

Monocular 3D Object Detection with Bounding Box Denoising in 3D by Perceiver

Xianpeng Liu^{1*}, Ce Zheng², Kelvin Cheng¹, Nan Xue³, Guo-Jun Qi^{4,5}, Tianfu Wu¹

¹North Carolina State University ²University of Central Florida ³Ant Group

⁴OPPO Seattle Research Center, USA ⁵Westlake University

{xliu59, kbcheng, tianfu_wu}@ncsu.edu; cezheng@knights.ucf.edu;

xuenan@ieee.org; guojunq@gmail.com

<https://xianpeng919.github.io/monoxiver>

Abstract

The main challenge of monocular 3D object detection is the accurate localization of 3D center. Motivated by a new and strong observation that this challenge can be remedied by a 3D-space local-grid search scheme in an ideal case, we propose a stage-wise approach, which combines the information flow from 2D-to-3D (3D bounding box proposal generation with a single 2D image) and 3D-to-2D (proposal verification by denoising with 3D-to-2D contexts) in a top-down manner. Specifically, we first obtain initial proposals from off-the-shelf backbone monocular 3D detectors. Then, we generate a 3D anchor space by local-grid sampling from the initial proposals. Finally, we perform 3D bounding box denoising at the 3D-to-2D proposal verification stage. To effectively learn discriminative features for denoising highly overlapped proposals, this paper presents a method of using the Perceiver I/O model [20] to fuse the 3D-to-2D geometric information and the 2D appearance information. With the encoded latent representation of a proposal, the verification head is implemented with a self-attention module. Our method, named as **MonoXiver**, is generic and can be easily adapted to any backbone monocular 3D detectors. Experimental results on the well-established KITTI dataset and the challenging large-scale Waymo dataset show that MonoXiver consistently achieves improvement with limited computation overhead.

1. Introduction

Detecting and locating objects in 3D using a single image, known as monocular 3D object detection, is a highly challenging task due to its ill-posed nature. However, the practical applications of low-cost system setups in fields such as self-driving cars and robotic manipulation have led

*Work partially conducted during an internship of X. Liu at OPPO Seattle Research Center, USA in 2022.

to the development of robust monocular 3D object detection systems, making it a prominent research topic in the computer vision community.

Although significant progress has been achieved in this field [37], accurate 3D center localization remains a major challenge for state-of-the-art (SOTA) methods as indicated in [39]. Most of these methods follow a bottom-up paradigm, where a single 2D image is used to directly predict 3D bounding boxes (2D-to-3D) with or without extra information (e.g. LiDAR). However, due to the inherent depth ambiguity of this task, relying solely on bottom-up 2D-to-3D paradigms may not be enough to completely address this challenge.

In this paper, we observe that bottom-up 2D-to-3D predicted 3D bounding boxes are able to provide informative priors for monocular 3D object detection. We propose to leverage these contexts to improve detection performance in a top-down manner. Our proposed approach is motivated by a strong empirical upper-bound analysis with SOTA bottom-up monocular 3D object detectors.

The empirical upper-bound experiment. We analyze vehicle detection performance on the well-established KITTI validation set [5–7]. We start with the recently proposed MonoCon [30] because it achieves SOTA performance with a very simple design. By using MonoCon’s prediction results as bottom-up 3D anchors, we sample 3D bounding box proposals along both the x and z axes (i.e., in the bird-eye’s view) with a simple strategy. First, we define a 2D local grid with a range (e.g., 2 meter) and a stride (e.g., 0.1 meter). For each bottom-up anchor, with its center located at the local-grid origin, we replicate the anchor at each grid vertex without changing its size, orientation and prediction score. We use all the generated 3D bounding boxes as top-down proposals and compute the empirical upper-bound of 3D detection performance by searching for the best match within top-down proposals for each ground-truth 3D object bounding box based on 3D Intersection-over-Union (3D IoU).

