AeDet: Azimuth-invariant Multi-view 3D Object Detection

Chengjian Feng Zequn Jie Yujie Zhong Xiangxiang Chu Lin Ma
Meituan Inc.

Abstract

Recent LSS-based multi-view 3D object detection has made tremendous progress, by processing the features in Bird-Eye-View (BEV) via the convolutional detector. However, the typical convolution ignores the radial symmetry of the BEV features and increases the difficulty of the detector optimization. To preserve the inherent property of the BEV features and ease the optimization, we propose an azimuth-equivariant convolution (AeConv) and an azimuth-equivariant anchor. The sampling grid of AeConv is always in the radial direction, thus it can learn azimuth-invariant BEV features. The proposed anchor enables the detection head to learn predicting azimuth-irrelevant targets. In addition, we introduce a camera-decoupled virtual depth to unify the depth prediction for the images with different camera intrinsic parameters. The resultant detector is dubbed Azimuth-equivariant Detector (AeDet). Extensive experiments are conducted on nuScenes, and AeDet achieves a 62.0% NDS, surpassing the recent multi-view 3D object detectors such as PETRv2 and BEVDepth by a large margin. Project page: https://fcjian.github.io/aeDet.

1. Introduction

In the field of autonomous driving, multi-view 3D object detection has been one of the most widely researched problems and received a lot of attention due to its low assembly cost and high efficiency. In the recent literature, such vision-based 3D object detector has made tremendous progress, especially for the Lift-Splat-Shoot (LSS) based methods [11,15]. They first transfer the image features from the image-view to Bird-Eye-View (BEV), and then process the BEV features via the convolutional backbone and detection head similar to 2D object detection [6,7,28].

However, the BEV features in multi-view 3D object detection are significantly different from the image features in 2D object detection: (1) the BEV features from the same camera are naturally endowed with radial symmetry; (2) the cameras have different orientations in BEV, and the BEV features from the different cameras are also approximately with radial symmetry. Simply using the typical 2D backbone and detection head to perform BEV perception (such as [11,15]) ignores the inherent property of the BEV features and suffers from two limitations discussed as follows: First, the BEV representation of the same imaging in different azimuths is inconsistent. The typical convolution shares the kernel weights and adopts the same regular sampling grid at each location of the features. Such design may destroy the radial symmetry of the BEV features, and thus is unfriendly to representation learning. To be specific, assume the six cameras capture the same imaging of the object, and the detector takes the same imaging as the input of the six views. BEVDepth generates different features and predictions for the same bus in different azimuths, while AeDet yields almost the same feature and prediction for the same bus in different azimuths.
translation-invariant and samples the inconsistent BEV features of the object in different azimuths (see Figure 3a for detailed demonstration). Consequently, as demonstrated in the ‘BEV feature’ column in Figure 1, the convolution of BEVDepth learns different features in different azimuths, increasing the difficulty of the representation learning.

Second, the prediction targets of the same imaging in different azimuths are inconsistent. The typical detection head predicts the object orientation and velocity along the Cartesian coordinates, which requires the detection head to predict different targets for the same imaging in different azimuths. Concretely, assume the different-view cameras capture the same imaging of the object at different moments. After mapping the imaging from the image to BEV, the object would have different orientations and velocities along the Cartesian coordinates in different azimuths (see Figure 3b for detailed demonstration). As a result, the detection head is required to predict different targets even for the same imaging in different azimuths, and inevitably increases the difficulty of the predictions.

To address the two limitations, we propose an Azimuth-equivariant Detector (AeDet) that aims to perform an azimuth-invariant BEV perception by modeling the property of radial symmetry to the network: (1) Azimuth-equivariant convolution. In contrast to the typical convolution that uses the same regular sampling grid at each location, we design an Azimuth-equivariant Convolution (AeConv) to rotate the sampling grid according to the azimuth at each location. AeConv enables the sampling grid equivariant to the azimuth and always in the radial direction of the camera. This allows the convolution to preserve the radial symmetry of the BEV features and unify the representation in different azimuths. (2) Azimuth-equivariant anchor. To unify the prediction targets in different azimuths, we propose an azimuth-equivariant anchor. Specifically, different from the typical anchor (anchor point or anchor box) defined along the Cartesian coordinates, the azimuth-equivariant anchor is defined along the radial direction and equivariant to the azimuth. We predict both the bounding box and velocity along the new anchor and its orthogonal directions, yielding the same prediction target for the same imaging of the object in different azimuths. Thus the azimuth-equivariant anchor enables the detection head to learn predicting azimuth-irrelevant targets. Notably, AeConv and the azimuth-equivariant anchor can work collaboratively to improve the consistency between the representation learning and predictions, as shown in the ‘BEV feature’ and ‘Predictions’ columns of AeDet in Figure 1.

