

# 3D-MOOD: Lifting 2D to 3D for Monocular Open-Set Object Detection

Yung-Hsu Yang<sup>1</sup> Luigi Piccinelli<sup>1</sup> Mattia Segu<sup>1</sup> Siyuan Li<sup>1</sup> Rui Huang<sup>1,2</sup>  
 Yuqian Fu<sup>3</sup> Marc Pollefeys<sup>1,4</sup> Hermann Blum<sup>1,5</sup> Zuria Bauer<sup>1</sup>

<sup>1</sup>ETH Zürich <sup>2</sup>Tsinghua University <sup>3</sup>INSAIT, Sofia University <sup>4</sup>Microsoft <sup>5</sup>University of Bonn

## Abstract

Monocular 3D object detection is valuable for various applications such as robotics and AR/VR. Existing methods are confined to closed-set settings, where the training and testing sets consist of the same scenes and/or object categories. However, real-world applications often introduce new environments and novel object categories, posing a challenge to these methods. In this paper, we address monocular 3D object detection in an open-set setting and introduce the first end-to-end 3D Monocular Open-set Object Detector (**3D-MOOD**). We propose to lift the open-set 2D detection into 3D space through our designed 3D bounding box head, enabling end-to-end joint training for both 2D and 3D tasks to yield better overall performance. We condition the object queries with geometry prior and overcome the generalization for 3D estimation across diverse scenes. To further improve performance, we design the canonical image space for more efficient cross-dataset training. We evaluate 3D-MOOD on both closed-set settings (*Omni3D*) and open-set settings (*Omni3D* → *Argoverse 2*, *ScanNet*), and achieve new state-of-the-art results. Code and models are available at [royyang0714.github.io/3D-MOOD](https://royyang0714.github.io/3D-MOOD).

## 1. Introduction

Monocular 3D object detection (3DOD) aims to recognize and localize objects in 3D space from a single 2D image by estimating their 3D positions, dimensions, and orientations. Unlike stereo or LiDAR-based methods, monocular 3DOD relies solely on visual cues, making it significantly more challenging yet cost-effective for robotics and AR/VR applications [10, 16, 36, 53, 65].

While many methods [22, 28, 44, 49, 51, 55, 63] focus on improving 3DOD performance in specific domains, Cube R-CNN [4] and Uni-MODE [23] build unified models on the cross-dataset benchmark *Omni3D* [4], which consolidates six diverse 3D detection datasets [1, 2, 5, 14, 43, 47]. These advancements have driven the evolution of 3DOD

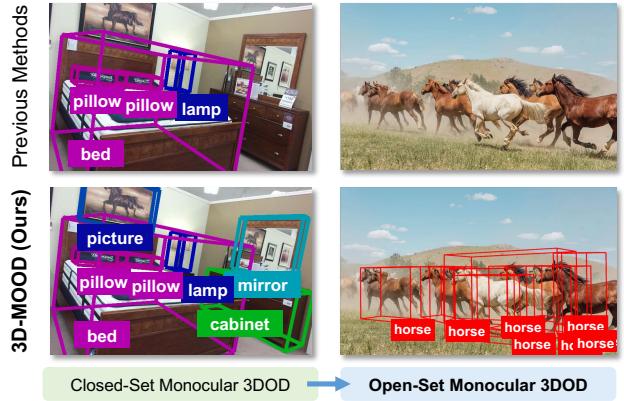


Figure 1. **Open-set Monocular 3D Object Detection.** Unlike previous methods focusing on achieving good results in the closed-set setting, we aim to resolve the open-set monocular 3D object detection problem. This challenge requires the model to classify arbitrary objects while precisely localizing them in unseen scenes.

from specialized models to more unified frameworks. However, as shown in Fig. 1, most existing methods, including the unified models, operate under an ideal assumption: the training set and testing set share identical scenes and object categories. This limits their generalizability in real-world applications for not being able to detect novel objects in unseen domains. This challenge motivates us to explore monocular open-set 3D object detection, further pushing the boundaries of existing 3DOD methods.

The first step towards the open-set monocular 3DOD is identifying the fundamental obstacles underlying this task. Our key observations are as follows: 1) **Cross-modality learning** is crucial to breaking the limitation of closed vocabulary for novel class classification [41]. However, 3D data lacks rich visual-language pairs, making it significantly more challenging to learn modality alignment and achieve satisfactory open-set results. 2) **Robust depth estimation** is essential for monocular 3DOD to generalize well across different scenes compared to LiDAR-based methods [57]. However, monocular depth estimation particularly in novel scenes, is inherently challenging for existing methods.

Given the scarcity of 3D data and text pairs, we propose to bridge the modality gap by lifting open-set 2D detection into open-set 3D detection. Fortunately, recent universal metric monocular depth estimation methods [3, 38–40, 58] have shown promising generalization across diverse scenes, which opens new opportunities for addressing open-set monocular 3DOD. Specifically, we design a *3D bounding box head* to predict the differentiable lifting parameters from 2D object queries and enable the lift of the detected 2D bounding boxes as 3D object detection. This allows us to jointly train the open-set 2D and 3D detectors in an end-to-end (e2e) way, using both 2D and 3D ground truth (GT). Furthermore, we propose the geometry-aware 3D query generation module, which conditions 2D object queries with the camera intrinsics and depth estimation and generates 3D object queries. These 3D queries encode essential geometric information and are used for the 3D bounding box head to improve the model’s accuracy and generalization ability in 3D object detection. Additionally, we design a more effective *canonical image space*, which proves crucial for handling datasets with varying image resolutions, as demonstrated in our experiments.

Formally, we introduce the first e2e **3D Monocular Open-set Object Detecter (3D-MOOD)** by integrating the proposed 3D bounding box head, geometry-aware 3D query generation module, and canonical image space into the open-set 2D detector [27]. Our method takes a monocular input image with the language prompts and outputs the 3D object detection for the desired objects in any given scene. Experimental results demonstrate that 3D-MOOD achieves state-of-the-art (SOTA) performance on the challenging closed-set Omni3D benchmark, surpassing all previous task-specific and unified models. More importantly, in open-set settings, *i.e.* transferring from Omni3D to Ar-goverse 2 [54] and ScanNet [8], our method consistently outperforms prior models, achieving clear improvements in generalization and novel classes recognition.

Our main contributions are: (1) We explore monocular 3D object detection in open-set settings, establishing benchmarks that account for both novel scenes and unseen object categories; (2) We introduce 3D-MOOD, the first end-to-end open-set monocular 3D object detector, via 2D to 3D lifting, geometry-aware 3D query generation, and canonical image space; (3) We achieve state-of-the-art performance in both closed-set and open-set settings, demonstrating the effectiveness of our method and the feasibility of open-set monocular 3D object detection.

## 2. Related Work

### 2.1. Open-set 2D Object Detection

In recent years, there has been tremendous progress in 2D object detection [7, 13, 27, 35, 60, 61, 64] by lever-

aging language models [9] or visual-language foundation models [41] to detect and classify objects from language queries. Among varying definitions of these works, *i.e.* *open-set object detection*, *open-world object detection*, and *open-vocabulary detection*, we do not distinguish them in this section and describe the formal problem definition in Sec. 3.1.

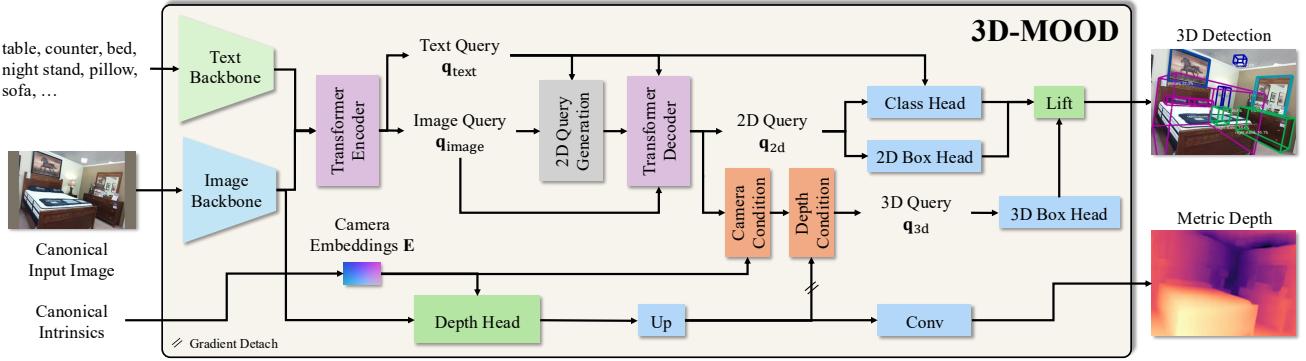
OVR-CNN [61] aligns ResNet [15] and BERT [9] features to detect novel classes, while OV-DETR [60] uses CLIP [41] image and text embeddings with Detection Transformer [6]. GLIP [27] presents grounded language-image pre-training to align object detection and captions. Detic [66] leverages image level labels to align object reasoning and text to enable tens of thousands of concepts for classification.

