SOGDet: Semantic-Occupancy Guided Multi-View 3D Object Detection

Qiu Zhou[1*], Jinming Cao[2†*], Hanchao Leng[3], Yifang Yin[4], Yu Kun[3], Roger Zimmermann[2]

[1]Independent Researcher
[2]National University of Singapore
[3]Xiaomi Car
[4]Institute for Infocomm Research (I^2R), A*STAR, Singapore
{zhouqiulv, jinming.ccao, hanchao.leng}@gmail.com, yin_yifang@i2r.a-star.edu.sg, yukun@xiaomi.com,
rogerz@comp.nus.edu.sg

Abstract

In the field of autonomous driving, accurate and comprehensive perception of the 3D environment is crucial. Bird's Eye View (BEV) based methods have emerged as a promising solution for 3D object detection using multi-view images as input. However, existing 3D object detection methods often ignore the physical context in the environment, such as sidewalk and vegetation, resulting in sub-optimal performance. In this paper, we propose a novel approach called SOGDet (Semantic-Occupancy Guided Multi-view 3D Object Detection), that leverages a 3D semantic-occupancy branch to improve the accuracy of 3D object detection. In particular, the physical context modeled by semantic occupancy helps the detector to perceive the scenes in a more holistic view. Our SOGDet is flexible to use and can be seamlessly integrated with most existing BEV-based methods. To evaluate its effectiveness, we apply this approach to several state-of-the-art baselines and conduct extensive experiments on the exclusive nuScenes dataset. Our results show that SOGDet consistently enhance the performance of three baseline methods in terms of nuScenes Detection Score (NDS) and mean Average Precision (mAP). This indicates that the combination of 3D object detection and 3D semantic occupancy leads to a more comprehensive perception of the 3D environment, thereby aiding build more robust autonomous driving systems. The codes are available at: https://github.com/zhouqiu/SOGDet.

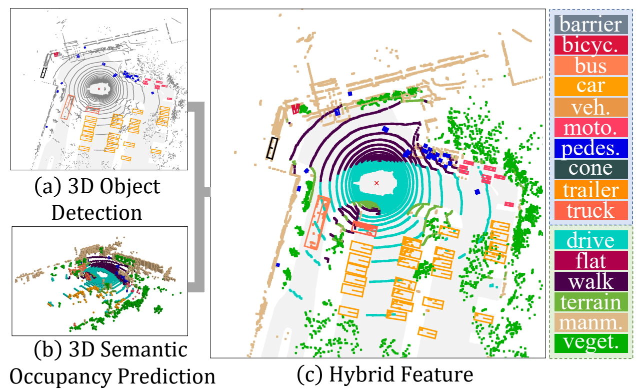


Figure 1: Illustration of 3D object detection and semantic occupancy prediction tasks. On the rightmost legend, the top 10 categories in the blue box are shared for both tasks, and the bottom 6 categories in the green box are exclusively used by semantic occupancy prediction. (a) 3D object detection usually focuses on objects on roads, such as bicycles and cars. In contrast, 3D semantic occupancy prediction (b) concerns more about physical contexts (e.g., sidewalk and vegetation) in the environment. By combining these two (c), we can obtain a more comprehensive perception of the traffic conditions, such as pedestrians and bicycles mainly on the sidewalk and cars and buses co-appearing on drive surface.

Introduction

Autonomous driving has become a burgeoning field for both research and industry, with a notable focus on achieving accurate and comprehensive perception of the 3D environment. Recently, Bird's Eye View (BEV) based methods (Huang et al. 2021; Li et al. 2022b,a) have attracted extensive attention in 3D object detection due to their effectiveness in reducing computational costs and footprints. The common paradigm is to take the multi-view images as inputs to detect objects, wherein the noticeable work BEVDet (Huang et al. 2021) serves as a strong baseline. BEVDet first extracts image features from multi-view images using a typical backbone network such as ResNet (He et al. 2016). The features are thereafter mapped to the BEV space with View Transformer (Philion and Fidler 2020), followed by a convolutional network and a target detection head. Inspired by BEVDet, following studies have integrated additional features into this framework, such as depth supervision (Li et al. 2022a) and temporal modules (Huang and Huang 2022).

Despite the significant improvement in localizing and classifying specific objects, *i.e.,* cars and pedestrians, most existing methods (Huang et al. 2021; Huang and Huang 2022; Li et al. 2022b,a) neglect the physical context in the environment. These contexts, such as roads, pavements and vegetation, though out of interest for detection, still offer important cues for perceiving the 3D scenes. For example, as shown in Figure 1, cars mostly appear on the drivable surface rather than the sidewalk. To harness such important features for object detection, we notice a recent emerging task

[*]These authors contributed equally.
[†]Corresponding author.

– 3D semantic-occupancy prediction (Huang et al. 2023; Li et al. 2023; Wei et al. 2023; Wang et al. 2023), that voxelizes the given image and then performs semantic segmentation of the resulting voxels. This task not only predicts the occupancy status but also identifies the objects within each occupied pixel, thereby enabling the comprehension of physical contexts. As shown in Figure 1, object detection and semantic occupancy prediction focuses on dynamic objects and environmental contexts, respectively. Combining these two leads to the hybrid features in Figure 1(c) would provide a more comprehensive description of the scene, such as the poses of cars driving on the drivable surface and the presence of pedestrians on sidewalk or crossings.

Motivated by this important observation, we propose a novel approach called SOGDet, which stands for **S**emantic-**O**ccupancy **G**uided Multi-view 3D Object **Det**ection. To the best of our knowledge, our method is the first of its kind to employ a 3D semantic-occupancy branch (OC) to enhance 3D object detection (OD). Specifically, we leverage a BEV representation of the scene to predict not only the pose and type of 3D objects (OD branch) but also the semantic class of the physical context (OC branch). SOGDet is a plug-and-play approach that can be seamlessly integrated with existing BEV-based methods (Huang et al. 2021; Huang and Huang 2022; Li et al. 2022a) for 3D object detection tasks. Moreover, to better facilitate the OD task, we extensively explore two labeling approaches for the OC branch, wherein the one predicts the *binary occupancy* label only and the other involves the *semantics* of each class. Based on these two approaches, we train two variants of SOGDet, namely SOGDet-BO and SOGDet-SE. Both variants significantly outperform the baseline method, demonstrating the effectiveness of our proposed method.