In addition, we introduce a camera-decoupled virtual depth to improve the depth prediction, and ease the optimization of the depth network. In specific, we decouple the camera’s intrinsic parameters from the depth network, enabling the depth network to model the relationship between the image feature and the virtual depth. In this way, the depth network only needs to learn to predict a universal virtual depth, regardless of the intrinsic parameters of different cameras. Finally, we map the virtual depth to the real depth according to the classic camera model.

To summarize, we make the following contributions: (1) We design an azimuth-equivariant convolution to unify the representation learning in different azimuths, and extract the azimuth-invariant BEV features. (2) We propose a new azimuth-equivariant anchor to redefine the anchor along the radial direction and unify the prediction targets in different azimuths. (3) We introduce a camera-decoupled virtual depth to unify the depth prediction for the images captured by different cameras. (4) We conducted extensive experiments on nuScenes [1], where our AeDet significantly improves the accuracy of the object orientation (by 5.2%) and velocity (by 6.6%). AeDet achieves 62.0% NDS on the nuScenes test set, surpassing the recent multi-view object detectors such as BEVDepth [15] by a large margin.

2. Related Work

Single-view 3D object detection. Many works attempt to predict the 3D bounding boxes directly from the single-view image. For example, Ding et al. [5] proposes to use the estimated depth map to improve the image representation for 3D object detection. Cai et al. [2] inversely projects the 2D structured polygon to a 3D cuboid by providing the object height prior. Wang et al. [25] makes FCOS [24] work in 3D detection by decoupling the 3D targets as 2D and 3D attributes. Different from these methods, some researchers [26, 26] predict the 3D bounding boxes based on a pseudo-LiDAR. They convert the image-based depth to pseudo-LiDAR representations and adopt the LiDAR-based detectors to obtain the 3D predictions. Instead of pseudo-LiDAR, Reading et al. [22] projects the image feature to the efficient BEV feature to perform 3D object detection.

Multi-view 3D object detection. Generally speaking, modern 3D object detection frameworks can be divided into LSS-based and query-based methods. LSS-based methods project the image feature from the image-view to BEV using Lift-Splat-Shoot [21], and process the BEV features via the BEV-based detectors [28]. Huang et al. [11] first proposes the LSS-based detector BEVDet for multi-view object detection. Li et al. [15] enhances the depth prediction by introducing a camera-aware depth network and supervising the network with the depth generated from LiDAR point cloud. In contrast, query-based methods project the object query to the image-view and sample the image features to perform 3D object detection with transformer. Wang et al. [27] first extends DETR [3] to DETR3D for multi-view 3D object detection. Liu et al. [17] further improves DETR3D with the 3D position-aware representations. Li et al. [16] leverages the deformable transformer to fuse the image fea-
depth estimation. Single-view depth estimation aims to predict the depth from a single image. Saxena et al. [23] collects the monocular images and the corresponding ground-truth depth, and learns the depth with a supervised learning manner. Garg et al. [9] learns the single-view depth via an unsupervised framework from the stereo image pair. Kuznietsov et al. [13] trains the network in a semi-supervised way where they use the sparse ground-truth depth for supervision. Fu et al. [8] recasts the depth network learning as an ordinal regression problem and uses a multi-scale network to improve the depth prediction. In the multi-view 3D detection, Huang et al. [11] learns the single-view depth with the supervision from the 3D bounding boxes, and Li et al. [15] improves the depth learning by using the supervision from the LiDAR point cloud.