In contrast, G-DINO [27] deeply fuses image and text features at all stages of the detection model [62] and proposes language-guided query generation to allow the open-set 2D detector to detect the desired object classes according to input language prompts. This is more natural and intuitive for humans and can help robots understand scenes in many applications. However, in 3D monocular object detection, the lack of data annotations in 3D due to the cost also increases the difficulty of tackling the open-set classification with visual-language alignment. Thus, in this work, we aim to propose a framework that can universally lift the open-set 2D detection to 3D to address the limitation of the data annotation for open-set classification.

### 2.2. 3D Monocular Object Detection

3D monocular object detection is crucial for autonomous driving and indoor robotic navigation. In the past years, a large number of works [12, 17, 22, 22, 25, 28, 32, 36, 49, 55] proposed various methods to address the 3D multi-object detection for specific scenes, *i.e.* one model for one dataset. Recently, a challenging dataset called Omni3D [4] was proposed, providing a new direction for 3D monocular object detection. This dataset contains six popular datasets including outdoor scenes (KITTI [14] and nuScenes [5]), and indoor scenes (ARKitScenes [2], Hypersim [43], and SUN-RGB-D [47]), and object centric dataset (Objectron [1]). Cube R-CNN [4] proposes virtual depth to address the various focal lengths across the diverse training datasets. Uni-MODE [23] proposes the domain confidence to jointly train the Bird-eye-View (BEV) detector on indoor and outdoor datasets.

Although these methods work well on Omni3D, they are still limited by the closed-set classification design, hence they lack the ability to detect novel categories. To address this, OVM3D-Det [18] proposes a pipeline to generate pseudo GT for novel classes by using 2D foundation models [19, 27, 38] with Large Language Model (LLM) priors. However, while evaluating the quality of pseudo GT on



**Figure 2. 3D-MOOD.** We propose an end-to-end 3D monocular open-set object detector that takes a monocular image and the language prompts of the interested objects as input and classifies and localizes the 3D objects in the scenes. Our design will transform the input image and camera intrinsics into the proposed canonical image space and achieve the open-set ability for diverse scenes.

open-set benchmarks, the performance is limited because the pipeline can not be e2e trained with 3D data. On the contrary, our method is designed to estimate the differentiable lifting parameters of the open-set 2D detection with geometry prior. Thus, it can be supervised in the e2e manner while also no longer constrained by the closed-set classification. Furthermore, to address open-set regression in 3D, we use the canonical image space to better train 3D detectors across datasets. With our proposed components, 3D-MOOD outperforms these prior works on both closed-set and open-set benchmarks.

### 3. Method

We aim to propose the first e2e open-set monocular 3D object detector that can be generalized to different scenes and object classes. We first discuss the problem setup in Sec. 3.1 to define the goal of monocular open-set 3D object detection. Then, we introduce the overall pipeline of our proposed open-set monocular 3D object detector, 3D-MOOD, in Sec. 3.2. We illustrate our 3D bounding box head design in Sec. 3.3 and introduce the proposed canonical image space for training monocular 3DOD models across datasets in Sec. 3.4. In Sec. 3.5, we introduce the metric monocular auxiliary depth head, which enhances 3D-MOOD by providing a more comprehensive understanding of the global scene. Finally, in Sec. 3.6, we illustrate the proposed geometry-aware 3D query generation, designed to improve generalization in both closed-set and open-set settings.

#### 3.1. Problem Setup

The goal of 3D monocular open-set object detection is to detect any objects in any image, giving a language prompt for the objects of interest. To achieve this, one needs to extend the concept of open-set beyond the distinction of seen (base) and unseen (novel) classes within the same dataset [61]: We follow the manner of G-DINO [27] that

trains the model on other datasets but tests on COCO, which contains base and novel classes in unseen domains. In this work, we aim to extend this research direction to 3DOD. Thus, our main focus is on how to train the open-set detectors using the largest and most diverse pre-training data to date, *i.e.* Omni3D, and achieve good performance on unseen datasets, *e.g.* Argoverse 2 and ScanNet.

#### 3.2. Overall Architecture

As shown in Fig. 2, we address the monocular open-set 3DOD by lifting the open-set 2D detection. Formally, we estimate 2D bounding boxes  $\hat{\mathbf{D}}_{2D}$  from an input image  $\mathbf{I}$  and language prompts  $\mathbf{T}$ , and lift them as 3D orientated bounding boxes  $\hat{\mathbf{D}}_{3D}$  in the corresponding camera coordinate frame with the object classes  $\hat{\mathcal{C}}$ . A 2D box is defined as  $\hat{\mathbf{b}}_{2D} = [\hat{x}_1, \hat{y}_1, \hat{x}_2, \hat{y}_2]$ , where  $\hat{\mathbf{b}}_{2D} \in \hat{\mathbf{D}}_{2D}$  in the pixel coordinate. A 3D bounding box is defined as  $\hat{\mathbf{b}}_{3D} = [\hat{x}, \hat{y}, \hat{z}, \hat{w}, \hat{l}, \hat{h}, \hat{R}]$ , where  $\hat{\mathbf{b}}_{3D} \in \hat{\mathbf{D}}_{3D}$ .  $[\hat{x}, \hat{y}, \hat{z}]$  stands for the 3D location in the camera coordinates,  $[\hat{w}, \hat{l}, \hat{h}]$  stands for the object's dimensions as width, length, and height, and  $\hat{R}$  is the rotation matrix  $\hat{R} \in SO(3)$  of the object.

We choose G-DINO [27] as our 2D open-set object detector for its early visual-language features fusion design. On top of it, we build 3D-MOOD with the proposed *3D bounding box head*, *canonical image space*, and *geometry-aware 3D query generation module* for end-to-end open-set 3D object detection. We use an image encoder [30] to extract image features  $q_{image}$  from  $\mathbf{I}$  and use a text backbone [9] from  $\mathbf{T}$  to extract text features  $q_{text}$ . Then, following detection transformer architectures [6, 62, 67], we pass  $q_{image}$  and  $q_{text}$  to the transformer [50] encoder with early visual-language features fusion [21]. The image and text features will be used in the proposed language-guided query selection to generate encoder detection results  $\hat{\mathbf{D}}_{2D}^{enc}$  and bounding box queries  $q_{2d}^0$  for the decoder. For each cross-modality transformer decoder layer  $TrD_i$ , it uses

a text cross-attention  $\text{CA}_{\text{text}}^i$  and an image cross-attention  $\text{CA}_{\text{image}}^i$  to combine  $\mathbf{q}_{2d}^i$  with the multi-modality information as:

$$\begin{aligned}\mathbf{q}_{2d}^i &= \text{CA}_{\text{text}}^i(\text{SA}^i(\mathbf{q}_{2d}^i), \mathbf{q}_{\text{text}}), \\ \mathbf{q}_{2d}^{i+1} &= \text{FFN}^i(\text{CA}_{\text{image}}^i(\mathbf{q}_{2d}^i, \mathbf{q}_{\text{image}})),\end{aligned}\quad (1)$$

where  $i$  starts from 0 to  $l - 1$  and FFN stands for feed-forward neural network. Each layer bounding box queries  $\mathbf{q}_{2d}^i$  will be decoded as 2D bounding boxes prediction  $\hat{\mathbf{D}}_{2D}^i$  by the 2D box head as  $\hat{\mathbf{D}}_{2D}^i = \text{MLP}_{2D}^i(\mathbf{q}_{2d}^i)$ , where MLP stands for Multi-Layer Perceptron. The object classes  $\hat{\mathbf{C}}$  are estimated based on the similarity between  $\mathbf{q}_{2d}^i$  and the input text embeddings.

### 3.3. 3D Bounding Box Head

Given the estimated 2D bounding boxes  $\hat{\mathbf{D}}_{2D}$  and the corresponding object queries, our 3D bounding box head predict the 3D properties of  $\hat{\mathbf{D}}_{2D}$  to lift it and get  $\hat{\mathbf{D}}_{3D}$  in the camera coordinate frame.

**3D Localization.** To localize the 3D center of the 3D bounding boxes in the camera coordinates, 3D-MOOD predicts the projected 3D center and the metric depth of the 3D center of the object as [4, 12, 17]. To be more specific, we predict  $[\hat{u}, \hat{v}]$  as the distance between the projected 3D center and the center of the 2D detections. We lift the projected center to the camera coordinate with the given camera intrinsic  $\mathbf{K}$  and the estimated metric depth  $\hat{z}$  of the 3D bounding boxes center. We estimate the scaled logarithmic depth prediction from our 3D bounding box head noted as  $\hat{d}$  with depth scale  $s_{depth}$ . Thus, the metric depth will be acquired as  $\hat{z} = \exp(\hat{d}/s_{depth})$  during inference.