We conduct extensive experiments on the exclusive nuScenes (Caesar et al. 2020) dataset to evaluate the effectiveness of our proposed method. In particular, we apply SOGDet to several state-of-the-art backbone networks (He et al. 2016; Liu et al. 2021; Cao et al. 2021) and compare it to various commonly used baseline methods (Huang and Huang 2022; Li et al. 2022a). Our experimental results demonstrate that SOGDet consistently improves the performance of all tested backbone networks and baseline methods on the 3D OD task in terms of nuScenes Detection Score (NDS) and mean Average Precision (mAP). On the flip side, our OC approach surprisingly achieves comparable performance to state-of-the-art methods (Huang et al. 2023). This finding represents another promising side product and is beyond our expectation, as our intention is to design a simple network and sheds little light on it. The above results together highlight the effectiveness of the combination of 3D OD and OC in achieving comprehensive 3D environment understanding, and further enabling the development of robust autonomous driving systems.

Related Work

3D Object Detection (OD) constituents an indispensable component in autonomous driving (Arnold et al. 2019; Chen et al. 2017). Prior monocular methods (Ding et al. 2020; Cai et al. 2020; Kumar, Brazil, and Liu 2021; Reading et al. 2021) predict 3D bounding boxes using single-view images. For example, D4LCN (Ding et al. 2020) uses an estimated depth map to enhance image representation. Cai et al. (Cai et al. 2020) used object height prior to invert a 2D structured polygon into a 3D cuboid. However, due to the limitation of scarce data and single-view input, the model demonstrates difficulties in tackling more complex tasks (Huang et al. 2021). To overcome this problem, recent studies (Huang et al. 2021; Huang and Huang 2022; Li et al. 2022a) have been devoted to the development of large-scale benchmarks (Caesar et al. 2020; Sun et al. 2020) with multiple camera views. For example, inspired by the success of FCOS (Tian et al. 2019) in 2D detection, FCOS3D (Wang et al. 2021) treats the 3D OD problem as 2D-version. Based on FCOS3D, PGD (Wang et al. 2022a) presents using geometric relation graph to facilitate the targets' depth prediction. Benefited from the DETR (Carion et al. 2020) method, some approaches have also explored the validity of Transformer, such as DETR3D (Wang et al. 2022b) and Graph-DETR3D (Chen et al. 2022).

Unlike the aforementioned methods, BEVDet (Huang et al. 2021) leverages the Lift-Splat-Shoot(LSS) based (Philion and Fidler 2020) detector to perform 3D OD in multi-view. The framework is explicitly designed to encode features in the BEV space, making it scalable for multi-task learning, multi-sensor fusion and temporal fusion (Huang and Huang 2022). The framework is extensively studied by following work, such as BEVDepth (Li et al. 2022a) which enhances depth prediction by introducing a camera-aware depth network, and BEVFormer (Li et al. 2022b) which extends BEVDet on spatiotemporal dimension. Our proposed method also builds upon the BEVDet framework. Specifically, we introduce the semantic occupancy branch to guide the prediction of object detectors, a paradigm that has not been studied by existing efforts.

3D Semantic Occupancy Prediction (OC) has emerged as a popular task in the past two years (Cao and de Charette 2022; Huang et al. 2023; Li et al. 2023; Miao et al. 2023; Wei et al. 2023; Wang et al. 2023). It involves assigning an occupancy probability to each voxel in 3D space. The task offers useful 3D representations for multi-shot scene reconstruction, as it ensures the consistency of multi-shot geometry and helps obscured parts to be recovered (Shi et al. 2023).

The existing methods are relatively sparse in the literature. MonoScene (Cao and de Charette 2022) is the pioneering work that uses monocular images to infer dense 3D voxelized semantic scenes. However, simply fusing multi-camera results with cross-camera post-processing often leads to sub-optimal results. VoxFormer (Li et al. 2023) devises a two-stage framework to output the full 3D volumetric semantics from 2D images where the first stage uses a sparse collection of depth-estimated visible and occupied voxels, followed by a densification stage that generates dense 3D voxels from the sparse ones. TPVFormer (Huang et al. 2023) performs end-to-end training by using sparse LiDAR points as supervision, resulting in more accurate occupancy predictions.

Multi-Task Learning has become a common practice to employ perception tasks in BEV domain. Noteworthy con-

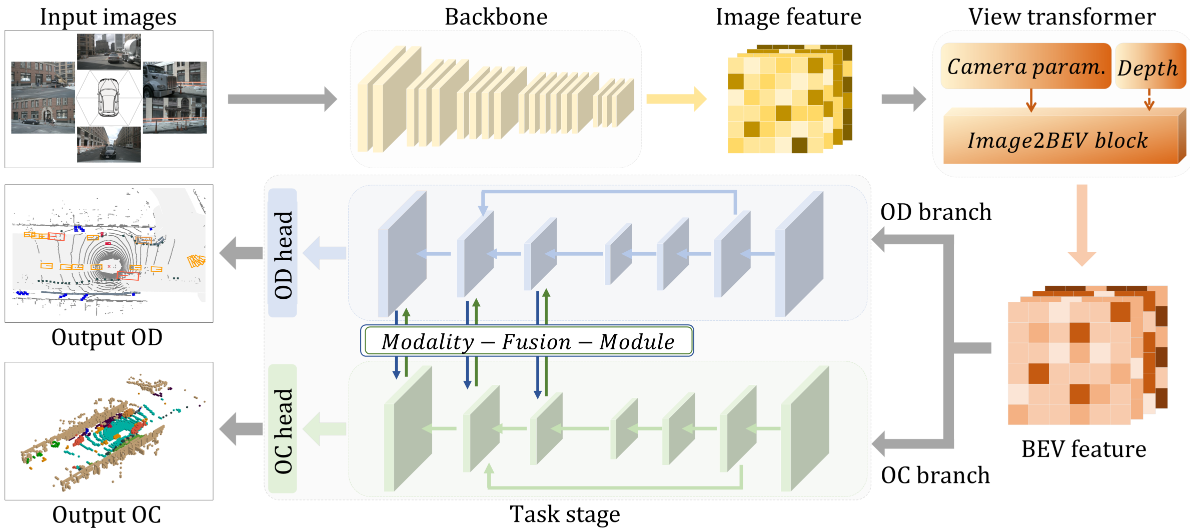


Figure 2: The overall network architecture. Our approach includes an image backbone (yellow) to encode multi-view input images to the vision feature, a view transformer (orange) to transform the vision feature into BEV feature, and a task stage comprising OD (blue) and OC (green) branches that respectively predict the OD and OC outputs in the same time.