3. Methodology

In this paper, we aim to make an azimuth-invariant Bird-Eye-View (BEV) perception for multi-view 3D object detection, and thus unify the representation learning and predictions in different azimuths from BEV and boost the detection performance, as illustrated in Figure 2. This section is organized as follows, we start by introducing the common paradigm of the LSS-based multi-view 3D object detection. In Section 3.2, we propose an Azimuth-equivariant Convolution (AeConv) to extract the azimuth-invariant BEV representation. In Section 3.3, we further design a new azimuth-equivariant anchor to unify the prediction targets in different azimuths. Additionally, we introduce a camera-decoupled virtual depth to unify the depth prediction for the multi-camera images and ease the prediction in Section 3.4.

3.1. LSS-based Multi-view 3D Object Detection

Generally speaking, a popular LSS-based multi-view 3D object detector, for example, BEVDepth [15], is consisted of an image-view encoder, a view transformer, a BEV encoder and a detection head.

Image-view encoder. It is a typical 2D backbone such as ResNet [10], with the aim to extract the high-level features of the multi-view images.

View transformer. It first predicts the depth based on the image features, and then transforms the image features from image-view to BEV according to the predicted depth. BEVDepth [15] improves the depth network by introducing the camera’s intrinsic parameters into the features with a camera-aware attention module.

BEV encoder. It adopts a typical convolutional encoder similar to the image-view 2D backbone to further encode the BEV features.

Detection head. It usually leverages the head structure designed in LiDAR point-cloud based 3D object detection, for example, CenterPoint [28]. The detection head of CenterPoint (the first stage) adopts an anchor-free structure, which predicts the center offset, size, orientation and velocity of the object at each location.

3.1.1 Discussion on Limitations

The BEV features from the same camera are naturally endowed with radial symmetry. Furthermore, the cameras...
Figure 3. Illustration of the azimuth-equivariant convolution and the azimuth-equivariant anchor. Assume the same car precisely faces the focal point of the camera as the center point of the system, azimuth system based on the camera, in which we let the ascending center point of the azimuth system. The ascending direction and reference direction of the azimuth are the same as the ones defined above. In this work, we build an AeConv-based BEV network by replacing the typical convolution with the proposed AeConv. It enables to extract the azimuth-invariant BEV features along the radial direction, largely easing the optimization of the BEV network.

3.3. Azimuth-equivariant Anchor

Till here, we have extracted the azimuth-invariant representation. In this section, we further introduce a new azimuth-equivariant anchor to enable the detection head to make a unified prediction in different azimuths. The detection head based on the azimuth-equivariant anchor predicts the detection results different from the previous methods in two aspects. First, it predicts the relative angle of the object orientation w.r.t the azimuth at each location, instead of the absolute object orientation. Second, it predicts both
the center offset and velocity along the radial direction and its orthogonal direction. In the following, we first discuss the implicit anchor existing in the anchor-free methods, and then elaborate on the proposed azimuth-equivariant anchor.

**Implicit anchor.** The anchor-free detection head is the most commonly used head structure in LSS-based 3D object detectors such as BEVDet and BEVDepth. It actually could be seen as a specific case of the anchor-based detection head with an implicit anchor, namely, an anchor box with zero length, width and height.

To be specific, the anchor-based detection head densely defines the anchor \((x, y, z, l, w, h, \theta)\) at each location, where \((x, y, z)\), \((l, w, h)\) and \(\theta\) are the location, size and orientation of the anchor respectively. The anchor-free detection head follows a similar paradigm, but sets \(l = w = h = \theta = 0\), which can be viewed as having implicit anchors.

**Azimuth-equivariant anchor.** Due to the azimuth-irrelevant design, the existing anchor-free detection head is required to predict different targets even for the same imaging in different azimuths. This inevitably increases the difficulty of the predictions. To ease the prediction, we design an azimuth-equivariant anchor to enable the detection head to predict the same targets for the same imaging, i.e., being azimuth-independent. Essentially, the detection head can therefore predict the relative angle (shown in Figure 4) and the velocity/center offset along the consistent directions (shown in Figure 3b).