A strong observation from the experiment. As shown

Range	Stride	Easy	Moderate	Hard	#Bboxes	mean IoU _{avg} ⁵
MonoCon [30]		26.33	19.01	15.98	n	-
± 2	0.1	76.98 ^{↑50.65}	64.93 ^{↑45.92}	56.91 ^{↑40.93}	1600-n	0.931
± 1.5	0.1	76.45 ^{↑50.12}	62.27 ^{↑43.26}	54.34 ^{↑38.36}	900-n	0.931
± 1.5	0.2	74.00 ^{↑47.67}	61.78 ^{↑42.77}	53.64 ^{↑37.66}	225-n	0.868
± 1.5	0.3	64.80 ^{↑38.47}	56.20 ^{↑37.19}	50.66 ^{↑34.68}	121-n	0.812
± 1.5	0.5	54.00 ^{↑27.67}	45.93 ^{↑26.92}	40.99 ^{↑25.01}	49-n	0.714
± 1.5	0.75	41.58 ^{↑15.25}	34.44 ^{↑15.43}	30.12 ^{↑14.14}	25-n	0.612

Table 1: A strong empirical upper-bound analysis. Our MonoXiver is motivated by the observation that bottom-up monocular 3D object detectors can be significantly improved by leveraging a simple 3D-space local-grid search scheme in an ideal case. This highlights the potential of exploring the 3D proposal space. However, this improvement comes at a cost: 1) the proposal verification stage experiences significant increases in number of bounding box proposals, and 2) a more powerful refinement module is required to handle highly overlapped boxes.

in Table 1, despite the evaluation result of initial bottom-up proposals is relatively low in 3D average precision (AP_{3D}) (e.g., 19.01 in Moderate settings), *most of predicted 3D centers are already in the close proximity of the GT ones. An empirical upper-bound of 34.4 AP_{3D} can be achieved even with a coarse sampling scheme* (e.g., the last row), which is significantly higher than SOTA methods. We also obtained similar observations from other backbone 3D object detectors such as the MonodLE [39] and the SMOKE [33]. These strong empirical observations demonstrate the potential of integrating the bottom-up initial proposals with top-down sampling and verification.

The challenge in the 3D-to-2D proposal verification.

The 3D-to-2D proposal verification phase can be treated as a 3D bounding box denoising process, as we want to search for the “best” bounding boxes from the top-down proposal set. This process is extremely challenging, as the proposals generated from the same bottom-up anchor are highly overlapped in both 3D and 2D (after projection), which leads to the long-standing problem of handling the “crowd” in detection. To quantitatively analyze the overlap extent after projection for top-down proposals, we consider one proposal and its top- k overlapping neighbors in terms of Intersection-over-Union (IoU), which is denoted by IoU_{avg}^k as the average IoU over the top- k neighbors. The last column in Table 1 shows the statistics.

The statistics clearly demonstrates the difficulty of denoising densely generated top-down proposal set (e.g., stride=0.1, $\text{IoU}_{avg}^5=0.931$). Even for a relative sparse generated top-down proposal set (e.g., stride=0.75, $\text{IoU}_{avg}^5=0.612$), the challenge still exists because proposals sampled in front and behind of the same anchor will almost collapse to the same 2D bounding box, especially when they are far away from the camera. So, how can we

encode the 3D bounding box proposal for verification under the monocular setting? From this perspective, we note that many methods that work well in multi-view 3D object detection are often not applicable for monocular 3D objection because they mainly rely on features that are obtained by fusing projection features from multi-view feature maps.

Based on the intuition that even though highly overlapped proposals have similar appearance features, their inherent 3D-to-2D geometric features (e.g. 3D location, projected geometry, etc.) are extremely different, we propose to fuse these 3D-to-2D geometric feature with their correspondent appearance features to learn discriminative features for bounding box denoising. We present a method of using the Perceiver I/O model [20] to effectively fuse these contexts, as Perceiver has shown strong capabilities to fuse multi-modal inputs. With the encoded latent representation of a proposal, the verification head is implemented by a self-attention module (see the top in Fig. 1). The proposed method is named as **MonoXiver**, indicating its general applicability to any backbone monocular 3D detectors and the integration of the Perceiver model.