tributions such as BEVFormer (Li et al. 2022b) and BEVerse(Zhang et al. 2022) exemplify this approach by integrating OD and map segmentation to enhance overall perception capabilities. LidarMultiNet (Ye et al. 2023) further extends the paradigm by utilizing OD as an auxiliary task, elevating semantic segmentation performance within the LiDAR context. The adoption of a multi-task framework is gaining prominence due to its ability to exploit the complementary advantages of diverse tasks, surpassing the capabilities of single-task approaches. This trend is increasingly recognized and favored within the industry.

Method

Overall Architecture and Notations

The overall architecture of our proposed method is illustrated in Figure 2 which is composed of three main components: an image backbone, a view transformer, and a task stage that predicts both OC and OD simultaneously. Specifically, the multi-view input images are first encoded by the image backbone, and then aggregated and transformed into the Bird-Eye-View (BEV) feature by the view transformer. With inherent camera parameters, the view transformer conducts depth-aware multi-view fusion and 4D temporal fusion simultaneously. Thereafter, the task stage generates both OC and OD features, which are interacted using a modality-fusion module. We finally predict the OD and OC outputs using their respective features.

To ensure the clearance and consistency throughout our presentation, we first define the following notations following the order of data flow within our pipeline.

I represents an image group with same height and width from N cameras using the same timestamp. $\boldsymbol{F_{img}} \in \mathbb{R}^{N\times C\times H\times W}$ represents feature map produced by the image backbone, where H, W and C means the height, width and channels of the feature map, respectively. $\boldsymbol{F_d} \in \mathbb{R}^{N\times D\times H\times W}$ represents depth estimation of the image group I. $\boldsymbol{F_{bev}} \in \mathbb{R}^{C_{bev}\times X\times Y}$ represents BEV features extracted by the view transformer, where $X \times Y$ and C_{bev} means the dimensions and the channels of the BEV feature following (Huang and Huang 2022), respectively. $\boldsymbol{F_{od}}$ and $\boldsymbol{F_{oc}}$ represent task-specific intermediate features of OD and OC branches in task stage.

For the camera parameters, we combine the offset vector and rotation matrix to represent the translation $\boldsymbol{TR} \in \mathbb{R}^{4\times 4}$ from source coordinate system to target coordinate system. For example, $\boldsymbol{TR_{cam}^{lid}}$ means a translation from camera coordinate system to lidar coordinate system. And $\boldsymbol{TR_{in}}$ represents the intrinsic parameters of all cameras.

For the output, the OD branch has two outputs: Bounding Box $\boldsymbol{B} \in \mathbb{R}^{M\times(3+3+2+2+1)}$ and Heatmap $\boldsymbol{H}$, where M is the total number of bounding boxes and the second dimension of $\boldsymbol{B}$ represents location, scale, orientation, velocity and attribute respectively. $\boldsymbol{Occ} \in \mathbb{R}^{O\times X\times Y\times Z}$ represents the OC branch output, which means that for the different grids from voxel grid $\boldsymbol{V} \in \mathbb{R}^{X\times Y\times Z}$, there are O semantic labels in total. And we generate the occupancy voxel grid from point cloud $\boldsymbol{P} \in \mathbb{R}^{K\times 3}$ of K points.

Image Backbone

The image backbone encodes the multi-view input images I into the feature map $\boldsymbol{F_{img}}$. Following previous work (Huang et al. 2021; Huang and Huang 2022), we sequentially concatenate ResNet (He et al. 2016) and FPN (Lin et al. 2017a) as our image backbone to extract the image feature. Moreover, we empirically found that using ShapeConv (Cao et al. 2021) instead of traditional convolutional layers in the image backbone leads to improved accuracy in the OD task without increasing model complexity during inference. In view of this, all ResNet-50 and -100 models in our method and baseine are replaced with ShapeConv for a fair comparison.

View Transformer

The view transformer converts the image feature $\boldsymbol{F_{img}}$ to the BEV feature $\boldsymbol{F_{bev}}$. We implement this module with the combination of BEVDepth (Li et al. 2022a) and BEVDet4D (Huang and Huang 2022) for better performance, namely BEVDet4D-depth, which jointly conducts depth-aware multi-view fusion and 4D temporal fusion based on BEVDepth and BEVDet4D, respectively.

Depth-Aware Multi-View Fusion. Following BEVDepth (Li et al. 2022a), the $\boldsymbol{F_d}$ feature is estimated by a Depth Network based on image feature $\boldsymbol{F_{img}}$ and camera parameter $\boldsymbol{TR_{in}}$ by,

$$\boldsymbol{F_d} = DepthNet(\boldsymbol{F_{img}}, \boldsymbol{TR_{in}}). \tag{1}$$

Here, we use the notation $DepthNet(*, *)$ to refer to the sub-network introduced in (Li et al. 2022a), which is composed of a series of convolutional layers and MLPs.