**3D Object Dimensions.** To estimate the universal 3D objects, we follow [12, 17] to directly predict dimensions instead of using the pre-computed category prior as in [4]. Our bounding box head predicts the scaled logarithmic dimensions as  $[s_{dim} \times \ln \hat{w}, \ln \hat{l} \times s_{dim}, \ln \hat{h} \times s_{dim}]$  as the output space and can obtain the width, length, and height with exponential and divided by scale  $s_{dim}$  during inference.

**3D Object Orientation.** Unlike [12, 17], we follow [20] to predict 6D parameterization of  $\hat{R}$ , denoted as  $\hat{rot}_{6d}$ , instead of only estimating yaw as autonomous driving scenes.

Following detection transformer (DETR) [6]-like architecture design, we use an MLP as the 3D bounding box head to estimate the 12 dimension 3D properties from 2D object queries  $\mathbf{q}_{2d}^i$  for each transformer decoder layer  $i$ . The 3D detection  $\hat{\mathbf{D}}_{3D}^i$  for each layer is estimated by separate 3D bounding box heads ( $\text{MLP}_{3D}^i$ ) as:

$$\hat{\mathbf{D}}_{3D}^i = \text{Lift}(\text{MLP}_{3D}^i(\mathbf{q}_{2d}^i), \hat{\mathbf{D}}_{2D}^i, \mathbf{K}). \quad (2)$$

where **Lift** stands for we obtain the final 3D detections in the camera coordinate by lifting the projected 3D center with the estimated dimensions and rotation.

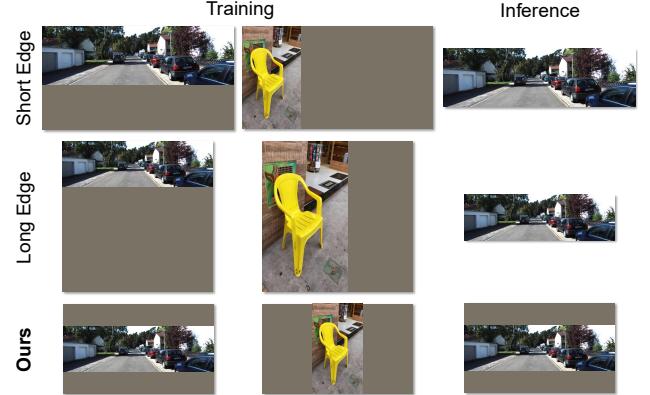


Figure 3. **Canonical Image Space.** We compare the difference between different resizing and padding strategies. It is worth noting that the same image will have the same camera intrinsic  $\mathbf{K}$  despite having very different image resolutions for previous methods.

### 3.4. Canonical Image Space

To train the model across datasets that contain images with different resolutions from various datasets, previous works [4, 23, 27] either resize the short or long edge to a particular value, then use right and bottom padding to align the image resolution of the training batches. However, as shown in Fig. 3, previous methods will heavily pad zeros when the training batches have very different resolutions and won't change the camera intrinsics. This wastes resources for non-informative information but will also cause the same camera intrinsic  $\mathbf{K}$  but with different image resolutions between training and inference time while also breaking the center projection assumption.

As illustrated in [58], the ambiguity among image, camera intrinsic, and metric depth will confuse the depth estimation model during training with multiple datasets. Thus, we proposed the canonical space where the model can have a unified observation for both training and testing time. We use the fixed input image resolution  $[\mathbf{H}_c \times \mathbf{W}_c]$  and resize the input images and intrinsics so that the height or width reaches  $\mathbf{H}_c$  or  $\mathbf{W}_c$  to keep the original input image ratio. Then, we center pad the images to  $[\mathbf{H}_c \times \mathbf{H}_W]$  with value 0 and pad the camera intrinsic accordingly. This alignment is necessary for the model to learn the universal settings consistent across training and test time, and we demonstrate the effectiveness in closed-set and open-set experiments. We show more details in the supplementary material.

### 3.5. Auxiliary Metric Depth Estimation

A significant challenge in monocular 3DOD is accurately estimating object localization in 3D. 3D object localization is directly tied to the localization in the image plane and the object metric depth, making the metric depth estimation sub-task crucial for 3DOD. Previous methods [26, 36, 55]

have emphasized the importance of incorporating an auxiliary depth estimation head to improve 3D localization. However, achieving accurate depth localization becomes more difficult when attempting to generalize depth estimation across different datasets. Recent methods [38, 39, 58] demonstrate the possibility of overcoming the universal depth estimation by leveraging the camera information. As Cube R-CNN [4] uses a similar approach as Metric3D [58] to have virtual depth, we argue that conditioning depth features on camera intrinsics yields a more robust solution. This approach avoids being limited by variations in camera models and enhances generalizability. To this end, we design an auxiliary depth estimation head conditioned on the camera information, as proposed in UniDepth [38, 40] to achieve a generalizable monocular depth estimation.

In particular, our model architecture incorporates an additional Feature Pyramid Network (FPN) [24] to extract depth features  $\mathbf{F}$  from the image backbone [30]. We rescale them to 1/16 of the input image height  $H$  and width  $W$  and generate the depth features  $\mathbf{F}_{16}^d$  using a Transformer block [50]. We condition  $\mathbf{F}_{16}^d$  using camera embeddings,  $\mathbf{E}$ , as described in [38]. We then upsample the depth features to 1/8 of the input image height and width, *i.e.*  $\mathbf{F}_8^d|\mathbf{E}$  to estimate the metric depth by a convolutional block. We generate the scaled logarithmic depth prediction  $\hat{d}_{\text{full}}$  with the same depth scale  $s_{\text{depth}}$  as our 3D bounding box head. Thus, the final metric depth  $\hat{z}_{\text{full}}$  will be acquired as  $\hat{z}_{\text{full}} = \exp(\hat{d}_{\text{full}}/s_{\text{depth}})$ .

### 3.6. Geometry-aware 3D Query Generation

To ensure the 3D bounding box estimation can be generalized for diverse scenes, we propose a geometry-aware 3D query generation to condition the 2D object query  $\mathbf{q}_{2d}$  with the learned geometry prior. First, we use the camera embeddings  $\mathbf{E}$  in our auxiliary depth head to make the model aware of the scene-specific prior via a cross-attention layer. Due to the sparsity of the 3D bounding box annotations compared to the per-pixel depth supervision, we further leverage the depth features  $\mathbf{F}_8^d|\mathbf{E}$  to condition the object query. This allows us to align the metric depth prediction and 3D bounding box estimation while leveraging the learned depth estimation. Our full geometry-aware query generation will generate the 3D box queries  $\mathbf{q}_{3d}$  as:

$$\begin{aligned}\mathbf{q}_{3d}^i &= \text{FFN}_{\text{cam}}^i(\text{CA}_{\text{cam}}^i(\text{SA}_{\text{cam}}^i(\mathbf{q}_{2d}^i), \mathbf{E})), \\ \mathbf{q}_{3d}^i &= \text{FFN}_{\text{depth}}^i(\text{CA}_{\text{depth}}^i(\text{SA}_{\text{depth}}^i(\mathbf{q}_{3d}^i), \mathbf{F}_8^d|\mathbf{E})).\end{aligned}\quad (3)$$

We replace the 2D object queries in Eq. (2) with the generated 3D queries  $\mathbf{q}_{3d}^i$  for each decoder layer as

$$\hat{\mathbf{D}}_{3D}^i = \text{Lift}(\text{MLP}_{3D}^i(\mathbf{q}_{3d}^i), \hat{\mathbf{D}}_{2D}^i, \mathbf{K}). \quad (4)$$

It is worth noting that we detach the gradient from the cross attention between 3D query and depth features to stabilize

the training. We validate our geometry-aware 3D query generation in our ablation studies for both closed-set and open-set settings. The results suggest that incorporating geometric priors enhances model convergence during closed-set multi-dataset training and improves the robustness of 3D bounding box estimation in real-world scenarios.

### 3.7. Training Loss

We train 3D-MOOD with 2D losses  $L_{2D}$ , 3D losses  $L_{3D}$ , and auxiliary depth loss  $L_{\text{depth}}^{\text{aux}}$  in conjugation. For 2D losses, we follow MM G-DINO [64] and use L1 loss and GIoU [42] loss for the 2D bounding box regression and contrastive between predicted objects and language tokens for bounding box classification as GLIP [21]. For the 3D losses, we use L1 loss to supervise each estimated 3D properties. We compute 2D and 3D losses for each transformer decoder layer  $i$  and obtain  $L_{2D}^i$  and  $L_{3D}^i$ . For auxiliary depth estimation, we refer to each original dataset of Omni3D to find the depth GT or using the projected LiDAR points or structure-from-motion (SfM) [45, 46] points. We use Scale-invariant log loss [11] as auxiliary depth loss  $L_{\text{depth}}^{\text{aux}}$  with  $\lambda_{\text{depth}}$  as loss weight for supervision. Finally, we set the loss weights for 2D and 3D detection to 1.0 and  $\lambda_{\text{depth}}$  to 10 and obtain the final loss  $L_{\text{final}}$  as

$$L_{\text{final}} = \sum_{i=0}^l (L_{2D}^i + L_{3D}^i) + \lambda_{\text{depth}} L_{\text{depth}}^{\text{aux}}. \quad (5)$$

## 4. Experiments

We first describe our implementation details for 3D-MOOD and datasets in Sec. 4.1 and discuss the evaluation metrics in Sec. 4.2. Then, we show the open-set, cross-domain, and closed-set results in Sec. 4.3, Sec. 4.4, and Sec. 4.5, and analyze the results of ablation studies in Sec. 4.6. We show some qualitative results in Sec. 4.7 and more in the supplementary material.

