tail prompts.

Evaluation protocol. We follow the class splits of prior works [11, 18, 28], without using "seen" classes. We compute axis-aligned bounding boxes from ground-truth segmentations, following [11]. We report the performance on the validation sets using the mean Average Precision at an IoU threshold of 0.25, denoted mAP_{25}.

Method	GT Depth	3D proposal	Mean	toilet	bed	chair	sofa	dresser	table	cabinet	bookshelf	pillow	sink
OV-3DETIC [17]	✓	3DETR†	12.7	49.0	2.6	7.3	18.6	2.8	14.3	2.4	4.5	3.9	21.1
Object2Scene [46]	✓	L3DET†	24.6	56.3	36.2	16.1	23.0	8.1	23.1	14.7	17.3	23.4	27.9
FM-OV3D [41]	✓	3DETR†	21.5	55.0	38.8	19.2	41.9	23.8	3.5	0.4	6.0	17.4	8.8
OpenIns3D [11]	✓	Mask3D†	**43.7**	79.5	**70.5**	**76.9**	15.8	0.0	**53.1**	**40.1**	**41.2**	7.1	53.1
SMOV3D (Ours) RGB-D	✓	-	42.2	**83.5**	29.9	29.8	**73.5**	**25.8**	27.8	4.1	28.4	**61.1**	**58.6**
SMOV3D (Ours)	✗	-	28.9	61.8	25.9	12.2	61.0	16.2	20.0	1.0	23.3	36.1	31.8

Table 1. **Open-vocabulary Object Detection on ScanNet10.** We compare our method to point cloud-based methods (mAP_{25}, %).

Methods	GT depth	Mean	toilet	bed	chair	sofa	dresser	table	cabinet	bookshelf	pillow	sink	bathtub	refrigerator	desk	night stand	counter	door	curtain	box	lamp	bag
CLIP-3D [25]	✓	12.7	44.8	23.8	17.5	12.6	4.9	13.2	1.9	4.0	11.4	17.6	32.2	14.9	11.4	2.4	0.5	14.5	8.6	7.5	5.1	4.7
OV-3DET [18]	✓	18.0	57.3	42.3	27.1	31.5	8.2	14.2	3.0	5.6	23.0	31.6	56.3	11.0	19.7	0.8	0.3	9.6	10.5	3.8	2.1	2.7
OpenIns3D [11]	✓	**37.1**	79.5	**70.5**	**76.9**	15.8	0.0	**53.1**	**40.1**	**41.2**	7.1	53.1	14.3	**32.1**	29.1	4.8	**55.6**	**40.4**	**41.1**	2.6	**48.0**	6.2
SMOV3D (Ours) RGB-D	✓	36.2	**85.0**	28.7	29.9	**74.2**	**28.9**	20.1	3.8	26.3	**60.9**	**60.4**	**65.9**	27.5	**32.8**	**53.4**	14.4	22.5	15.0	**14.6**	13.7	**45.8**
SMOV3D (Ours)	✗	21.7	54.5	29.0	12.0	59.9	19.1	16.3	1.0	21.0	35.7	25.7	36.3	15.1	22.4	33.7	7.1	2.6	7.8	10.7	5.5	19.4

Table 2. **Open-vocabulary 3D Object Detection on ScanNet20.** (mAP_{25}, %).

Implementation details. For each scene, we sample 2D views such that each frame has a relative translation greater than $0.5m$ or a relative rotation angle greater than $15°$ from any previously sampled frame. Unless otherwise stated, we then sample at most 32 random views to cover a scene. We present results averaged over three random seeds for sampling the views. We use OWLv2 [20] as open-vocabulary 2D object detector, and SAM2 [12, 27] to produce the 2D masks. For monocular depth we use MoGe [36]. To extract dense CLIP feature maps, we use the MaskCLIP [44] reparametrization trick with CLIP ViT-L/14. During depth refinement, we use the AdamW optimizer [15] with a learning rate of 0.005 and sample 100 points randomly at each iteration. The weight λ is set to 1.0. We tuned our method's hyperparameters on a subset of ScanNet's training set.