Then the Lift-Solat-Shoot(LSS) (Philion and Fidler 2020) is applied to calculate BEV feature $\boldsymbol{F_{bev}}$ as follows,

$$\boldsymbol{F_{bev}} = LSS(\boldsymbol{F_{img}}, \boldsymbol{F_d}, \boldsymbol{TR_{cam}^{lid}}), \tag{2}$$

where $LSS(*, *, *)$ is a depth-aware transformation following (Li et al. 2022a) which first lift the image feature $\boldsymbol{F_{img}}$ and its depth feature $\boldsymbol{F_d}$ into 3D lidar system by $\boldsymbol{TR_{cam}^{lid}}$, then splat 3D feature into 2D BEV plane to obtain $\boldsymbol{F_{bev}}$.

4D Temporal Fusion. Let $\boldsymbol{F_{bev}^{curr}}$ and $\boldsymbol{F_{bev}^{adj}}$ represent the BEV feature in the current timestamp and an adjacent timestamp respectively. We then apply a temporal fusion step following (Huang and Huang 2022) to aggregate $\boldsymbol{F_{bev}^{curr}}$ and $\boldsymbol{F_{bev}^{adj}}$ using Equation 3,

$$\boldsymbol{F_{bev}} = Concat[\boldsymbol{F_{bev}^{curr}}, \boldsymbol{F_{bev}^{adj}}] \tag{3}$$

where $Concat[*, *]$ represents the concatenation of two matrices along the channel dimension.

Task Stage

The task stage consists of two branches that take the BEV feature $\boldsymbol{F_{bev}}$ as input to obtain the Bounding Boxes $\boldsymbol{B}$ and Heatmap $\boldsymbol{H}$ outputs for OD branch and the Occpancy output $\boldsymbol{Occ}$ for OC branch, respectively.

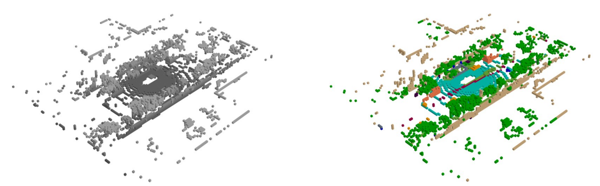
On the one hand, the OD branch is our primary task branch, which performs a 10-class object detection on car, truck, etc. On the other hand, the OC branch is to facilitate object detection by generating a 3D geometrical voxel around the ego vehicle.

To refine the BEV feature $\boldsymbol{F_{bev}}$ in both branches, we first apply a 3-layers ResNet (He et al. 2016) to extract intermediate features $\boldsymbol{F_{od}}$ and $\boldsymbol{F_{oc}}$ in three different resolution, which are 1/2, 1/4, 1/8 of the height, width respectively. A pyramid network (Lin et al. 2017a) is then employed to upsample the features to the same size as the original one. For the OD branch, we use CenterPoint (Yin, Zhou, and Krahenbuhl 2021) to produce the final predicted heatmap $\boldsymbol{H}$ and bounding boxes $\boldsymbol{B}$ from $\boldsymbol{F_{od}}$. For the OC branch, a simple 3D-Conv Head (Fang Ming 2023) is used to generate occupancy voxel grid $\boldsymbol{Occ}$ from $\boldsymbol{F_{oc}}$.

Modality-Fusion Module. The modality-fusion module is essential in our method to perform interactions between the above two branches. We define $\mathbb{G}_{C\to D}$ to adapt the features from OC to OD, and vice versa with $\mathbb{G}_{D\to C}$. We employ a weighted average operation parameterized by λ to fuse features from different modalities and empirically set $\lambda = 0.9$,

$$\begin{cases} \boldsymbol{F_{od}} = (1-\lambda)\cdot \mathbb{G}_{C\to D}(\boldsymbol{F_{oc}}) + \lambda \cdot \boldsymbol{F_{od}}, \\ \boldsymbol{F_{oc}} = (1-\lambda)\cdot \mathbb{G}_{D\to C}(\boldsymbol{F_{od}}) + \lambda \cdot \boldsymbol{F_{oc}}. \end{cases} \tag{4}$$

Taking OC to OD as example, the Equation 4 above shows that feature $\boldsymbol{F_{od}}$ in branch OD are $1-\lambda$ replaced by feature $\mathbb{G}_{C\to D}(\boldsymbol{F_{oc}})$ from branch OC. $\mathbb{G}_{C\to D}$ serves as a filter to reduce the modality gap between OD and OC. The operation takes effect when the BEV feature is upsampled in their own branches each time in the pyramid network (Lin et al. 2017a) mentioned above. We will demonstrate that this strategy can effectively enhance the information that is ignored by their original branch and thus fill the modality gap.



(a) Occupancy coarse labeling (b) Semantic fine labeling

Figure 3: Illustration of the two types of labels.

Occupancy Label Generation

We leverage two types of supervision signals for the OC branch. One is *binary occupancy* label **BO**, whose supervision is binary with 0 and 1 representing empty and occupied voxels, respectively. The other is *semantic* label **SE**, containing 16 semantic labels such as barrier, bicycle, etc. Figure 3 illustrates the two types of label.

To generate the binary occupancy labels, we consider only the geometry features of each voxel and illustrate this procedure in Algorithm 1. This approach is cost-friendly and require no extra manual annotations.

For semantic label, we observe that directly using the sparse semantic occupancy points as ground-truth labels leads to unstable training. Therefore, we follow TPV-Former (Huang et al. 2023) to optimize the supervision voxel generation, where the voxels without semantic labels are masked and ignored.

Training Objectives

Losses of OD Branch. We adopt the CenterPoint Head (Yin, Zhou, and Krahenbuhl 2021) to produce the final OD bounding box prediction, based on which a Gaussian focal loss (Lin et al. 2017a) and an L1 loss are jointly computed. In the following, we will sequentially elaborate these two loss functions.