Specifically, similar to AeConv, we define an azimuth-equivariant anchor \((x, y, z, l, w, h, \alpha)\) according to the azimuth at each location. Then we could compute the box residual (i.e., prediction target) between the azimuth-equivariant anchors and the marching ground-truth boxes. The orientation residual is the relative angle of the object orientation \(w: r: t\) the azimuth:

\[
\nabla \theta = \hat{\theta} - \alpha, \tag{3}
\]

where \(\hat{\theta}\) denotes the object orientation. In contrast to the existing methods [11, 15] that predict the center offset and velocity along the radial direction and its orthogonal direction. The new center offset \((\nabla r, \nabla o)\) and the velocity \((v_r, v_o)\) can be computed by:

\[
\begin{align*}
\begin{bmatrix}
\nabla r \\
\nabla o
\end{bmatrix} &= \begin{bmatrix}
\cos(\alpha) & \sin(\alpha) \\
-\sin(\alpha) & \cos(\alpha)
\end{bmatrix} \begin{bmatrix}
\nabla x \\
\nabla y
\end{bmatrix}, \tag{4}
\end{align*}
\]

\[
\begin{align*}
\begin{bmatrix}
v_r \\
v_o
\end{bmatrix} &= \begin{bmatrix}
\cos(\alpha) & \sin(\alpha) \\
-\sin(\alpha) & \cos(\alpha)
\end{bmatrix} \begin{bmatrix}
v_x \\
v_y
\end{bmatrix}. \tag{5}
\end{align*}
\]

Note that the orientation of the azimuth-equivariant anchor is always in the radial direction of the camera. Therefore, the detection head could predict the orientation, center offset and velocity of the object based on the azimuth-equivariant direction, yielding the azimuth-irrelevant prediction targets, as illustrated in Figure 3b.

### 3.4. Camera-decoupled Virtual Depth

The depth plays an important role in rendering a precise BEV feature. Recent BEVDepth improves the depth network by introducing the camera’s intrinsic parameters into the features with a camera-aware attention module. However, naively introducing the camera’s intrinsic parameters into the features would require the depth network to model the relationship among the image feature, the focal length of the cameras and the depth, and thus increase the difficulty of the depth prediction. To reduce the burden of the depth network, we decouple the camera from the depth network. The depth network can now focus on modeling the relationship between image feature, depth score sampling, and the velocity/center offset along the consistent directions.

The proposed method contains two steps: (1) we first leverage a camera-decoupled depth network to predict a virtual depth based on a virtual focal length; (2) we then map the virtual depth to the real depth according to the focal length of the camera. We elaborate on the two steps in the following:

**Virtual depth prediction.** We leverage a camera-decoupled depth network to predict a virtual depth that is independent of the intrinsic parameters of the camera. For simplification, we use the same structure of the depth network as BEVDepth, but remove its camera-aware attention.
Table 1. Ablation study of the components in AeDet. AeNet denotes AeConv and the azimuth-equivariant anchor, and CDN denotes the camera-decoupled depth network.

| Method | mAP↑ | mAOE↓ | mATE↓ | mASE↓ | mAWE↓ | mAEE↓ | NDS↑ |
|--------|------|-------|-------|-------|-------|-------|------|
| BEVDepth | 0.315 | 0.621 | 1.042 | 0.367 |       |       | 0.367 |
| AeDet   | 0.331 | 0.563 | 0.895 | 0.397 |       |       | 0.397 |

Table 2. Comparison between BEVDepth and AeDet in the single-frame scheme.

| M     | mAP↑ | mAOE↓ | mATE↓ | NDS↑ |
|-------|------|-------|-------|------|
| 150   | 0.353 | 0.665 | 0.465 |      |
| 180   | 0.358 | 0.655 | 0.473 |      |
| 210   | 0.354 | 0.654 | 0.469 |      |

Table 3. Analysis of different numbers of virtual depth bins.

4. Experiment

Dataset & Evaluation Metrics. All experiments are implemented on the nuScenes benchmark [1], which includes 1000 scenes with images from six cameras. Following the standard practice [3,11,15], we split the dataset into 700, 150, and 150 scenes for training, validation and testing. The detection performance is measured by the official evaluation metrics: mean Average Precision (mAP), mean Average Translation Error (mATE), mean Average Scale Error (mASE), mean Average Orientation Error (mAWE), mean Average Velocity Error(mAVE), mean Average Attribute Error (mAEE) and nuScenes Detection Score (NDS).