Baselines. Since no previous work has tackled open-vocabulary 3D object detection from sparse multi-view 2D images, we compare our method to state-of-the-art point cloud-based methods: OpenIns3D [11], FM-OV3D [41], Object2Scene [46], OV-3DETIC [17] and OV-3DET [18]. For comparison, we also evaluate our method using the depth maps provided by the datasets. In this setting, we use the same view sampling as for our RGB-based method, and do not perform depth refinement.

4.2. Quantitative Comparison

ScanNet. We first evaluate our method on the ScanNet10 and ScanNet20 benchmarks. As can be seen in Tab. 1 and Tab. 2, when using ground-truth depth maps our method is comparable to OpenIns3D and surpasses all other point cloud-based methods. In this setting, our method is very simple, as it only involves backprojecting 2D mask proposals. Conversely, all other point cloud-based methods use a 3D box or mask proposal network trained on ScanNet.

Method	GT depth	mAP_{25}
OpenIns3D [11] point-cloud + Snap	✓	21.1
OpenIns3D [11] point-cloud + RGB-D	✓	32.9
SMOV3D (Ours) RGB-D	✓	**38.9**
SMOV3D (Ours)	✗	29.3

Table 3. **Open-vocabulary Object Detection on Replica.** The performance of our method on Replica is comparable to its performance on ScanNet, demonstrating its ability to generalize. Our approach has significantly higher mAP than OpenIns3D using known depth either from RGB-D or from ground truth point-clouds. Even without ground truth depth, it is competitive.

Somewhat surprinsingly, when using only sparse 2D views, our method still surpasses most existing point cloud-based methods. Some are monocular methods [17, 18, 46] that rely on pseudo-3D annotations constrained to the view frustum during training. As ScanNet images have a narrow field of view, many objects are truncated and may have varying shapes during training, which may lead to lower performance. As MoGe was not trained on ScanNet, our method is the only truly zero-shot method in this benchmark.

Replica. We also evaluate our method on the Replica dataset, using the same hyperparameters as for ScanNet, to confirm its ability to generalize. Although there are no previous results for open-vocabulary 3D object detection on Replica, OpenIns3D is easily adaptable to this task, by converting generated masks into axis-aligned bounding boxes. We also evaluated OpenIns3D with RGB-D data, using the same settings used for open-vocabulary instance segmentation on Replica in the original implementation. In this setting, OpenIns3D takes as input the point cloud and 200

Method	Head	Common	Tail
Object2Scene [46]	-	10.1	3.4
OpenIns3D [11] with RGB-D	**25.6**	20.4	16.5
SMOV3D (Ours) RGB-D	23.2	**25.9**	**33.2**
SMOV3D (Ours)	13.3	18.2	16.7

Table 4. **Open-vocabulary Object Detection on ScanNet 200.** We use the whole ScanNet200 vocabulary as prompt (except *'wall'* and *'floor'*), and present results (mAP_{25}) for the *Head*, *Common* and *Tail* category splits. Our method performs just as well on long-tail classes (Tail) as on the most frequent ones (Head).

RGB-D images per scene.

As shown in Tab. 3, the performance of our method on Replica is comparable to its performance on ScanNet, demonstrating its ability to generalize. On the other hand, OpenIns3D's performance dropped. This decrease is compatible with the decrease observed on the segmentation task in the original paper, and might be due to the mask proposal model's struggle to generalize to Replica.

Long-tail 3D Object Detection. To study our method's zero-shot generalization ability, we evaluate it on the ScanNet200 benchmark. During evaluation, we use the whole ScanNet200 vocabulary (except, as is usual, *'wall'* and *'floor'*), and present results for the *Head*, *Common* and *Tail* category splits. We compare with Object2Scene [46], which used the *Head* as seen classes, and OpenIns3D. Following the authors' advice, we only present results for OpenIns3D with RGB-D data. The results can be seen in Tab. 4. SMOV3D achieves strong performance on tail categories, even surpassing its performance on common categories. While counter-intuitive, this can be explained by our training-free approach. Unlike methods trained or fine-tuned on 3D datasets, SMOV3D relies solely on the generalization of the 2D foundation models and does not inherit the dataset's label frequency biases.