Gaussian focal loss emphasizes more on the overall difference between predicted and actual values across the entire plane. $\boldsymbol{H}$ denotes the heatmap output by the OD branch,

Algorithm 1: Binary occupancy label generation

Data: Point Cloud $\mathbf{P}$, Dimension Bound X_{min}, X_{max}, Y_{min}, Y_{max}, Z_{min}, Z_{max}, Resolution R_X, R_Y, R_Z
Result: Voxel Grid $\mathbf{V}$
/* Transform position of points into grid index */
for $p \in \mathbf{P}$ **do**
  $p_X, p_Y, p_Z \leftarrow p$
  for $axis \in \{X, Y, Z\}$ **do**
    if $axis_{min} \leq p_{axis} \leq axis_{max}$ **then**
      $p_{axis} \leftarrow \frac{(p_{axis} - axis_{min})}{R_{axis}}$
    else
      $\mathbf{P} \leftarrow \mathbf{P} - \{p\}$ /* Delete out of bound */
      break
/* Calculate the scale of output voxel grid */
$X \leftarrow \frac{X_{max} - X_{min}}{R_X}$, $Y \leftarrow \frac{Y_{max} - Y_{min}}{R_Y}$,
$Z \leftarrow \frac{Z_{max} - Z_{min}}{R_Z}$
build $\mathbf{V} \in \mathbb{R}^{X \times Y \times Z}$
/* Fill voxels */
for $v \in \mathbf{V}$ **do**
  if $index(v) \in \mathbf{P}$ **then** $v \leftarrow 1$ **else** $v \leftarrow 0$

which is a probability matrix recording the likelihood of each pixel belonging to any of the 10 classes. We then embed the real annotations into a 2D image with the same size as $\boldsymbol{H}$, forming the ground-truth heatmap $\widehat{\boldsymbol{H}}$, namely, a one-hot matrix. The Gaussian focal loss is then computed as,

$$L_G = -\lfloor \widehat{\boldsymbol{H}} \rfloor log(\boldsymbol{H})(1-\boldsymbol{H})^{\alpha} - (1-\widehat{\boldsymbol{H}})^{\gamma} log(1-\boldsymbol{H})\boldsymbol{H}^{\alpha}, \tag{5}$$

where $\lfloor * \rfloor$ denotes the floor operation, $\alpha = 2.0$ and $\gamma = 4.0$ are parameters of intensity following (Lin et al. 2017b).

L1 loss is employed to optimize bounding box statistics, *i.e.,* absolute distance location, scale, orientation, velocity and attribute, from a micro perspective. To this end, we estimate the L1 distance between predicted bounding box $\boldsymbol{B}$ and its ground-truth $\widehat{\boldsymbol{B}}$ as,

$$L_1 = \frac{1}{M} \cdot \sum_{m}^{M} |\boldsymbol{B_m} - \widehat{\boldsymbol{B_m}}|. \tag{6}$$

In this way, the total loss of OD branch is shown as,

$$L_{OD} = L_G + \mu_{od} L_1, \tag{7}$$

where μ_{od}=0.25 is the weight coefficient of OD branch.

Losses of OC Branch. We combine the cross entropy loss L_{ce} with class weight and lovász-softmax loss (Berman, Triki, and Blaschko 2018) L_{lova} following (Huang et al. 2023) in OC branch as Equation 8,

$$L_{OC} = L_{lova} + \mu_{oc} L_{ce} \tag{8}$$

where μ_{oc}=1 for SOGDet-SE and 6 for SOGDet-BO is the weight coefficient of OC branch. We set the same loss weight for all classes in SOGDet-SE and 1:2 for empty and occupied voxels in SOGDet-BO within L_{ce}, respectively.

Overall Objective. Combined the above loss functions together, we can define our final objective as below,

$$L = L_{OD} + \omega L_{OC}, \tag{9}$$

where ω is the balancing factor between the OC and OD branches. We empirically set $\omega = 10$ to maximize the effectiveness of our multi-task learning framework.

Method	Venue	NDS(%)↑	mAP(%)↑
PETR-Tiny	ECCV22	43.1	36.1
BEVDet-Tiny	arXiv22	39.2	31.2
DETR3D-R50	CoRL22	37.4	30.3
Ego3RT-R50	ECCV22	40.9	35.5
BEVDet-R50	arXiv22	37.9	29.8
BEVDet4D-R50	arXiv22	45.7	32.2
BEVDepth-R50	AAAI23	47.5	35.1
AeDet-R50	CVPR23	50.1	38.7
SOGDet-BO-R50	-	50.2	38.2
SOGDet-SE-R50	-	**50.6**	**38.8**
BEVerse-Small	arXiv22	49.5	35.2
PETR-R101	ECCV22	42.1	35.7
UVTR-R101	NIPS2022	48.3	37.9
PolarDETR-T-R101	arXiv22	48.8	38.3
BEVFormer-R101	ECCV22	51.7	41.6
BEVDepth-R101	AAAI23	53.5	41.2
PolarFormer-R101	AAAI23	52.8	43.2
AeDet-R101	CVPR23	56.1	44.9
SOGDet-BO-R101	-	55.4	43.9
SOGDet-SE-R101	-	**56.6**	**45.8**

Table 1: Performance comparison on the nuScenes validation set. As indicated in (Liu et al. 2021), the complexity of Swin-Tiny and -Small are similar to those of ResNet-50 and -101, respectively.

Experiments

Experimental Setup

Dataset and Metrics. We conducted extensive experiments on the nuScenes (Caesar et al. 2020) dataset, which is currently the exclusive benchmark for both 3D object detection and occupancy prediction. Following the standard practice (Huang et al. 2021; Feng et al. 2022), we used the official splits of this dataset: 700 and 150 scenes respectively for training and validation, and the remaining 150 for testing.

For OD task, we reported nuScenes Detection Score (NDS), mean Average Precision (mAP), mean Average Translation Error (mATE), mean Average Scale Error (mASE), mean Average Orientation Error (mAOE), mean Average Velocity Error (mAVE), and mean Average Attribute Error (mAAE). Among them, NDS and mAP are the more representative ones.

For OC task, we designed two types of occupancy labeling approaches. For the binary occupancy labeling approach, as we are the first to employ such labeling approach in the literature to the best of our knowledge, we only performed qualitative experiments. For the semantic labeling one, we maintained a consistent experimental protocol with the state-of-the-art method TPVFormer(Huang et al. 2023). Accordingly, we report the mean Intersection over Union (mIoU) of all semantic categories.