### 4.1. Implementation Details

**Model.** We use the Vis4D [56] as the framework to implement 3D-MOOD in PyTorch [37] and CUDA [33]. We train the full model for 120 epochs with batch size of 128 and set the initial learning rate of 0.0004 following [64]. For the ablation studies, we train the model for 12 epochs with batch size of 64. We choose  $800 \times 1333$  as our canonical image shape, as described in Sec. 3.4. During training, we use random resize with scales between [0.75, 1.25] and random horizontal flipping with a probability of 0.5 as data augmentation. We decay the learning rate by a factor of 10 at epochs 8 and 11 for the 12 epoch setting and by 80 and 110 for the 120 epoch setting.

**Closed-set Data.** We use Omni3D [4] as training data, which contains six popular monocular 3D object detection datasets, *i.e.* KITTI [14], nuScenes [5], SUN RGB-D [47],

Objectron [1], ARKitScenes [2], Hypersim [43]. There are 176573 training images, 19127 validation images, and 39452 testing images with 98 classes. We follow [4, 23] using the training and validation split from Omni3D [4] with 50 classes for training and test the model on the test split.

**Open-set Data.** We choose two challenging datasets for indoor and outdoor scenes as the open-set monocular 3D object detection benchmarks. For outdoor settings, we use the validation split of Argoverse 2 (AV2) [54] Sensor Dataset as the benchmark. We sample 4806 images from the ring front-center camera, which provides portrait resolution ( $2048 \times 1550$ ), and use all the official classes that appear in the validation set to be evaluated. For indoor settings, we use the validation split of ScanNet [8] with official 18 classes as the indoor benchmark. We uniformly sample 6240 images with  $968 \times 1296$  resolution along with the axis-aligned 3D bounding boxes. We provide more details in the supplementary material.

## 4.2. Evaluation

We use the average precision (AP) metric to evaluate the performance of 2D and 3D detection results. Omni3D [4] matches the predictions and GT by computing the intersection-over-union (IoU<sub>3D</sub>) of 3D cuboids. The mean 3D AP, *i.e.* AP<sub>3D</sub>, is reported across classes and over a range of IoU<sub>3D</sub> thresholds  $\in [0.05, 0.1, \dots, 0.5]$ . However, this matching criterion is too restrictive for small or thin objects for monocular object detection, especially for open-set scenarios. As shown in Fig. 4, we report the difference between different matching criteria over three classes and methods under open-set settings. The performance of large objects, such as Regular Vehicles (cars), remains consistent between center-distance (CD) based and IoU-based matching. However, for smaller objects (*e.g.*, Sinks) and thinner objects (*e.g.*, Pictures), IoU-based matching fails to accurately reflect the true performance of 3D monocular object detection. Thus, we refer to nuScenes detection score (NDS) [5] and composite detection score (CDS) [54] to propose a new 3D object detection score for open-set monocular object detection noted as open detection score (**ODS**).

To use ODS for both indoor and outdoor datasets, we use 3D Euclidean distance instead of the bird-eye view (BEV) distance used in autonomous driving scenes. Furthermore, unlike NDS and CDS using the fixed distances as matching thresholds, we set the matching distances as the uniform range  $\in [0.5, 0.55, \dots, 1.0]$  of the *radius* of the 3D GT boxes. This allows a flexible matching criterion given the object size and strikes a balance between IoU matching and other distance matching. We report mean 3D AP using normalized distance-based matching as AP<sub>3D</sub><sup>dist</sup> over classes. We compute several true positive (TP) errors for the matched prediction and GT pair. We report mean average translation error (mATE), mean average scale error (mASE), and mean

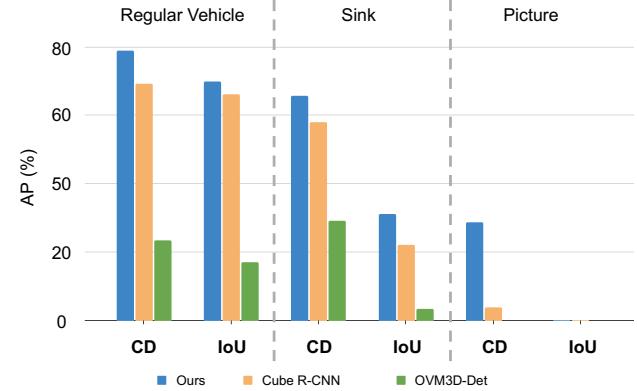


Figure 4. **Matching function.** Different matching criteria over three methods on three different classes on AV2 and ScanNet. CD stands for matching prediction and GT using our proposed normalized center distance matching, while IoU stands for using IoU<sub>3D</sub>.

average orientation error (mAOE) to evaluate how precise the true positive is compared to the matched GT. The final ODS is computed as the weighted sum of AP<sub>3D</sub><sup>dist</sup>, mATE, mASE, and mAOE as:

$$\text{ODS} = \frac{1}{6}[3 \times \text{AP}_{3D}^{\text{dist}} + \sum(1 - \text{mTPE})], \quad (6)$$

where mTPE  $\in [\text{mATE}, \text{mASE}, \text{mAOE}]$ . ODS considers the average precision and true positive errors under the flexible distance matching, making it suitable for evaluating the monocular 3D detection results, especially for open-set settings. In this work, we report AP<sub>3D</sub>, AP<sub>3D</sub><sup>dist</sup>, and ODS in the form of percentage by default. Additional details are provided in the supplementary material.

## 4.3. Open-Set Results

**Benchmarks.** We establish the first 3D monocular open-set object detection benchmark as Tab. 1. We treat the diverse Omni3D [4] dataset as the training set and test the model performance on Argoverse 2 (outdoor) [54] and ScanNet (indoor) [8] validation splits as open-set testing.

**Baselines.** We validate the performance of 3D-MOOD by comparing it to several baselines. To the best of our knowledge, there are only two methods [4, 23] trained on the entire Omni3D training set. However, until the submission, Uni-MODE [23] did not release their model weights. Hence, we use Cube R-CNN [4] to build several baselines. We further compare the generalizability of 3D-MOOD with Uni-MODE in Sec. 4.4. We use three different Cube R-CNN models, which are trained with indoor-only, outdoor-only, or full Omni3D training sets, as the specialized closed-set models for indoor (In), outdoor (Out), and universal data (Cube R-CNN). We map the predicted categories from Omni3D to Argoverse 2 (AV2) and ScanNet to conduct the 3D detection on open data, which can provide 11 and

**Table 1. Open-set Results.** We propose the first 3D monocular open-set object detection benchmark with Argoverse 2 [54] (Outdoor) and ScanNet [8] (Indoor). Each dataset contains seen (base) and unseen (novel) categories in the unseen scenes. Besides Cube R-CNN [4] full model, we evaluate Cube R-CNN (In/Out) as each domain expert variant, which is only trained and tested on Omni3D indoor/outdoor datasets. It is worth noting that OVM3D-Det’s depth estimation model [38] is trained on AV2 and ScanNet. We further evaluate the generalization of seen classes and the ability to detect novel classes through **ODS (B)** and **ODS (N)**. 3D-MOOD establishes the SOTA performance on this new challenging open-set benchmark.