4.3. Qualitative Results

In Fig. 1, we show an example of 3D object detection with a free-text prompt (*"Playing music"*) in a ScanNet scene. The figure also shows posed cameras and 2D mask proposals. In Fig. 3, we visualize 3D bounding boxes predicted by our method when prompted with the ScanNet10 categories.

Figure 5 presents qualitative results using scenes and queries from the OpenSUN3D challenge [8], which features ARKitScenes [1] scans paired with long-tail prompts. It demonstrates the ability of our method to perform zero-shot detection from free-form text queries in realistic scenes.

4.4. Using Few Views

Current open-vocabulary 3D detection benchmarks rely either on 3D scenes reconstructed offline, or on monocular RGB-D images following a scanning trajectory. We argue that these approaches are ill-suited for many practical use cases, as detecting new or moved objects would require a complete rescan of the scene. A more realistic setting for many applications such as facility management or retail is to consider systems with few fixed, pre-calibrated cameras, delivering predictions at regular intervals.

To address this gap, we emulate a fixed-camera installation in the Replica dataset by placing 4 cameras at room corners near the ceiling and recursively placing a camera between each pair to obtain 8 and 16 views. We compare to OpenIns3D using the same views, and to OpenIns3D with Snap & Lookup. Note that OpenIns3D still uses the full point cloud for mask proposal, including points that are not visible in any of the views.

SMOV3D takes 15.6, 26.7 and 50.5 seconds to perform full-scene detection on an RTX 4090 with 4, 8 and 16 RGB views respectively. As shown in Fig. 4, with as few as 4 RGB images, our method significantly outperforms the point-cloud-based OpenIns3D. We also present qualitative results in Fig. 6, showing camera placement and predicted bounding boxes.

This experiment shows that this simple approach is well-suited for applications requiring continuous monitoring under practical constraints.

4.5. Ablation Study

In Tab. 5, we analyze key components of our method on ScanNet10. With global scale initialization only, i.e. estimating one scale parameter for each view, our method, though simple, already achieves results comparable to point cloud-based methods. Refining the depth maps for each object mask separately using any combination of $\mathcal{L}_{rgb}$ and $\mathcal{L}_{sim}$ provides an improvement. We believe this is due to the depth maps only being locally consistent.

We also implemented a depth refinement method minimizing a multi-view depth consistency loss $\mathcal{L}_{depth}$, defined as the L_1 loss between the estimated depth of proposal points in one view and the depths of their reprojections in other views. This leads to degraded results, further showing that the initial monocular depth maps are geometrically inconsistent.The best performance is obtained by refining the depth maps for each object mask separately using both the photometric loss $\mathcal{L}_{rgb}$ and the CLIP similarity loss $\mathcal{L}_{sim}$, yielding a 12% relative improvement in the mAP_{25} metric.

5. Limitations and Future Work

Under severe occlusions, some objects are only visible in one view. While global depth initialization provides some