Implementation details. We use the most recent multi-view 3D object detector BEVDepth [15] as our baseline. We conduct the experiment with different image backbones including ResNet-50 [10], ResNet-101 [10] and ConvNeXt-B [19]. The deconvolution is replaced with the up-sampling layer and convolution in the BEV network. Unless specified, we train the model with a total batchsize of 64 for 24 epochs, where AdamW [20] is adopted as the optimizer with the learning rate of 2e-4 and the EMA technique. Following BEVDepth [15], both image and BEV data augmentations are used during training, and \( \frac{d_{f1}}{d_{f2}} \) and \( \nabla d_f \) are set to 2m, 54m and 0.5m. The mapped real depth range must include the fixed depth range \([2, 54]\)m. To achieve that, we set \( d_v \) and \( f_v \) to 54m (i.e. \( d_{f2} \)) and 800 pixels (i.e. less than the minimum focal length of the nuScenes cameras).

4.1. Ablation Study

For an ablation study, we use the ResNet-50 backbone with the input of 2 key frames and 256 × 704 resolution unless specified. The performances are reported on the val set.
Azimuth-equivariant network. To demonstrate the effectiveness of both AeConv and the azimuth-equivariant anchor, we apply them to BEVDepth. As shown in Table 1, AeNet (AeConv + the azimuth-equivariant anchor) improves the NDS from 44.2% to 46.2%, with the same number of parameters and only +1.7% FLOPs. To be more specific, the improvements in mAOE and mAVE are significant: mAOE and mAVE are improved by 3.6% and 4.8%, respectively. This validates the effectiveness of the design of AeConv and the azimuth-equivariant anchor, and demonstrates that they can work efficiently to obtain high performance, especially for the object orientation and velocity.

Camera-decoupled virtual depth. Here, we conduct the ablation study on the proposed camera-decoupled virtual depth. We first use the camera-decoupled depth network to predict the virtual depth, and then compute the real depth through the depth mapping. As shown in Table 1, the camera-decoupled depth network improves the mAP and NDS by 1.5% and 1.2%, comparing to BEVDepth. Specifically, the most improvement is in mATE (69.9% → 68.2%). This shows that the proposed virtual depth and depth mapping enhance the accuracy of the depth prediction, by unifying the depth prediction of the multi-camera images.

AeDet. We evaluate the performance of the complete AeDet (i.e. AeNet + CDN). As shown in Table 1, AeDet achieves 35.8% mAP and 47.3% NDS, surpassing BEVDepth by 2.8% mAP and 3.1% NDS. To be more specific, the mAOE and mAVE are improved by 5.2% and 6.6%. We also evaluate the performance of the single-frame AeDet, as shown in Table 2. Compared with BEVDepth, AeDet brings an improvement of 3.0% NDS in single-frame scheme. Note that estimating the velocity of the object is difficult in the single-view scheme. By unifying the prediction of velocity, AeDet improves the mAVE by 14.7%.

The number of virtual depth bins. We investigate the performance using different $M$ which denotes the number of the discretized virtual depth bins. Through a coarse search shown in Table 3, we find that the performance is insensitive to $M$. We adopt $M = 180$ for our experiments.

4.2. Revolving Test

Robustness to different azimuths. The detection robustness to different azimuths is important to the autonomous driving system, since sometimes the vehicle may turn at a large angle. For example, at small roundabouts or corner roads, the turning angle of the vehicle becomes large, resulting in a great change in the camera orientation. The autonomous vehicle should be able to accurately detect the surrounding objects even in such situations. To verify the robustness of the detector, we propose a revolving test to simulate this scenario: we turn the vehicle 60 degrees clockwise, namely, the original views/images ['FRONT LEFT', 'FRONT', 'FRONT RIGHT', 'BACK RIGHT', 'BACK', 'BACK LEFT'] become the revolved ones ['BACK LEFT', 'FRONT LEFT', 'FRONT', 'FRONT RIGHT', 'BACK RIGHT', 'BACK'], as illustrated in Figure 6. Then we evaluate the detector in the revolved view.
In this work, we propose an azimuth-equivariant detector AeDet, which is able to perform azimuth-invariant 3D object detection. Specifically, we first rotate the sampling grid of the convolution along the azimuth to extract the azimuth-invariant features. Then we rotate the anchor to allow the detection head to predict the azimuth-irrelevant targets. In contrast, AeDet achieves a similar performance (47.3% NDS) in both views. This validates the robustness of AeDet with respect to the changes in azimuths, thanks to the azimuth-invariant feature learning and the azimuth-irrelevant target prediction. The predictions from the two views are displayed in Figure 6. AeDet yields almost the same prediction in both views.