Method	Argoverse 2							ScanNet						
	AP <sub>3D</sub> <sup>dist</sup> ↑	mATE ↓	mASE ↓	mAOE ↓	ODS ↑	ODS (B) ↑	ODS (N) ↑	AP <sub>3D</sub> <sup>dist</sup> ↑	mATE ↓	mASE ↓	mAOE ↓	ODS ↑	ODS (B) ↑	ODS (N) ↑
Cube R-CNN (In)	-	-	-	-	-	-	-	19.5	0.725	0.771	0.858	20.5	24.6	0.0
Cube R-CNN (Out)	10.5	0.896	0.869	0.991	9.3	19.5	0.0	-	-	-	-	-	-	-
Cube R-CNN [4]	8.6	0.903	0.867	0.953	8.9	18.6	0.0	20.0	0.733	0.774	0.921	19.5	23.4	0.0
OVM3D-Det <sup>†</sup> [18]	7.7	0.914	0.893	0.899	8.8	16.5	1.7	15.6	0.798	0.871	0.818	16.3	17.8	8.8
<b>Ours (Swin-T)</b>	<b>14.8</b>	0.782	0.697	0.612	22.5	31.7	14.2	27.3	0.630	0.726	<b>0.650</b>	30.2	33.6	13.4
<b>Ours (Swin-B)</b>	14.7	<b>0.755</b>	<b>0.680</b>	<b>0.580</b>	<b>23.8</b>	<b>33.6</b>	<b>14.8</b>	<b>28.8</b>	<b>0.612</b>	<b>0.706</b>	0.655	<b>31.5</b>	<b>34.7</b>	<b>15.7</b>

**Table 2. Cross-domain results.** We validate 3D-MOOD cross-domain generalization by training on one of the indoor datasets from Omni3D while testing on the other two in a zero-shot manner. 3D-MOOD generalize better consistently for all three settings.

Method	Trained on Hypersim			Trained on SUN RGB-D			Trained on ARKitScenes		
	AP <sub>3D</sub> <sup>hyp</sup> ↑	AP <sub>3D</sub> <sup>sun</sup> ↑	AP <sub>3D</sub> <sup>ark</sup> ↑	AP <sub>3D</sub> <sup>hyp</sup> ↑	AP <sub>3D</sub> <sup>sun</sup> ↑	AP <sub>3D</sub> <sup>ark</sup> ↑	AP <sub>3D</sub> <sup>hyp</sup> ↑	AP <sub>3D</sub> <sup>sun</sup> ↑	AP <sub>3D</sub> <sup>ark</sup> ↑
Cube R-CNN [4]	15.2	9.5	7.5	9.5	34.7	14.2	7.5	13.1	38.6
Uni-MODE [23]	14.7	5.6	3.6	3.0	28.5	8.8	4.2	13.0	35.0
<b>Ours</b>	<b>25.6</b>	<b>15.9</b>	<b>14.5</b>	<b>13.8</b>	<b>42.1</b>	<b>21.4</b>	<b>12.9</b>	<b>23.8</b>	<b>43.9</b>

15 seen (base) classes, respectively. Another baseline is OVM3D-Det [18], which uses G-DINO [27], SAM [19], UniDepth [38] and LLM to generating pseudo GT for 3D detection. We run the OVM3D-Det pipeline on AV2 and ScanNet to generate the pseudo GT as open-set detection results and evaluate it with the real GT.

**Results.** As shown in Tab. 1, 3D-MOOD achieves the SOTA on both challenging datasets in open-set settings. The Cube R-CNN baselines (rows 1 to 3) show that the closed-set methods lack the ability to recognize the novel objects due to the closed-vocabulary model design, which further heavily affects the overall open-set performance when more than half of classes are novel, *e.g.* AV2. Furthermore, the performance differences between 3D-MOOD and Cube R-CNN on the seen (base) classes are more significant in the unseen domain. This suggests that 3D-MOOD benefits from the proposed canonical image spaces and geometry-aware 3D query generation, leading us to generalize better for unseen domains. The comparison to OVM3D-Det [18] shows the importance of e2e design to align better 2D open-set detector and 3D object detection. Given that UniDepth [38] is trained on AV2 and ScanNet, the depth estimation from OVM3D-Det is much more accurate. However, a lack of training in 3D data leads to worse performance for both base and novel classes.

#### 4.4. Cross Domain Results.

Since we can not directly compare Uni-MODE [23] on our proposed open-set benchmarks, we follow [4, 23] and conduct the cross-domain generalization experiments within Omni3D datasets. We train 3D-MOOD on one indoor

**Table 3. Results on Omni3D.** We compare 3D-MOOD with other closed-set detectors on Omni3D test set. AP<sub>3D</sub><sup>omni</sup> ↑ is the average scores over Omni3D 6 datasets. All methods are trained with Omni3D train and val splits and “-” represents the results not reported in previous literature [4, 23]. 3D-MOOD achieves SOTA performance on the closed-set setting with the open-set ability.

Method	AP <sub>3D</sub> <sup>kit</sup> ↑	AP <sub>3D</sub> <sup>miss</sup> ↑	AP <sub>3D</sub> <sup>sun</sup> ↑	AP <sub>3D</sub> <sup>hyp</sup> ↑	AP <sub>3D</sub> <sup>ark</sup> ↑	AP <sub>3D</sub> <sup>obj</sup> ↑	AP <sub>3D</sub> <sup>omni</sup> ↑
ImVoxelNet [44]	-	-	-	-	-	-	9.4
SMOKE [29]	-	-	-	-	-	-	9.6
FCOS3D [51]	-	-	-	-	-	-	9.8
PGD [52]	-	-	-	-	-	-	11.2
Cube R-CNN [4]	32.6	30.1	15.3	7.5	41.7	50.8	23.3
Uni-MODE* [23]	29.2	<b>36.0</b>	23.0	8.1	48.0	66.1	28.2
<b>Ours (Swin-T)</b>	<b>32.8</b>	31.5	21.9	<b>10.5</b>	51.0	64.3	28.4
<b>Ours (Swin-B)</b>	31.4	35.8	<b>23.8</b>	9.1	53.9	<b>67.9</b>	<b>30.0</b>

dataset at once and zero-shot test on the other two datasets. As shown in Tab. 2, our method can achieve higher performance for both in-domain data, *i.e.* seen dataset, and out-of-domain data. We believe it demonstrates the ability of the models to detect the base object in the unseen scenes, which benefits from our geometry-aware design.

#### 4.5. Closed-Set Results

We compare 3D-MOOD with the other closed-set models on the Omni3D [4] benchmark. As shown in Tab. 3, 3D-MOOD achieves the SOTA performance on Omni3D test split. Our model with Swin-Transformer [30] Tiny (Swin-T) as backbone achieves similar performance as previous SOTA Uni-MODE [23], which uses ConvNeXt [31] Base model. When we use the comparable image backbone to ConvNeXt-Base (89M), *i.e.* Swin-Transformer Base (Swin-B, 88M), 3D-MOOD achieves 30.1% AP on Omni3D test set and establish the new SOTA results on the benchmark.

Table 4. **Ablations of 3D-MOOD.** **CI** denotes canonical image space, **Depth** denotes auxiliary depth estimation head, and **GA** stands for geometry-aware 3D query generation. We report the IoU-based AP for the Omni3D test split and our ODS for the AV2 and ScanNet validation split. **AP<sub>3D</sub><sup>omni</sup>** is the average scores over Omni3D 6 datasets while **ODS<sup>open</sup>** is the average for open-set datasets. The results show that our proposed component help for both closed-set and open-set settings.

CI	Depth	GA	AP <sub>3D</sub> <sup>kit</sup> ↑	AP <sub>3D</sub> <sup>nus</sup> ↑	AP <sub>3D</sub> <sup>hyp</sup> ↑	AP <sub>3D</sub> <sup>sun</sup> ↑	AP <sub>3D</sub> <sup>ark</sup> ↑	AP <sub>3D</sub> <sup>obj</sup> ↑	AP <sub>3D</sub> <sup>omni</sup> ↑	ODS <sup>av2</sup> ↑	ODS <sup>scan</sup> ↑	ODS <sup>open</sup> ↑	
1	-	-	<b>32.5</b>	29.7	8.1	17.3	46.5	54.9	24.1	18.2	29.0	23.6	
2	✓	-	-	31.1	30.5	9.1	19.1	47.7	58.1	25.5	19.5	29.5	
3	✓	✓	-	29.8	30.7	<b>10.3</b>	19.9	48.6	58.8	26.2	20.0	29.4	
4	✓	✓	✓	<u>32.1</u>	<b>31.9</b>	<u>9.9</u>	<b>20.8</b>	<b>49.1</b>	<b>60.2</b>	<b>26.8</b>	<b>22.0</b>	<b>30.0</b>	<b>26.0</b>

#### 4.6. Ablation Studies

We ablate each contribution in Tab. 4 for both closed-set and open-set settings. We build the naive baseline as row 1 by directly using 2D queries  $q_{2d}$  and directly generate the 3D detection results.

**Canonical Image Space.** As shown in [38, 42], it is crucial to resolve the ambiguity between image, intrinsic, and depth. With the proposed canonical image (CI) space, we align the training and testing time image shape and camera intrinsics. Row 2 outlines how CI improves closed-set and open-set results by 1.4 and 0.9 for closed-set and open-set settings, respectively. This shows that the model learns the universal property and makes the detection ability generalize well across datasets for training and testing time.