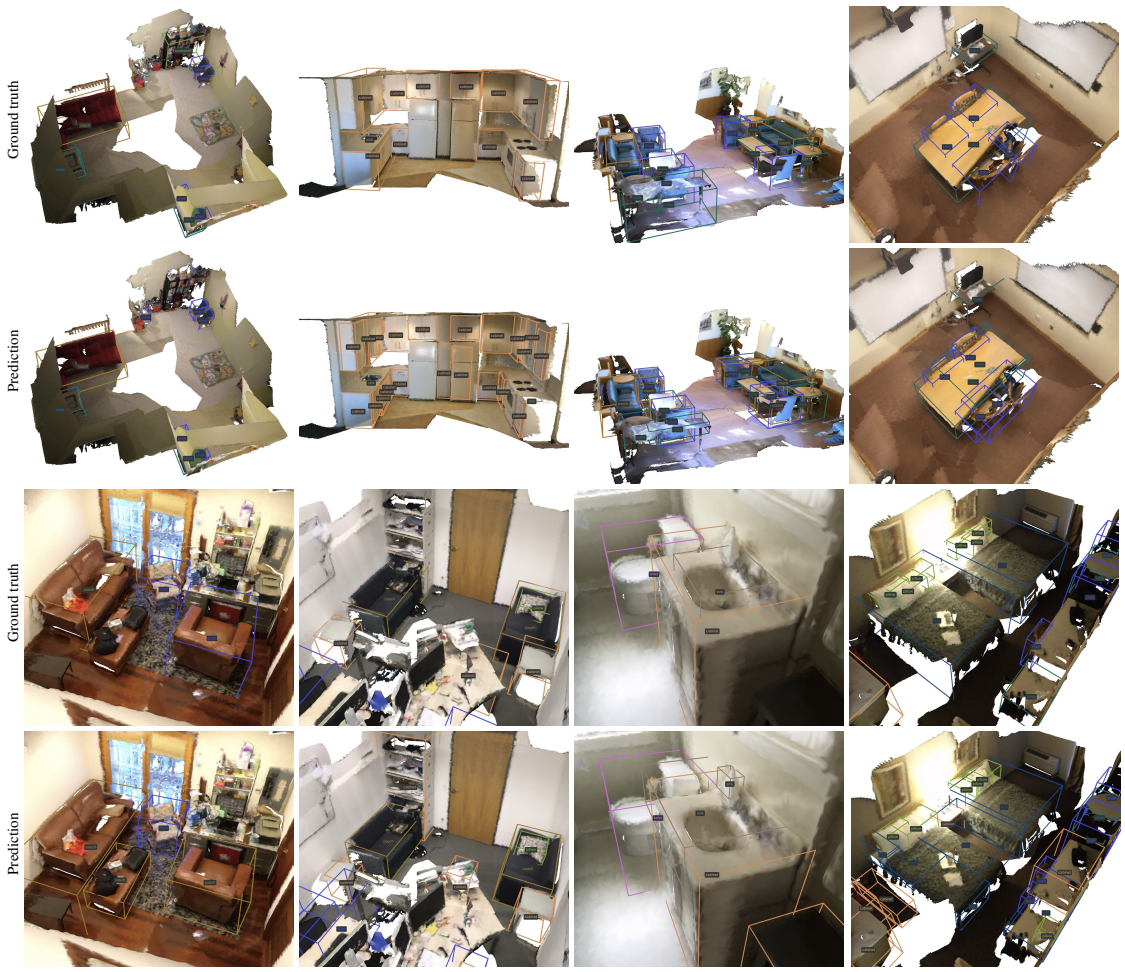

Figure 3. **Qualitative results on ScanNet.** Zero-shot 3D object detection, using the ScanNet10 categories as prompts. Best seen on screen.

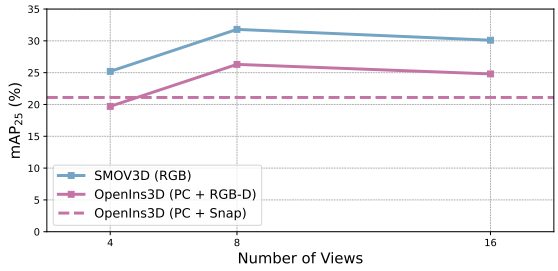

Figure 4. **Few views.** We report mAP_{25} scores (%) on the Replica dataset with few views chosen so that the entire scene is visible, and compare to OpenIns3D.

Depth refinement	$\mathcal{L}_{depth}$	$\mathcal{L}_{rgb}$	$\mathcal{L}_{sim}$	mAP_{25}
Global scale			✓	25.7
Per-mask	✓			24.1
		✓		26.1
			✓	27.8
		✓	✓	**28.9**

Table 5. **Ablation on different aspects of depth refinement.** Evaluated on ScanNet10 (mAP_{25}, %). Per-mask depth refinement using the photometric and CLIP consistency losses yields the best results.

robustness, severe cases can lead to mis-localization during per-mask refinement.

Due to motion blur, occlusions or bad lighting conditions, some objects may be classified differently across views, causing overlapping 3D detections. Some occurrences of this problem can be seen in Fig. 3 and Fig. 6, e.g. for the "chair" and "couch" classes. Future work could incorporate more sophisticated fusion logic.