In experiments, we evaluate our proposed MonoXiver on the well-established KITTI benchmark [14] and the challenging large-scale Waymo [51] dataset with various backbone monocular 3D detectors. It achieves consistent and significant performance improvement on both datasets with limited computation overhead. Moreover, it achieves the 1st place among monocular methods on the KITTI vehicle detection benchmark, outperforming the previous works by a large margin.

2. Related Works and Our Contributions

Monocular 3D Object Detection. The problem of monocular 3D object detection is considered ill-posed, prompting recent research efforts to leverage additional information sources, including LiDAR point clouds [4, 10, 19, 28, 45], depth estimation [12, 36, 42, 48, 53, 54, 62], CAD models [25, 34], temporal frames [2], etc. These approaches have shown improved detection performance when compared to purely image-based methods [1, 33, 41, 57]. However, they often come with increased computation load and inference time, making them less practical for real-time applications such as autonomous vehicles. In contrast, purely image-based methods have seen significant performance improvements in the literature by exploiting geometric constraints [26, 35, 41, 47, 59, 60], perspective projections [24, 38], auxiliary training [30], novel loss designs [8, 39, 49] and second-stage processing techniques [23]. These works focus on developing better bottom-up paradigms. Our proposed method, MonoXiver, takes a top-down paradigm, which explores 3D space differently than the aforementioned methods. It provides a generic and highly efficient second-stage refinement module for any pretrained state-of-

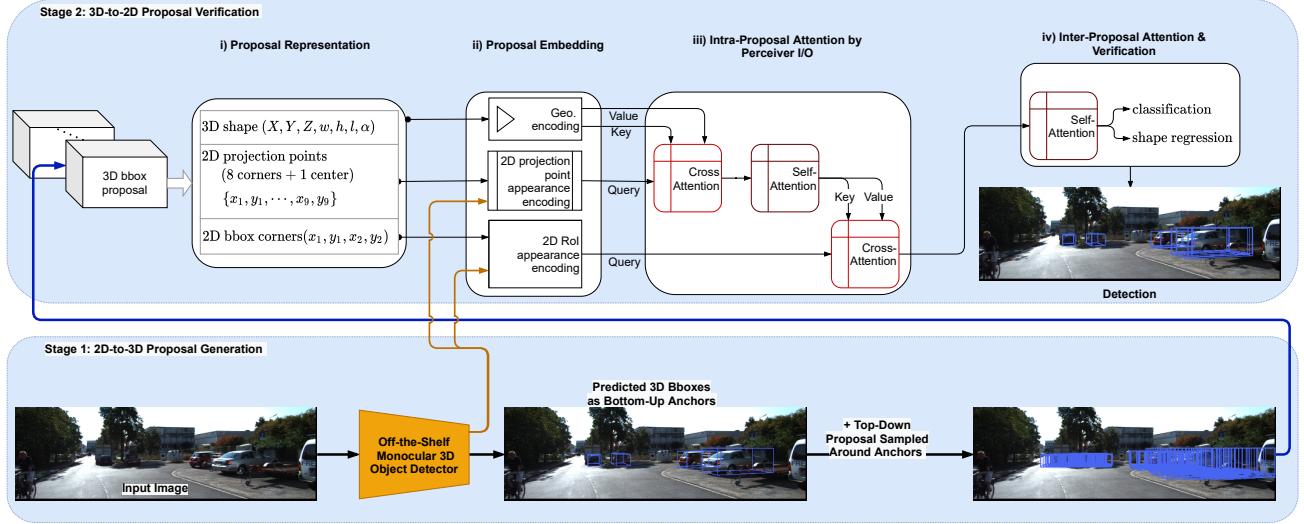


Figure 1: Illustration of the proposed MonoXiver method. MonoXiver is built for any off-the-shelf backbone monocular 3D object detectors. It consists of a 2D-to-3D proposal generation phase using a bottom-up anchoring and top-down sampling strategy, and a 3D-to-2D proposal verification (or denoising) phase using the Perceiver I/O [20] with a carefully designed 3D/2D input space to address the unique challenges. See text for details.

the-art monocular 3D object detection method, making it a complementary approach.