**Auxiliary Depth head.** We validate the effect of the auxiliary depth head as row 3. Learning metric depth is essential for the network to better understand the geometry of the 3D scene instead of merely relying on the sparse depth supervision signal from the 3D bounding boxes loss. With the auxiliary depth head, 3D-MOOD improves 0.7 AP on the closed-set settings yet only slightly improve the open-set settings by 0.2 ODS. We hypothesize that the depth data is not diverse and rich enough in Omni3D compared to the data that other generalizable depth estimation methods [3, 38, 58] use for training. Thus, the benefits from the depth head is little in open-set settings.

**Geometry-aware 3D Query Generation.** Finally, we ablate our proposed geometry-aware 3D query generation module in row 4. We show that for both closed-set and open-set settings, the geometry condition can improve the performance by 0.6 and 1.3, respectively. It is worth noting that the geometry information can significantly improve the model’s generalizability, which demonstrates our contribution to 3D monocular open-set object detection.

#### 4.7. Qualitative Results

We show the open-set qualitative results in Fig. 5 to demonstrate the generalizability of 3D-MOOD, where we successfully detect novel objects in unseen scenes. More results are reported in the supplementary material.

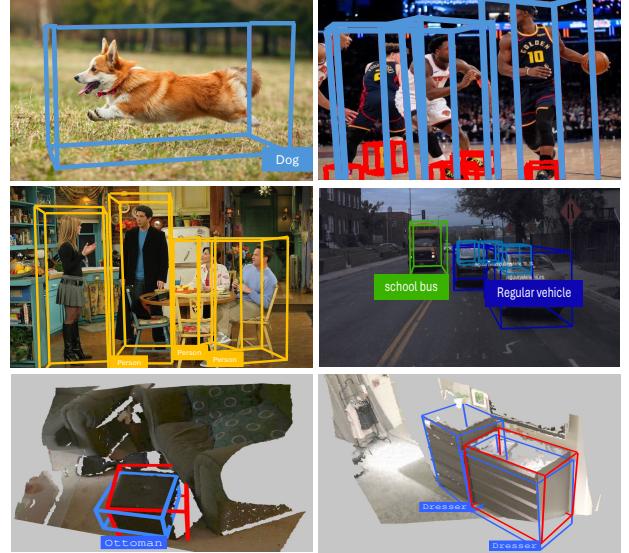


Figure 5. **In-the-wild Qualitative Results.** We show the visualization of 3D-MOOD for in-the-wild images. The red boxes in the 3D visualization (last row) are the GT annotations.

#### 5. Conclusion

In this work, we introduce 3D-MOOD, the first end-to-end 3D monocular open-set object detection method, which achieves state-of-the-art performance in closed-set settings while proving strong generalization to unseen scenes and object classes in open-set scenarios. We design a 3D bounding box head with the proposed geometry-aware 3D query generation to lift the open-set 2D detection to the corresponding 3D space. Our proposed method can be trained end-to-end and yield better overall performance. Furthermore, our proposed canonical image space resolves the ambiguity between image, intrinsic, and metric depth, leading to more robust results in closed-set and open-set settings. We propose a challenging 3D monocular open-set object detection benchmark using two out-of-domain datasets. 3D-MOOD sets the new state-of-the-art performance on the challenging Omni3D benchmark compared to other closed-set methods. Moreover, the results on the open-set benchmark demonstrate our method’s ability to generalize the monocular 3D object detection in the wild.

## Acknowledgement

This research is supported by the ETH Foundation Project 2025-FS-352, Swiss AI Initiative and a grant from the Swiss National Supercomputing Centre (CSCS) under project ID a03 on Alps, and the Lamarr Institute for Machine Learning and Artificial Intelligence. The authors thank Linfei Pan and Haofei Xu for helpful discussions and technical support.

## References

- [1] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021. [1](#), [2](#), [6](#)
- [2] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, 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. [1](#), [2](#), [6](#)
- [3] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. [2](#), [8](#)
- [4] Garrick Brazil, Abhinav Kumar, Julian Straub, Nikhila Ravi, Justin Johnson, and Georgia Gkioxari. Omni3d: A large benchmark and model for 3d object detection in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13154–13164, 2023. [1](#), [2](#), [4](#), [5](#), [6](#), [7](#)
- [5] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*, 2019. [1](#), [2](#), [5](#), [6](#)
- [6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. [2](#), [3](#)
- [7] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. *arXiv preprint arXiv:2401.17270*, 2024. [2](#)
- [8] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. [2](#), [6](#), [7](#), [3](#), [4](#)
- [9] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. [2](#), [3](#)
- [10] Xingshuai Dong, Matthew A Garratt, Sreenatha G Anavatti, and Hussein A Abbass. Towards real-time monocular depth estimation for robotics: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):16940–16961, 2022. [1](#)
- [11] David Eigen, Christian Puhrs, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in Neural Information Processing Systems (NeurIPS)*, 27, 2014. [5](#)
- [12] Tobias Fischer, Yung-Hsu Yang, Suryansh Kumar, Min Sun, and Fisher Yu. Cc-3dt: Panoramic 3d object tracking via cross-camera fusion. In *6th Annual Conference on Robot Learning*, 2022. [2](#), [4](#)
- [13] Yuqian Fu, Yu Wang, Yixuan Pan, Lian Huai, Xingyu Qiu, Zeyu Shangguan, Tong Liu, Yanwei Fu, Luc Van Gool, and Xingqun Jiang. Cross-domain few-shot object detection via enhanced open-set object detector. In *European Conference on Computer Vision*, pages 247–264. Springer, 2024. [2](#)
- [14] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. [1](#), [2](#), [5](#)
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. [2](#)
- [16] Sicheng He, Zeyu Shangguan, Kuanning Wang, Yongchong Gu, Yuqian Fu, Yanwei Fu, and Daniel Seita. Sequential multi-object grasping with one dexterous hand. *IROS*, 2025. [1](#)
- [17] Hou-Ning Hu, Yung-Hsu Yang, Tobias Fischer, Trevor Darrell, Fisher Yu, and Min Sun. Monocular quasi-dense 3d object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1992–2008, 2022. [2](#), [4](#)
- [18] Rui Huang, Henry Zheng, Yan Wang, Zhuofan Xia, Marco Pavone, and Gao Huang. Training an open-vocabulary monocular 3d detection model without 3d data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [2](#), [7](#)
- [19] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. [2](#), [7](#)
- [20] Abhijit Kundu, Yin Li, and James M Rehg. 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3559–3568, 2018. [4](#)
- [21] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng 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. [3](#), [5](#)
- [22] Zhiqi Li, Wenhui Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer:

- Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022. 1, 2
- [23] Zhuoling Li, Xiaogang Xu, SerNam Lim, and Hengshuang Zhao. Unimode: Unified monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16561–16570, 2024. 1, 2, 4, 6, 7
- [24] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 5
- [25] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. *arXiv preprint arXiv:2211.10581*, 2022. 2
- [26] Xuewu Lin, Zixiang Pei, Tianwei Lin, Lichao Huang, and Zhizhong Su. Sparse4d v3: Advancing end-to-end 3d detection and tracking. *arXiv preprint arXiv:2311.11722*, 2023. 4
- [27] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 2, 3, 4, 7, 1
- [28] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. *arXiv preprint arXiv:2203.05625*, 2022. 1, 2
- [29] Zechen Liu, Zizhang Wu, and Roland Tóth. Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 996–997, 2020. 7
- [30] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3, 5, 7, 1
- [31] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 7, 3
- [32] Andretti Naiden, Vlad Paunescu, Gyeongmo Kim, Byeong-Moon Jeon, and Marius Leordeanu. Shift r-cnn: Deep monocular 3d object detection with closed-form geometric constraints. In *2019 IEEE international conference on image processing (ICIP)*, pages 61–65. IEEE, 2019. 2
- [33] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with cuda: Is cuda the parallel programming model that application developers have been waiting for? *Queue*, 6(2):40–53, 2008. 5
- [34] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al.
- Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3
- [35] Jiancheng Pan, Yanxing Liu, Yuqian Fu, Muyuan Ma, Jiahao Li, Danda Pani Paudel, Luc Van Gool, and Xiaomeng Huang. Locate anything on earth: Advancing open-vocabulary object detection for remote sensing community. *arXiv preprint arXiv:2408.09110*, 2024. 2
- [36] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 4
- [37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 8024–8035. Curran Associates, Inc., 2019. 5
- [38] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. UniDepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 5, 7, 8
- [39] Luigi Piccinelli, Christos Sakaridis, Mattia Segu, Yung-Hsu Yang, Siyuan Li, Wim Abbeloos, and Luc Van Gool. UniK3D: Universal camera monocular 3d estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 5
- [40] Luigi Piccinelli, Christos Sakaridis, Yung-Hsu Yang, Mattia Segu, Siyuan Li, Wim Abbeloos, and Luc Van Gool. UniDepthV2: Universal monocular metric depth estimation made simpler. *arXiv:2502.20110*, 2025. 2, 5
- [41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2
- [42] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019. 5, 8
- [43] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *International Conference on Computer Vision (ICCV) 2021*, 2021. 1, 2, 6
- [44] Danila Rukhovich, Anna Vorontsova, and Anton Konushin. Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2397–2406, 2022. 1, 7