Our method relies on accurate camera poses for back-projection. Its performance may degrade with noisy camera calibrations, a factor we have not explored in this work.

There remains a performance gap between our RGB-

Figure 5. **Qualitative results on ARKitScenes.** Zero-shot 3D object detection, using the prompts proposed in the OpenSUN3D challenge.

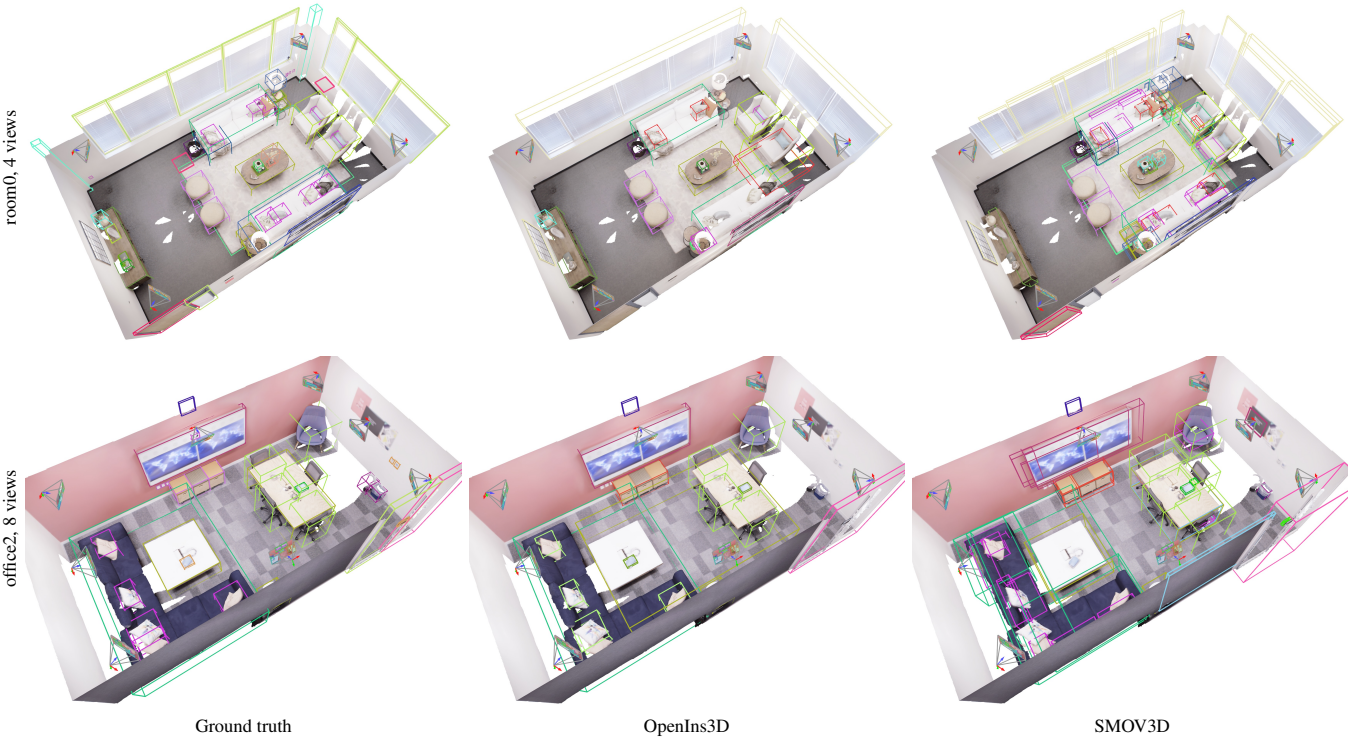

Figure 6. **Qualitative results with few views results on Replica.** Zero-shot 3D object detection, using the Replica categories as prompts.

only method and methods using ground-truth depth. While SMOV3D narrows this gap significantly, improving monocular depth estimation or developing new refinement strategies to close it further are promising directions for future research.