Bird’s-Eye-View 3D Object Detection. Bird’s-Eye-View (BEV) 3D object detection [40] has recently gained significant attention as it provides a physical-interpretable feature space for integrating data from diverse sensors [27, 31, 44, 56] and modalities [13, 18]. To enable the learning of a physical-interpretable feature space, accurate physical measurements from depth perception systems such as multi-view cameras and LiDAR are crucial [16]. However, detecting 3D objects in BEV is challenging for monocular systems due to the intrinsic depth ambiguity. Directly accumulating image features based on projection reference points similar to recent query-based multi-view 3D detectors [31, 56] might fail because of lacking stereo-matching mechanism to resolve the depth ambiguity. To overcome this issue, recent BEV-based monocular methods [9, 45, 55] lift the image feature space to voxel-represented world-space using monocular depth estimation models. Our proposed method, MonoXiver, samples top-down proposals based on bottom-up initial proposals in BEV space. Unlike prior arts [9, 45, 55] that extract features represented in world-space, MonoXiver’s top-down proposals extract features from both projected appearance features and inherent geometric features. We explore to address the challenging intrinsic feature ambiguity problem by applying a Perceiver-like attention mechanism to query informative features for 3D bounding box verification and refinement without the need for dense depth supervision.

Transformers for Multi-modal Data. Transformers [52],

known for their powerful attention mechanism, have been successfully applied in multiple research areas and applications [3, 11, 32, 50]. With the powerful attention mechanism, transformers can conveniently align multi-modal structured data, enabling the generation of universal multi-modal representation. The recent proposed Perceiver [20, 21] extends the transformers’ querying mechanism and makes it capable of efficiently handling data from arbitrary settings. In our work, we adapt Perceiver to fuse appearance features and geometric features from the same bounding box to generate appearance-geometric aligned representation, which is then used as queries for the down-stream 3D bounding box refinement task.

Our Contributions. MonoXiver makes three main contributions to the field of monocular 3D object detection:

- We conduct a new and strong empirical upper-bound analysis for state-of-the-art monocular 3D object detectors. We demonstrate that a simple, universally applicable sampling strategy can lead to consistent and significant performance improvement in an ideal situation. Based on these observations, we propose to explore the top-down, stage-wise detection paradigm with 2D-to-3D proposal generation and 3D-to-2D proposal verification for monocular 3D object detection.
- To address the challenge at the bounding box denoising stage, we propose to learn discriminative features by enhancing appearance features with 3D-to-2D inherent geometric features. We therefore propose a carefully-designed module with the integration of the powerful Perceiver and self-attention mechanism. The resulting mod-

ule can be applied generically to any off-the-shelf monocular 3D object detectors.

- Extensive experiments demonstrate the effectiveness and efficiency of our proposed method, MonoXiver, under a separate two-stage training setting. It achieves consistent improvement with limited overhead on both KITTI and Waymo dataset. It achieves the 1st place among monocular methods on the KITTI vehicle detection benchmark, outperforming the previous works by a large margin.

3. Approach

In this section, we first present the straightforward 2D-to-3D proposal generation on top of a backbone monocular 3D object detector. Then, we present the details of the proposed MonoXiver method (Fig. 1).

3.1. The 2D-to-3D Proposal Generation

Let I be an RGB input image defined on the domain Λ . The goal of monocular 3D object detection is to detect 3D bounding boxes, along with their class labels denoted by ℓ for each object instance in I . The 3D bounding box is parameterized by the 3D center position $\mathbf{P} = (X, Y, Z)$ in meters, the shape dimensions $\mathbf{D} = (h, w, l)$ in meters, and the observation angle $\alpha \in [-\pi, \pi]$, all measured in the camera coordinate system.