- [45] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5
- [46] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 5
- [47] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015. 1, 2, 5
- [48] Gemini Team, Rohan Anil, Sébastien Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 3
- [49] Tao Tu, Shun-Po Chuang, Yu-Lun Liu, Cheng Sun, Ke Zhang, Donna Roy, Cheng-Hao Kuo, and Min Sun. Imageonet: Image-induced geometry-aware voxel representation for multi-view 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6996–7007, 2023. 1, 2
- [50] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 3, 5
- [51] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 913–922, 2021. 1, 7
- [52] Tai Wang, ZHU Xinge, Jiangmiao Pang, and Dahua Lin. Probabilistic and geometric depth: Detecting objects in perspective. In *Conference on Robot Learning*, pages 1475–1485. PMLR, 2022. 7
- [53] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudolidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8445–8453, 2019. 1
- [54] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemeyer Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021. 2, 6, 7, 1, 3, 4
- [55] Chenyu Yang, Yuntao Chen, Haofei Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Y. Qiao, Lewei Lu, Jie Zhou, and Jifeng Dai. Bevformer v2: Adapting modern image backbones to bird’s-eye-view recognition via perspective supervision. *ArXiv*, 2022. 1, 2, 4
- [56] Yung-Hsu Yang, Tobias Fischer, Thomas E. Huang, René Zurbrügg, Tao Sun, and Fisher Yu. Vis4D. <https://github.com/SysCV/vis4d>, 2024. 5
- [57] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021. 1
- [58] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9043–9053, 2023. 2, 4, 5, 8
- [59] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412, 2018. 3
- [60] Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. Open-vocabulary detr with conditional matching. In *European Conference on Computer Vision*, pages 106–122. Springer, 2022. 2
- [61] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14393–14402, 2021. 2, 3
- [62] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 2, 3
- [63] Renrui Zhang, Han Qiu, Tai Wang, Xuanzhuo Xu, Ziyu Guo, Yu Qiao, Peng Gao, and Hongsheng Li. Monodetr: Depth-guided transformer for monocular 3d object detection. *ICCV 2023*, 2022. 1
- [64] Xiangyu Zhao, Yicheng Chen, Shilin Xu, Xiantai Li, Xinjiang Wang, Yining Li, and Haian Huang. An open and comprehensive pipeline for unified object grounding and detection. *arXiv preprint arXiv:2401.02361*, 2024. 2, 5
- [65] Brady Zhou, Philipp Krähenbühl, and Vladlen Koltun. Does computer vision matter for action? *Science Robotics*, 4, 2019. 1
- [66] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022. 2
- [67] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 3

# 3D-MOOD: Lifting 2D to 3D for Monocular Open-Set Object Detection

## Supplementary Material

This supplementary material elaborates more details of our main paper. In Sec. A, we illustrate how the proposed canonical image space can reduce the required GPU resource for training and how it helps the model learn better prior geometry. In Sec. B, we compare our proposed geometry-aware query generation to the virtual depth proposed by Cube R-CNN [4]. In Sec. C, we provide further details of our proposed open-set benchmark. We analyze the depth estimation performance in Sec. D, backbone comparison in Sec. E and FPS in Sec. F, respectively. In Sec. G, we discuss the importance of our proposed open detection score (ODS) and compare the evaluation results in detail to the IoU-based AP. Finally, we provide more qualitative results in Sec. H for both closed-set and open-set settings.

### A. Canonical Image Space

As shown in the main paper, we compare different resizing and padding strategies for training the model. Given the training batch size as 2, we have two training samples having very different image ratios, *e.g.*  $[376 \times 1241]$  and  $[1920 \times 1080]$ . The first strategy as [4] will find the shortest edge and resize it to the desired value, *e.g.* 512, and conduct the right-bottom padding to align the two samples' resolutions. This will lead to considerable padding for the portrait image, while not change the camera intrinsic  $\mathbf{K}$ .

The second strategy is like Grounding DINO (G-DINO) [27], which will find the longest edge and resize it to the desired value, *e.g.* 1333, and use the same right-bottom padding to align the resolutions. This leads to considerable padding for the landscape image, while the padding will also not change the camera intrinsic  $\mathbf{K}$ . Both strategies will increase the GPU usage for unnecessary padding and lead to different image resolutions for the same sampled image according to the paired images.

On the other hand, our proposed canonical image space fixes the image resolutions, *e.g.*  $[800 \times 1333]$ , and will resize the longest or the shortest edge considering the image ratios. As shown in Tab. 5, our methods successfully reduce the needed GPU resources compared to the previous methods. Furthermore, we use the center padding to ensure our image space will affect the camera intrinsic  $\mathbf{K}$  accordingly to unify it across not only the training and testing time but also across datasets.

During inference time, the same camera will capture the same image shape with the same camera intrinsics. The previous methods will fail to align the same observation between training and inference time. We speculate that this will hinder the model's understanding of the relation be-

**Table 5. GPU RAM Consumption.** We compare the GPU resource usage of different training padding and resizing methods. We show the results of training our full model using Swin-T [30] and batch size of 2 using gradient checkpointing on a RTX 4090.

Resize	Padding	Image Resolutions	GPU RAM (G)
Short Edge	Right-Bottom	Short Edge to 512	21
Long Edge	Right-Bottom	Long Edge to 1333	23
Ours	Center	$800 \times 1333$	17

**Table 6. Comparison with Virtual Depth.** We compare our geometry-aware 3D query generation (GA) with the virtual depth proposed by Cube R-CNN [4]. The results shows that GA converge better than the virtual depth mechanism.

Method	Virtual Depth	GA	$AP_{3D}^{omni} \uparrow$
Cube R-CNN [4]	✓	-	23.3
Ours (Swin-T)	✓	-	21.6
Ours (Swin-T)	-	✓	<b>26.8</b>

tween intrinsics, image shape, and metric depth. On the contrary, our canonical image space will keep the image and intrinsic consistent. As shown in the ablation studies of the main paper, the model benefits from the proposed canonical image space for both closed-set and open-set settings with even fewer GPU resource requirements for training.

### B. Comparison with Virtual Depth

We compare our proposed geometry-aware 3D query generation with the virtual depth proposed in [4]. As shown in Tab. 6, the virtual depth leads our model to converge much slower than our proposed geometry-aware 3D query generation. We speculate that virtual depth requires much more training time to learn universal geometry, which also leads to underperformance.

### C. Open-set Benchmark

We show more details about our proposed open-set benchmark and list the full classes for each datasets in Tab. 7.

#### C.1. Argoverse 2

Compared to other autonomous driving datasets, Argoverse 2 (AV2) [54] possesses more diverse classes. Moreover, the resolutions of the front camera are portrait images, providing unseen cameras and domains. Those factors make AV2 a challenging dataset for benchmarking open-set monocular 3D object detection. We sample every 5 frame from the official validation split and obtain 4806 images as the open-set

Table 7. **Classes for Argoverse 2 and ScanNet.** We list the base and novel categories for our proposed open-set benchmark. For ScanNet, the bold categories are the supercategories. We further list all 168 categories of ScanNet200 settings.