Method	Venue	NDS(%)↑	mAP(%)↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
FCOS3D (Wang et al. 2021)	ICCV21	42.8	35.8	0.690	0.249	0.452	1.434	0.124
DD3D (Park et al. 2021)	ICCV21	47.7	41.8	0.572	0.249	**0.368**	1.014	0.124
PGD (Wang et al. 2022a)	CoRL22	44.8	38.6	0.626	0.245	0.451	1.509	0.127
BEVDet (Huang et al. 2021)	arXiv22	48.2	42.2	0.529	**0.236**	0.395	0.979	0.152
BEVFormer (Li et al. 2022b)	ECCV22	53.5	44.5	0.631	0.257	0.405	0.435	0.143
DETR3D (Wang et al. 2022b)	CoRL22	47.9	41.2	0.641	0.255	0.394	0.845	0.133
Ego3RT (Lu et al. 2022)	ECCV22	47.3	42.5	0.549	0.264	0.433	1.014	0.145
PETR (Liu et al. 2022)	ECCV22	50.4	44.1	0.593	0.249	0.383	0.808	0.132
CMT-C (Yan et al. 2023)	ICCV23	48.1	42.9	0.616	0.248	0.415	0.904	0.147
PETRv2 (Liu et al. 2023)	ICCV23	55.3	45.6	0.601	0.249	0.391	0.382	**0.123**
X3KD (Klingner et al. 2023)	CVPR23	56.1	45.6	0.506	0.253	0.414	0.366	0.131
SOGDet-BO	-	57.8	47.1	0.482	0.248	0.390	**0.329**	0.125
SOGDet-SE	-	**58.1**	**47.4**	**0.471**	0.246	0.389	0.330	0.128

Table 2: Performance comparison on the nuScenes test set.

Method	**Venue**	**mIoU(%)↑**	category-wise IoU (%)↑															
			barr.	bicy.	bus	car	veh.	mot.	ped.	trai.	cone	truc.	driv.	flat	walk	terr.	man.	veg.
TPVFormer	CVPR23	**59.3**	**64.9**	27.0	**83.0**	**82.8**	38.3	27.4	**44.9**	24.0	55.4	**73.6**	**91.7**	**60.7**	**59.8**	**61.1**	**78.2**	76.5
SOGDet-SE	-	58.6	57.8	**30.7**	74.9	74.7	**43.7**	**42.0**	44.5	**32.7**	**62.6**	63.9	85.9	54.3	54.6	58.9	76.9	**80.2**

Table 3: Comparison with the State-of-the-Art OC method on the nuScenes val set.

Implementation Details. To demonstrate the effectiveness and generalization capabilities of SOGDet, we used several popular architectures (Li et al. 2022a; Huang and Huang 2022). To ensure that any improvements were solely due to our SOGDet, we kept most experimental settings, such as backbone and batch size untouched, and added only the OC branch. Unless otherwise noted, our baseline model is BEVDet4D-depth, which is a fusion of two recent multi-view 3D object detectors, BEVDepth (Li et al. 2022a) and BEVDet4D (Huang and Huang 2022). We followed the experimental protocol of AEDet (Feng et al. 2022) and training on eight 80G A100 GPUs with a mini-batch size of 8, for a total batch size of 64, and trained the model for 24 epochs with CBGS (Zhu et al. 2019) using AdamW as the optimizer with a learning rate of 2e-4.

Comparison with State-of-the-Art

We evaluated the performance of our SOGDet against other state-of-the-art multi-view 3D object detectors on the nuScenes validation and test sets.

Table 1 reports the results for the validation set using Swin-Tiny, -Small, ResNet-50 and -101 backbones. As shown in the table, our method achieves highly favorable model performance, with NDS scores of 50.2% and 55.4% for SOGDet-BO and 50.6% and 56.6% for SOGDet-SE on ResNet-50 and -101, respectively. These results surpass current state-of-the-art multi-view 3D object detectors with a large margin, including BEVDepth (Li et al. 2022a) (3.1% improvement in NDS at both ResNet-50 and -100) and AEDet (Feng et al. 2022) (0.5% improvement in NDS at both ResNet-50 and -100).

In Table 2, we present the results obtained by SOGDet with the ResNet-101 backbone on the nuScenes test set, where we report the performance of state-of-the-art methods that use the same backbone network for a fair comparison. We follow the same training strategy of existing approaches (Li et al. 2022a; Feng et al. 2022) that utilize both the training and validation sets to retrain the networks and without any test-time augmentation. SOGDet shows improved performance in multi-view 3D OD task with 58.1% NDS and 47.4% mAP, further verifying the effectiveness of our proposed approach.

Ablation Study

Comparison with the State-of-the-Art OC Method. To further evaluate the effectiveness of our approach, we compared our method with respect to semantic categories with TPVFormer (Huang et al. 2023) and presented the results in Table 3. Backbones from both methods take equivalent complexities.

The primary goal of our work is to enhance the 3D OD by integrating 3D OC. Despite its simpleness, results shown in Table 3 demonstrate that our SOGDet are comparable to TPVFormer, a state-of-the-art method specifically designed for the OC task. Moreover, our method even outperforms this baseline in certain categories such as bicycles, vegetation, and others, which indicates that the combination of the two branches can bring benefits for the OC branch as well, serving as another byproduct.