6. Conclusion

In this work, we have conducted an in-depth study on the effectiveness of 2D foundation models for open-vocabulary 3D detection from sparse RGB views. Our proposed baseline, SMOV3D, demonstrates that a straightforward, training-free approach can achieve highly competitive results, particularly in challenging sparse-view and long-tail scenarios where trained methods may struggle. Our work shows that dense 3D geometry is not always a prerequisite for accurate 3D perception, and that the generalized knowledge embedded in 2D foundation models represents a powerful and practical resource that can be leveraged directly. Further, by not having any learned 3D component, our method does not rely on having access to 3D data for training or fine-tuning, making it easy to apply in new settings.

Acknowledgment

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

[1] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Yuri Feigin, Peter Fu, Thomas Gebauer, Daniel Kurz, Tal Dimry, Brandon Joffe, Arik Schwartz, and Elad Shulman. ARKitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 4, 6

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, pages 1877–1901. Curran Associates, Inc., 2020. 2

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *ArXiv*, abs/2005.12872, 2020. 2

[4] Xi Chen, Shuang Li, Ser-Nam Lim, Antonio Torralba, and Hengshuang Zhao. Open-vocabulary Panoptic Segmentation with Embedding Modulation . In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1141–1150, Los Alamitos, CA, USA, 2023. IEEE Computer Society. 3

[5] Bowen Cheng, Lu Sheng, Shaoshuai Shi, Ming Yang, and Dong Xu. Back-tracing representative points for voting-based 3d object detection in point clouds. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8959–8968, 2021. 2

[6] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 4

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 3

[8] Francis Engelmann, Ayça Takmaz, Jonas Schult, Elisabetta Fedele, Johanna Wald, Songyou Peng, Xi Wang, Or Litany, Siyu Tang, Federico Tombari, Marc Pollefeys, Leonidas J. Guibas, Hongbo Tian, Chunjie Wang, Xiaosheng Yan, Bingwen Wang, Xuanyang Zhang, Xiao Liu, Phuc Nguyen, Khoi Nguyen, Anh Tran, Cuong Pham, Zhening Huang, Xiaoyang Wu, Xi Chen, Hengshuang Zhao, Lei Zhu, and Joan Lasenby. Opensun3d: 1st workshop challenge on open-vocabulary 3d scene understanding. *CoRR*, abs/2402.15321, 2024. 4, 6

[9] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, pages 226–231, 1996. 3

[10] Georg Hess, Adam Tonderski, Christoffer Petersson, Kalle Åström, and Lennart Svensson. Lidarclip or: How i learned to talk to point clouds. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7438–7447, 2024. 2

[11] Zhening Huang, Xiaoyang Wu, Xi Chen, Hengshuang Zhao, Lei Zhu, and Joan Lasenby. Openins3d: Snap and lookup for 3d open-vocabulary instance segmentation. *European Conference on Computer Vision*, 2024. 3, 4, 5, 6

[12] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 3, 5

[13] Yingfei Liu, Tiancai Wang, X. Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. *ArXiv*, abs/2203.05625, 2022. 2

[14] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2929–2938, 2021. 2

[15] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 5

[16] Yan Lu, Xinzhu Ma, Lei Yang, Tianzhu Zhang, Yating Liu, Q. Chu, Junjie Yan, and Wanli Ouyang. Geometry uncertainty projection network for monocular 3d object detection. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3091–3101, 2021. 2

[17] Yuheng Lu, Chenfeng Xu, Xiaobao Wei, Xiaodong Xie, Masayoshi Tomizuka, Kurt Keutzer, and Shanghang Zhang. Open-vocabulary 3d detection via image-level class and debiased cross-modal contrastive learning. *arXiv pre-print*, 2022. 3, 4, 5

[18] Yuheng Lu, Chenfeng Xu, Xiaobao Wei, Xiaodong Xie, Masayoshi Tomizuka, Kurt Keutzer, and Shanghang Zhang. Open-vocabulary point-cloud object detection without 3d annotation. In *CVPR*, 2023. 3, 4, 5

[19] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pages 405–421. Springer, 2020. 2

[20] Matthias Minderer, Alexey A. Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 3, 5