For simplicity, consider the pure image-based settings, a monocular 3D object detector often consists of two main components in inference: the feature backbone (e.g., the DLA34 network [58]), denoted by \mathbf{F}_I for computing deep features from the raw image I , and the regression heads for inferring the 3D bounding box parameters using the computed feature map. We generate bottom-up proposals using one off-the-shelf monocular 3D object detector which is first trained following its own training receipts. Without loss of generality, we consider one of its detected 3D bounding boxes indexed by i in an image I ,

$$\mathbb{B}_i = (\ell_i, \mathbf{P}_i, \mathbf{D}_i, \alpha_i), \quad (1)$$

As shown in Table 1, the top-down sampling with a bottom-up anchor proposal is a straightforward process. We start by generating a 2D local grid in the X - Z plane (i.e. in the bird’s-eye-view, BEV), centered at the position (X, Z) of the anchor box \mathbb{B} . To accomplish this, we specify a search range and a stride. Based on the trade-off between the empirical upper bound and the computing overhead and the mean IoU among the proposals, we adopt a conservative strategy using the search range ± 1.5 meters and the stride 0.75, which will generate 25 proposals per bottom-up anchor (inclusive). We then place the anchor box \mathbb{B} at the 25 grid points, with only the position \mathbf{P} updated. To simplify the notation, we do not differentiate between the bottom-up anchor and the top-down sampled proposal, and we generally index them using i , unless otherwise stated.

3.2. The 3D-to-2D Proposal Verification

The proposal verification will rely essentially on the information from the input 2D images in purely monocular 3D object detection. Due to the aforementioned unique challenges in the 3D proposal space, we utilize Perceiver I/O [20] to design an expressive proposal representation learning scheme.

3.2.1 Proposal Representation

As illustrated in Fig. 1, we utilize three types of features in encoding a 3D bounding box proposal \mathbb{B} ,

- The 3D Shape Features* represented by $(\mathbf{P}, \mathbf{D}, \alpha)$. It is a 7-dim vector. It encodes where the proposal is in the 3D space (the camera coordinate system), as well as its 3D occupancy. The position \mathbf{P} is empirically normalized by the typical detection range (e.g., X, Y, Z are normalized by 50, 2 and 80 meters respectively).
- The 2D Projection Points* from the 8 corners and the center of the 3D bounding box (with known camera intrinsic matrix). It is an 18-dim vector in the image coordinate system, and normalized by the image size. This 9-point projection provides the finegrained placement of a proposal in the 2D space, that is the geometric bond between a proposal in the 3D space and its 2D placement.
- The 2D Bounding Box* projected from the proposal and truncated by the image boundary. We use the left-top and right-bottom two points to encode a 2D bounding box. It gives a 4-dim vector normalized by the image size. For a proposal whose 2D projection is entirely inside the image plane, the two points of the 2D bounding box will be just redundant with respect to the 9 project points. Otherwise, the truncation points on the image boundary will facilitate the learning to be truncation aware.
- The Appearance Features* of the 9 projection points. We directly extract features from the final layer of the feature backbone \mathbf{F}_I . If a projection point is out of the image plane, we encode it using an all-zero vector.
- The ROI Appearance Features* of the 2D bounding box. We use the RoIAlign features [17] extracted from the final layer of the feature backbone \mathbf{F}_I . Since the 2D IoU overlapping between proposals is high on average (e.g., 0.612 in Table 1), we use a finer grid, 14×14 , in computing the RoIAlign features, such that the extracted appearance features contain “sufficient” details.

3.2.2 Proposal Embedding

Let N be the total number of proposals in an image I . With the above representation scheme, denote by $f_{N \times 29}^{geo}$ the geometric parameter matrix from the first three items above,

and by $f_{N \times 9 \times C}^{pt}$ the projection point appearance feature matrix (where C is the number of channels of the final layer of the feature backbone, e.g., $C = 64$), and by $f_{N \times (14 \times 14) \times C}^{roi}$ the RoI feature matrix. Before applying the Perceiver I/O model, we embed the three to form “tokens” using multi-layer perceptron (MLP).