Different Baseline Architecture. Our proposed SOGDet is a flexible method that can be seamlessly integrated into most BEV-based multi-view object detection architectures. In order to evaluate the generalization capabilities of our method, we tested its effectiveness on several representative baseline architectures, namely BEVDet (Huang et al. 2021), BEVDet4D (Huang and Huang 2022), BEVDepth (Li et al. 2022a), and BEVDet4D-depth, using the nuScenes validation set. The results in Table 4 show that SOGDet consistently surpasses these baselines under various settings, which demonstrates the validity of our method to general-

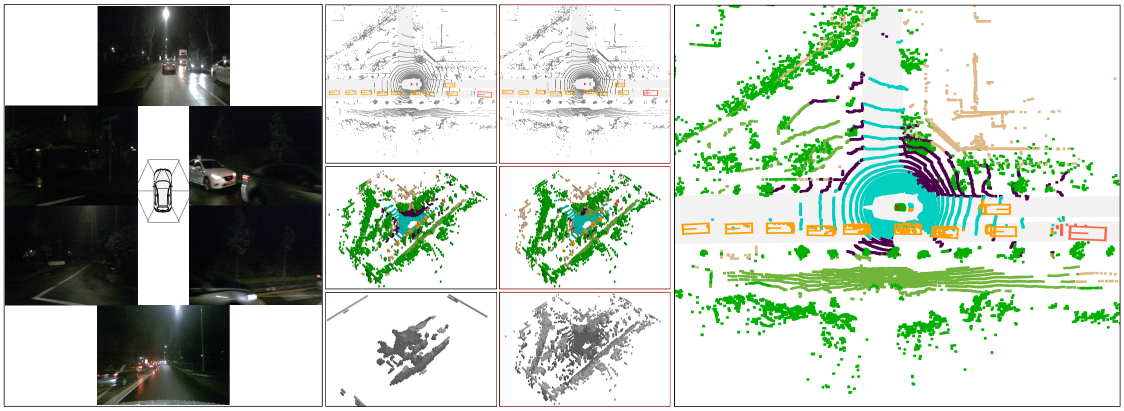


Figure 4: Visualization for the OD and OC branches of SOGDet. The input consists of six multi-view images. For both the output and the GT (red box) column, from top to bottom, we sequentially show the predictions of SOGDet-SE for OD, SOGDet-SE for OC and SOGDet-BO for OC. The Hybrid feature is blended from OD and OC branch predictions of SOGDet-SE.

ize to different model architectures.

BN.	Architecture	Method	mAP(%)	NDS(%)
Tiny	BEVDet	Baseline	31.2	39.2
		SOGDet-SE	32.9	41.5
	BEVDet4D	Baseline	33.8	47.6
		SOGDet-SE	34.6	48.7
R50	BEVDepth	Baseline	35.1	47.5
		SOGDet-SE	37.2	48.3
	BEVDet4D-depth	Baseline	37.0	49.0
		SOGDet-SE	38.8	50.6

Table 4: Performance comparison with different baselines.

Complexity Analysis. The efficiency concern is highly significant under resource-constrained environments. Pertaining to this aspect, we estimate metrics including floating point operations (FLOPs.) and parameter count (Param.), and show the results in Figure 5. It can be observed that compared with the state-of-the-art method AeDet (Feng et al. 2022), our SOGDet is more efficient especially on the more important metric FLOPs, *i.e.,* 252G v.s. 473G. Further, SOGDet outperforms AeDet by 0.5% in terms of NDS. This indicates that our method achieves a better trade-off between efficiency and model performance.

Visualization

Figure 4 illustrates qualitative results of our approach on the nuScenes (Caesar et al. 2020) dataset using ResNet-50 as the backbone for both OD and the OC branch. Pertaining to the object detection task, we focus only on occupied voxels, and therefore, locations marked as "empty" are not shown. The hybrid features reveal strong correlations between the physical structures and the location of the detected objects, such as vehicles, bicycles, and pedestrians. For example, vehicles are typically detected in drive surface, while bicycles and pedestrians are often detected on sidewalk. These findings are consistent with the observations and motivations of our paper and demonstrate that the integration of the two branches can lead to a better perception and understanding of the real world.

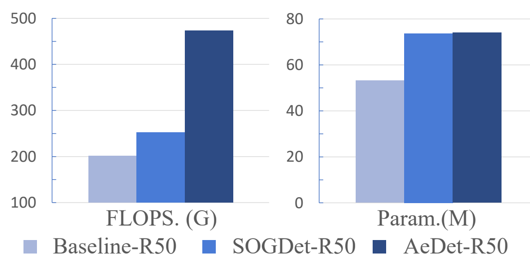


Figure 5: Parameter count (Param.) and floating-point operations (FLOPs).

Conclusion and Future Work

The Bird's Eye View (BEV) based method has shown great promise in achieving accurate 3D object detection using multi-view images. However, most existing BEV-based methods unexpectedly ignore the physical contexts in the environment, which is critical to the perception of 3D scenes. In this paper, we propose the SOGDet approach to incorporate such context using a 3D semantic occupancy approach. In particular, our SOGDet predicts not only the pose and type of each 3D object, but also the semantic classes of the physical contexts for finer-grained detection. Extensive experimental results on the nuScenes dataset demonstrate that our SOGDet consistently improves the model performance of several popular backbone networks and baseline methods.

In future work, we plan to explore the application of SOGDet with more auxiliary data inputs, such as lidar and radar, to further help the 3D object detection. Additionally, we believe that integrating 3D semantic-occupancy prediction into other autonomous driving tasks beyond 3D object detection, such as path planning and decision-making, may contribute a promising avenue for future research.

Acknowledgements

This work is supported by the Advanced Research and Technology Innovation Centre (ARTIC), the National University of Singapore under Grant (project number: A-8000969-00-00).

References

Arnold, E.; Al-Jarrah, O. Y.; Dianati, M.; Fallah, S.; Oxtoby, D.; and Mouzakitis, A. 2019. A survey on 3d object detection methods for autonomous driving applications. *IEEE Transactions on Intelligent Transportation Systems*, 20(10): 3782–3795.

Berman, M.; Triki, A. R.; and Blaschko, M. B. 2018. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4413–4421.

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.

Cai, Y.; Li, B.; Jiao, Z.; Li, H.; Zeng, X.; and Wang, X. 2020. Monocular 3d object detection with decoupled structured polygon estimation and height-guided depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 10478–10485.

Cao, A.-Q.; and de Charette, R. 2022. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3991–4001.

Cao, J.; Leng, H.; Lischinski, D.; Cohen-Or, D.; Tu, C.; and Li, Y. 2021. Shapeconv: Shape-aware convolutional layer for indoor rgb-d semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 7088–7097.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, 213–229. Springer.