[21] Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2886–2897, 2021. 2

[22] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *European Conference on Computer Vision*, 2020. 2

[23] Charles R. Qi, Or Litany, Kaiming He, and Leonidas J. Guibas. Deep hough voting for 3d object detection in point

clouds. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9276–9285, 2019. 2

[24] Zengyi Qin, Jinglu Wang, and Yan Lu. MonoGRNet: A geometric reasoning network for monocular 3d object localization. In *AAAI Conference on Artificial Intelligence*, 2018. 2

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763, 2021. 2, 5

[26] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18082–18091, 2022. 3

[27] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5

[28] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *ECCV*, 2022. 4

[29] Danila D. Rukhovich, Anna Vorontsova, and Anton Konushin. Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1265–1274, 2021. 2

[30] Xuepeng Shi, Qianru Ye, Xiaozhi Chen, Chuangrong Chen, Zhixiang Chen, and Tae-Kyun Kim. Geometry-based distance decomposition for monocular 3d object detection. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15152–15161, 2021. 2

[31] Andrea Simonelli, Samuel Rota Bulò, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1991–1999, 2019. 2

[32] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The replica dataset: A digital replica of indoor spaces. In *arXiv*, 2019. 4

[33] Ching-Yu Tseng, Yi-Rong Chen, Hsin-Ying Lee, Tsung-Han Wu, Wen-Chin Chen, and Winston H. Hsu. Crossdtr: Cross-view and depth-guided transformers for 3d object detection. *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4850–4857, 2022. 2

[34] Tao Tu, Shun-Po Chuang, Yu-Lun Liu, Cheng Sun, Kecheng Zhang, Donna Roy, Cheng-Hao Kuo, and Min Sun. Imgeonet: Image-induced geometry-aware voxel representation for multi-view 3d object detection. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6973–6984, 2023. 2

[35] Haiyang Wang, Shaoshuai Shi, Ze Yang, Rongyao Fang, Qishu Qian, Hongsheng Li, Bernt Schiele, and Liwei Wang. Rbgnet: Ray-based grouping for 3d object detection. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1100–1109, 2022. 2

[36] Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5261–5271, 2025. 3, 5

[37] Yue Wang, Vitor Campanholo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. *ArXiv*, abs/2110.06922, 2021. 2

[38] Yiming Xie, Huaizu Jiang, Georgia Gkioxari, and Julian Straub. Pixel-aligned recurrent queries for multi-view 3d object detection. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 18324–18334, 2023. 2

[39] Chenfeng Xu, Bichen Wu, Ji Hou, Sam Tsai, Ruilong Li, Jialiang Wang, Wei Zhan, Zijian He, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Nerf-det: Learning geometry-aware volumetric representation for multi-view 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23320–23330, 2023. 2

[40] Timing Yang, Yuanliang Ju, and Li Yi. Imov3d: Learning open-vocabulary point clouds 3d object detection from only 2d images. *NeurIPS 2024*, 2024. 3

[41] Dongmei Zhang, Chang Li, Ray Zhang, Shenghao Xie, Wei Xue, Xiaodong Xie, and Shanghang Zhang. Fm-ov3d: Foundation model-based cross-modal knowledge blending for open-vocabulary 3d detection. In *AAAI*, 2024. 3, 5

[42] Zaiwei Zhang, Bo Sun, Haitao Yang, and Qi-Xing Huang. H3dnet: 3d object detection using hybrid geometric primitives. In *European Conference on Computer Vision*, 2020. 2

[43] Zhuowen Tu Zheng Ding, Jieke Wang. Open-vocabulary universal image segmentation with maskclip. In *International Conference on Machine Learning*, 2023. 3

[44] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *European Conference on Computer Vision*, pages 696–712. Springer, 2022. 3, 5

[45] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Phillip Krahenbuhl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. *ArXiv*, abs/2201.02605, 2022. 3

[46] Chenming Zhu, Wenwei Zhang, Tai Wang, Xihui Liu, and Kai Chen. Object2scene: Putting objects in context for open-vocabulary 3d detection. In *arXiv*, 2023. 3, 5, 6