Dataset	Base	Novel
Argoverse 2	regular vehicle, pedestrian, bicyclist, construction cone, construction barrel, large vehicle, bus, truck, vehicular trailer, bicycle, motorcycle	motorcyclist, wheeled rider, bollard, sign, stop sign, box truck, articulated bus, mobile pedestrian crossing sign, truck cab, school bus, wheeled device, stroller
ScanNet	<b>cabinet</b> (cabinet, kitchen cabinet, file cabinet, bathroom vanity, bathroom cabinet, cabinet door, trash cabinet, media center), <b>bed</b> (bed, mattress, loft bed, sofa bed, air mattress), <b>chair</b> (chair, office chair, armchair, sofa chair, folded chair, massage chair, recliner chair, rocking chair), <b>sofa</b> (couch, sofa), <b>table</b> (table, coffee table, end table, dining table, folded table, round table, side table, air hockey table), <b>door</b> (door, doorframe, bathroom stall door, closet door, mirror door, glass door, sliding door, closet doorframe), <b>window</b> , <b>picture</b> (picture, poster, painting), <b>counter</b> (kitchen counter, counter, bathroom counter), <b>desk</b> , <b>curtain</b> , <b>refrigerator</b> (refrigerator, mini fridge, cooler), <b>toilet</b> (toilet, urinal), <b>sink</b> , <b>bathtub</b>	<b>bookshelf</b> , <b>shower curtain</b> , <b>other furniture</b> (trash can, radiator, recycling bin, ottoman, bench, tv stand, wardrobe, trash bin, seat, closet, ladder, piano, water cooler, stand, washing machine, rack, wardrobe, clothes dryer, ironing board, keyboard piano, music stand, furniture, crate, drawer, footrest, piano bench, foosball table, footstool, compost bin, tripod, treadmill, chest, folded ladder, drying rack, pool table, heater, toolbox, beanbag chair, dollhouse, ping pong table, clothing rack, podium, luggage stand, rack stand, futon, book rack, workbench, easel, headboard, display rack, crib, bedframe, bunk bed, magazine rack, furnace, stepladder, baby changing station, flower stand, display)
ScanNet200	chair, table, door, couch, cabinet, shelf, desk, office chair, bed, pillow, sink, picture, window, toilet, bookshelf, monitor, curtain, book, armchair, coffee table, box, refrigerator, lamp, kitchen cabinet, towel, clothes, tv, nightstand, counter, dresser, stool, plant, bathtub, end table, dining table, keyboard, bag, backpack, toilet paper, printer, tv stand, whiteboard, blanket, shower curtain, trash can, closet, stairs, microwave, stove, shoe, computer tower, bottle, bin, ottoman, bench, board, washing machine, mirror, copier, basket, sofa chair, file cabinet, fan, laptop, shower, paper, person, paper towel dispenser, oven, blinds, rack, plate, blackboard, piano, suitcase, rail, radiator, recycling bin, container, wardrobe, soap dispenser, telephone, bucket, clock, stand, light, laundry basket, pipe, clothes dryer, guitar, toilet paper holder, seat, speaker, column, ladder, cup, jacket, storage bin, coffee maker, dishwasher, paper towel roll, machine, mat, windowsill, bar, bulletin board, ironing board, fireplace, soap dish, kitchen counter, doorframe, toilet paper dispenser, mini fridge, fire extinguisher, ball, hat, shower curtain rod, water cooler, paper cutter, tray, pillar, ledge, toaster oven, mouse, toilet seat cover dispenser, cart, scale, tissue box, light switch, crate, power outlet, decoration, sign, projector, closet door, vacuum cleaner, headphones, dish rack, broom, range hood, hair dryer, water bottle, vent, mailbox, bowl, paper bag, projector screen, divider, laundry detergent, bathroom counter, stick, bathroom vanity, closet wall, laundry hamper, bathroom stall door, ceiling light, trash bin, dumbbell, stair rail, tube, bathroom cabinet, coffee kettle, shower head, case of water bottles, power strip, calendar, poster, mattress	

testing set. Among the official 26 classes, 23 appeared in the testing set, which contains 11 *base* and 12 *novel* classes.

## C.2. ScanNet

ScanNet [8] provides diverse indoor scenes with 18 super-categories as shown in Tab. 7. We uniformly sample maximum 20 frames from each scan in the official ScanNet validation splits and obtain total 6240 images as open-set testing set. Given that 15 supercategories are seen in the Omni3D training set, this benchmark still allows us to evaluate domain generalization, where Tab. 4 of the main paper indicates issues of previous works. Furthermore, in the rest 3 novel classes, the supercategory *other furniture* requires models to detect various types of furniture.

To further test 3D-MOOD, we extend ScanNet using the ScanNet200 setting, which has **168** thing classes appeared in the testing set. As shown in Tab. 8, 3D-MOOD can still achieve best performance given more diverse classes.

Table 8. **ScanNet200 Results.** 3D-MOOD achieves SOTA results given diverse novel categories in unseen scenes.

Method	AP <sub>3D</sub> <sup>dist</sup> ↑	mATE ↓	mASE ↓	mAOE ↓	ODS ↑
Cube R-CNN [4]	2.1	0.962	0.970	0.985	2.5
OVM3D-Det [18]	3.1	0.957	0.973	0.946	3.6
<b>Ours</b> (Swin-B)	<b>6.2</b>	<b>0.811</b>	<b>0.835</b>	<b>0.799</b>	<b>12.4</b>

## D. Metric Monocular Depth Estimation

Because auxiliary depth estimation (ADE) is only used to help with 3D object detection, we evaluated its effectiveness in this regard. As shown in Tab. 4 of our paper, ADE improves the closed set AP by 0.7, but reduces the performance for unseen scenes, indicating that ADE can only help for known scenes. We further evaluate our depth quality on the KITTI Eigen-split test set, where UniDepth [38] achieves 4.21% absolute relative error, Metric3Dv2 [58] has

Table 9. **Backbone comparison.** We ablate the choice of different model backbones. All experiments are trained with 12 epochs.

Backbone	Parameters	$AP_{3D}^{\text{omni}} \uparrow$
DLA-34 [59]	15M	24.9
Swin-Transformer (Tiny) [30]	29M	26.8
Swin-Transformer (Base) [30]	88M	28.2
ConvNeXt-B [31]	89M	28.4

Table 10. **Comparison between different matching criteria for evaluation.** The same detection results will have a huge AP difference when using different matching settings.

Matching	Pedestrian	Construction cone	Monitor	Door
IoU	7.4	0.5	0.5	2.0
Distance	26.2	6.5	9.4	24.2

4.4%, and 3D-MOOD obtains 9.1%. We believe it is due to limited training data, different training objectives, and the model backbones [30, 34].

## E. Backbone Comparison

As shown in Tab. 9, Swin-B works equally well with ConvNeXt-B, and 3D-MOOD is comparable to Cube R-CNN and Uni-MODE using the same backbone, but with much shorter training. This shows effectiveness of our proposed designs rather than the backbone [34].

## F. Inference Time

We compare the FPS on KITTI using an RTX 4090, and Cube R-CNN (DLA-34) can have 68 FPS while 3D-MOOD (Swin-T) can achieve 17 FPS. As a reference from the paper, Uni-MODE can obtain 21 FPS on a single A100.

## G. Open Detection Score (ODS)

Compared to point-cloud-based 3D object detectors, using a single image to estimate 3D objects requires the networks to predict metric depth, while the scales in depth are known in the point cloud. This extra challenge leads the monocular methods to fail to match the ground truth using IoU-based matching because of several centimeter error in depth, especially for the small or thin objects in the open-set settings.

To have a more suitable evaluation metric for monocular 3D object detection, we use the 3D Euclidean distance between prediction and GT as the matching criterion. With the dynamic matching threshold, *e.g.* *radius* of the GT 3D boxes,  $AP_{3D}^{\text{dist}}$  can be used for both indoor and outdoor scenes. As shown in Tab. 10, the same detection results on Argoverse 2 [54] and ScanNet [8] will have large AP differences depending on the matching criterion.

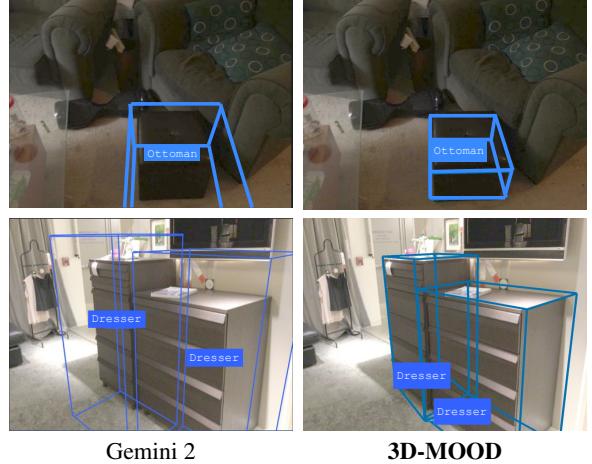


Figure 6. **Comparison with Gemini 2.** We qualitatively compare with Gemini 2 given the novel classes.

We also propose the normalized true positive errors (TPE) to further analyze the matched prediction. First, we compute the 3D Euclidean distance between prediction and GT, and we normalize the distance by the matching criterion as the translation error (TE). Second, we compute the IoU, *i.e.*  $IoU_{3D}$ , between prediction and GT after aligning the 3D centers and orientation and use  $1 - IoU_{3D}$  to measure the scale error (SE). Finally, we compute the SO3 relative angle between the prediction and GT normalized by  $\pi$  as the orientation error (OE). We average the TP errors across classes over different recall thresholds to get mATE, mASE, and mAOE. Using  $AP_{3D}^{\text{dist}}$  with the proposed normalized true positive errors to get ODS can provide a better matching criterion for 3D monocular object detection and still evaluate the localization, orientation, and dimension estimation at the same time.

## H. Qualitative Results

As shown in Fig. 6, we qualitatively compared our method with the closed-source Gemini 2 [48] beta functionality in 3D object detection, where 3D-MOOD provides more accurate localization. We provide more qualitative results in Fig. 7 for the open-set settings and Fig. 8 for the closed-set settings. We use the score threshold as 0.1 with class-agnostic nonmaximum suppression for better visualization.



Figure 7. **Open-set Qualitative Results.** We show more visualization on Argoverse 2 [54] and ScanNet [8].



Figure 8. **Closed-set Qualitative Results.** We show the qualitative results for 3D-MOOD on Omni3D [4] test set.