Chen, X.; Ma, H.; Wan, J.; Li, B.; and Xia, T. 2017. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 1907–1915.

Chen, Z.; Li, Z.; Zhang, S.; Fang, L.; Jiang, Q.; and Zhao, F. 2022. Graph-DETR3D: rethinking overlapping regions for multi-view 3D object detection. In *Proceedings of the 30th ACM International Conference on Multimedia*, 5999–6008.

Ding, M.; Huo, Y.; Yi, H.; Wang, Z.; Shi, J.; Lu, Z.; and Luo, P. 2020. Learning depth-guided convolutions for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition workshops*, 1000–1001.

Fang Ming, Z. L. 2023. Occupancy Dataset for nuScenes. https://github.com/FANG-MING/occupancy-for-nuscenes.

Feng, C.; Jie, Z.; Zhong, Y.; Chu, X.; and Ma, L. 2022. AeDet: Azimuth-invariant Multi-view 3D Object Detection. *arXiv preprint arXiv:2211.12501*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Huang, J.; and Huang, G. 2022. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*.

Huang, J.; Huang, G.; Zhu, Z.; Ye, Y.; and Du, D. 2021. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*.

Huang, Y.; Zheng, W.; Zhang, Y.; Zhou, J.; and Lu, J. 2023. Tri-Perspective View for Vision-Based 3D Semantic Occupancy Prediction. *arXiv preprint arXiv:2302.07817*.

Klingner, M.; Borse, S.; Kumar, V. R.; Rezaei, B.; Narayanan, V.; Yogamani, S.; and Porikli, F. 2023. X3KD: Knowledge Distillation Across Modalities, Tasks and Stages for Multi-Camera 3D Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13343–13353.

Kumar, A.; Brazil, G.; and Liu, X. 2021. Groomed-nms: Grouped mathematically differentiable nms for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8973–8983.

Li, Y.; Ge, Z.; Yu, G.; Yang, J.; Wang, Z.; Shi, Y.; Sun, J.; and Li, Z. 2022a. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. *arXiv preprint arXiv:2206.10092*.

Li, Y.; Yu, Z.; Choy, C.; Xiao, C.; Alvarez, J. M.; Fidler, S.; Feng, C.; and Anandkumar, A. 2023. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. *arXiv preprint arXiv:2302.12251*.

Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Qiao, Y.; and Dai, J. 2022b. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IX*, 1–18. Springer.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017a. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017b. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Liu, Y.; Wang, T.; Zhang, X.; and Sun, J. 2022. Petr: Position embedding transformation for multi-view 3d object detection. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVII*, 531–548. Springer.

Liu, Y.; Yan, J.; Jia, F.; Li, S.; Gao, A.; Wang, T.; Zhang, X.; and Sun, J. 2023. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, 10012–10022.

Lu, J.; Zhou, Z.; Zhu, X.; Xu, H.; and Zhang, L. 2022. Learning ego 3d representation as ray tracing. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, 129–144. Springer.

Miao, R.; Liu, W.; Chen, M.; Gong, Z.; Xu, W.; Hu, C.; and Zhou, S. 2023. Occdepth: A depth-aware method for 3d semantic scene completion. *arXiv preprint arXiv:2302.13540*.

Park, D.; Ambrus, R.; Guizilini, V.; Li, J.; and Gaidon, A. 2021. Is pseudo-lidar needed for monocular 3d object detection? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3142–3152.

Philion, J.; and Fidler, S. 2020. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, 194–210. Springer.

Reading, C.; Harakeh, A.; Chae, J.; and Waslander, S. L. 2021. Categorical depth distribution network for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8555–8564.

Shi, Y.; Jiang, K.; Li, J.; Wen, J.; Qian, Z.; Yang, M.; Wang, K.; and Yang, D. 2023. Grid-Centric Traffic Scenario Perception for Autonomous Driving: A Comprehensive Review. *arXiv preprint arXiv:2303.01212*.

Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. 2020. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2446–2454.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9627–9636.

Wang, T.; Xinge, Z.; Pang, J.; and Lin, D. 2022a. Probabilistic and geometric depth: Detecting objects in perspective. In *Conference on Robot Learning*, 1475–1485. PMLR.

Wang, T.; Zhu, X.; Pang, J.; and Lin, D. 2021. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 913–922.

Wang, X.; Zhu, Z.; Xu, W.; Zhang, Y.; Wei, Y.; Chi, X.; Ye, Y.; Du, D.; Lu, J.; and Wang, X. 2023. OpenOccupancy: A Large Scale Benchmark for Surrounding Semantic Occupancy Perception. *arXiv preprint arXiv:2303.03991*.

Wang, Y.; Guizilini, V. C.; Zhang, T.; Wang, Y.; Zhao, H.; and Solomon, J. 2022b. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, 180–191. PMLR.

Wei, Y.; Zhao, L.; Zheng, W.; Zhu, Z.; Zhou, J.; and Lu, J. 2023. SurroundOcc: Multi-Camera 3D Occupancy Prediction for Autonomous Driving. *arXiv preprint arXiv:2303.09551*.

Yan, J.; Liu, Y.; Sun, J.; Jia, F.; Li, S.; Wang, T.; and Zhang, X. 2023. Cross modal transformer via coordinates encoding for 3d object dectection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Ye, D.; Zhou, Z.; Chen, W.; Xie, Y.; Wang, Y.; Wang, P.; and Foroosh, H. 2023. Lidarmultinet: Towards a unified multi-task network for lidar perception. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 3231–3240.

Yin, T.; Zhou, X.; and Krahenbuhl, P. 2021. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11784–11793.

Zhang, Y.; Zhu, Z.; Zheng, W.; Huang, J.; Huang, G.; Zhou, J.; and Lu, J. 2022. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *arXiv preprint arXiv:2205.09743*.

Zhu, B.; Jiang, Z.; Zhou, X.; Li, Z.; and Yu, G. 2019. Class-balanced grouping and sampling for point cloud 3d object detection. *arXiv preprint arXiv:1908.09492*.