### 4.3. Comparison with the State-of-the-Art

We compare our AeDet with other multi-view 3D object detectors on both the nuScenes val set and test set. The models are trained with CBGS strategy [29] for 20 epochs. The performance on the nuScenes val set is shown in Table 5. For a fair comparison, we report the results at single-model single-scale testing. With ResNet-50 and ResNet-101, AeDet achieves 50.1% NDS and 56.1% NDS, outperforming the most current multi-view 3D object detectors such as BEVFormer [16] (by 4.4% NDS) and BEVDepth [15] (by 2.6% NDS). For the submitted results on the test set, we use the train set and val set for training. We adopt ConvNeXt-B as the image backbone, and report the results of single model with test-time augmentation. As Table 6 shows, AeDet sets a new state-of-the-art result with 53.1% mAP and 62.0% NDS in multi-view 3D object detection.

### 5. Conclusion

In this work, we propose an azimuth-equivalent detector AeDet, which is able to perform azimuth-invariant 3D object detection. Specifically, we first rotate the sampling grid of the convolution along the azimuth to extract the azimuth-invariant features. Then we rotate the anchor to allow the detection head to predict the azimuth-irrelevant targets. In addition, we further introduce a camera-decoupled virtual depth to unify the depth prediction in both views of different cameras. With these improvements, AeDet sets a new state-of-the-art result with 62.0% NDS on nuScenes.

---

**Table 5. Comparison on the nuScenes val set.**

| Method       | Backbone  | Image size | mAP↑ | mATE↓ | mASE↓ | mAOE↓ | mAVE↓ | mAAE↓ | NDS↑ |
|--------------|-----------|------------|------|-------|-------|-------|-------|-------|------|
| DETR3D [27]  | V2-99     | 900 × 1600 | 0.412| 0.641 | 0.255 | 0.394 | 0.845 | 0.133 | 0.479|
| UVTR [14]    | V2-99     | 900 × 1600 | 0.472| 0.577 | 0.253 | 0.391 | 0.508 | 0.123 | 0.551|
| BEVFormer    | V2-99     | 900 × 1600 | 0.481| 0.582 | 0.256 | 0.375 | 0.378 | 0.126 | 0.569|
| BEVDepth4D   | Swim-B    | 900 × 1600 | 0.451| 0.511 | 0.241 | 0.386 | 0.301 | 0.121 | 0.569|
| PolarFormer  | V2-99     | 900 × 1600 | 0.493| 0.556 | 0.256 | 0.364 | 0.439 | 0.127 | 0.572|
| PETRv2 [18]  | V2-99     | 640 × 1600 | 0.490| 0.561 | 0.243 | 0.361 | 0.343 | 0.120 | 0.582|
| BEVDepth15   | V2-99     | 640 × 1600 | 0.503| 0.445 | 0.245 | 0.378 | 0.320 | 0.126 | 0.600|
| AeDet (ours) | ConvNeXt-B| 640 × 1600 | 0.520| 0.445 | 0.243 | 0.352 | 0.347 | 0.127 | 0.609|

**Table 6. Comparison on the nuScenes test set.** † denotes data augmentation that randomly samples time intervals in previous frames.

| Method       | Backbone  | Image size | mAP↑ | mATE↓ | mASE↓ | mAOE↓ | mAVE↓ | mAAE↓ | NDS↑ |
|--------------|-----------|------------|------|-------|-------|-------|-------|-------|------|
| DETR3D [27]  | V2-99     | 900 × 1600 | 0.531| 0.439 | 0.247 | 0.344 | 0.292 | 0.130 | 0.620|
| UVTR [14]    | V2-99     | 900 × 1600 | 0.462| 0.540 | 0.254 | 0.353 | 0.379 | 0.200 | 0.558|
| AEDETR-T [4] | ResNet101-DCN | 900 × 1600 | 0.295| 0.806 | 0.268 | 0.511 | 1.131 | 0.170 | 0.372|
| UVTR [14]    | ResNet101-DCN | 900 × 1600 | 0.383| 0.707 | 0.269 | 0.344 | 0.518 | 0.196 | 0.488|
| PETRv2 [18]  | ResNet101-DCN | 900 × 1600 | 0.379| 0.731 | 0.267 | 0.350 | 0.510 | 0.200 | 0.483|
| PETRv2 [18]  | ResNet101-DCN | 900 × 1600 | 0.432| 0.648 | 0.270 | 0.348 | 0.409 | 0.201 | 0.528|
| BEVFormer    | ResNet101-DCN | 900 × 1600 | 0.416| 0.673 | 0.274 | 0.372 | 0.394 | 0.198 | 0.517|
| PETR [17]    | ResNet101 | 512 × 1408 | 0.412| 0.565 | 0.266 | 0.358 | 0.331 | 0.190 | 0.535|
| BEVDepth15   | ResNet101 | 512 × 1408 | 0.449| 0.501 | 0.262 | 0.347 | 0.330 | 0.194 | 0.561|
| AEDETR (ours) | ConvNeXt-B | 512 × 1408 | 0.483| 0.494 | 0.261 | 0.324 | 0.337 | 0.195 | 0.581|

---
References

[1] 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. In CVPR, pages 11621–11631, 2020. 2, 6

[2] Yingjie Cai, Buyu Li, Zeyu Jiao, Hongsheng Li, Xingyu Zeng, and Xiaogang Wang. Monocular 3d object detection with decoupled structured polygon estimation and height-guided depth estimation. In AAAI, volume 34, pages 10478–10485, 2020. 2

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In ECCV, pages 213–229. Springer, 2020. 2, 6

[4] Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Chang Huang, and Wenyu Liu. Polar parameterization for vision-based surround-view 3d detection. arXiv preprint arXiv:2206.10965, 2022. 3, 8

[5] Mingyu Ding, Yuqi Huo, Hongwei Yi, Zhe Wang, Jianping Shi, Zhiwu Lu, and Ping Luo. Learning depth-guided convolutions for monocular 3d object detection. In CVPR Workshops, pages 1000–1001, 2020. 2

[6] Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R Scott, and Weilin Huang. Tood: Task-aligned one-stage object detection. In ICCV, pages 3490–3499. IEEE Computer Society, 2021. 1

[7] Chengjian Feng, Yujie Zhong, and Weilin Huang. Exploring classification equilibrium in long-tailed object detection. In ICCV, pages 3417–3426, 2021. 1

[8] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In CVPR, pages 2002–2011, 2018. 3

[9] Ravi Garg, Vijay Kumar Bg, Gustavo Carneiro, and Ian Reid. Unifying voxel-based representation with transformer for 3d object detection. arXiv preprint arXiv:2206.00630, 2022. 8

[10] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yueli Lu, and Philipp Krahenbuhl. Center-guided depth estimation. In ECCV, pages 194–210. Springer, 2020. 2

[11] Ashutosh Saxena, Sung Chung, and Andrew Ng. Learning depth from single monocular images. NeurIPS, 18, 2005. 3

[12] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In ICCV, pages 9627–9636, 2019. 2

[13] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In ICCV, pages 913–922, 2021. 2, 8

[14] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In CVPR, pages 8445–8453, 2019. 2

[15] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In Conference on Robot Learning, pages 180–191. PMLR, 2022. 2, 8

[16] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In CVPR, pages 11784–11793, 2021. 1, 2, 3

[17] Benjin Zhu, Zhengkai Jiang, Xiaogang Wang, and Zeming Li. Class-balanced grouping and sampling for point cloud 3d object detection. arXiv preprint arXiv:1908.09492, 2019. 8