For the geometric parameters $f_{N \times 29}^{geo}$, we have,

$$f_{N \times 29}^{geo} \xrightarrow{\text{MLP}} z_{N \times (g \times d)} \xrightarrow{\text{Rearrange}} \mathbf{f}_{g \times N \times d}^{geo}, \quad (2)$$

where d is the hidden dimension (e.g., $d = 256$). g is the group number, similar in spirit to the multi-head setup in the self-attention. We want to encode the geometric parameters in different latent spaces to account for the large variations in the proposal space (e.g., $g = 4$).

For the project point features $f_{N \times 9 \times C}^{pt}$, we have,

$$f_{N \times 9 \times C}^{pt} \xrightarrow{\text{MLP}} z_{N \times 9 \times d} \xrightarrow{\text{Rearrange}} \mathbf{f}_{9 \times N \times d}^{pt}, \quad (3)$$

where we reuse z to denote the latent features for simplicity and without confusion.

For the RoI features $f_{N \times (14 \times 14) \times C}^{roi}$, we have,

$$f_{N \times (14 \times 14) \times C}^{roi} \xrightarrow{\text{MLP}} z_{N \times d} \xrightarrow{\text{Rearrange}} \mathbf{f}_{1 \times N \times d}^{roi}. \quad (4)$$

3.2.3 Intra-Proposal Attention via Perceiver

With the above proposal embedding, before the proposal verification, our goal is to compute a final latent feature representation in the d -dim space for each proposal by fusing information from the geometric encoding, projection-point-based appearance encoding and RoI-based appearance encoding (i.e., the intra-proposal attention), such that we can address the aforementioned unique challenges in the proposal space. We leverage the Perceiver I/O model for this intra-proposal attention.

The Perceiver first fuses the projection-point appearance encoding $\mathbf{f}_{9 \times N \times d}^{pt}$ and the geometric encoding $\mathbf{f}_{g \times N \times d}^{geo}$. As shown in Fig. 1, it consists of a cross-attention by treating the former as Query (consisting of 9 projection-point tokens) and computing Key and Value from the latter (consisting of g geometric tokens), followed by a self-attention module. We have,

$$(\mathbf{f}_{9 \times N \times d}^{pt}, \mathbf{f}_{g \times N \times d}^{geo}) \xrightarrow{\text{Cross-Attn}} \cdot \xrightarrow{\text{Self-Attn}} \mathbf{F}_{9 \times N \times d}^{geo-pt}, \quad (5)$$

which results in geometry-aware projection-point encoding.

Next, the Perceiver fuses the RoI appearance encoding $\mathbf{f}_{1 \times N \times d}^{roi}$ as the 1-token Query with the 9-token geometry-aware projection-point encoding using a cross attention module,

$$(\mathbf{f}_{1 \times N \times d}^{roi}, \mathbf{F}_{9 \times N \times d}^{geo-pt}) \xrightarrow{\text{Cross-Attn}} \mathbf{F}_{1 \times N \times d}^{geo-pt-roi}, \quad (6)$$

which results in the d -dim latent features for each proposal, which fuse the three types of information sources: geometry, point-level appearance and region-level appearance.

In the above, the cross-attention and self-attention modules are based on the standard formulation [52].

3.2.4 Inter-Proposal Attention

With all the above process, each proposal is still encoded individually with the hope of fusing geometry, point-level appearance and region-level appearance information in a semantically meaningful way via the Perceiver I/O model. With the compact latent vector computed for each proposal in $\mathbf{F}_{1 \times N \times d}^{geo-pt-roi}$, their interactions need to be taken into account in order to resolve their “crowding” issue at the underlying scene level.

To that end, we treat each proposal as a “token” and apply a standard self-attention module to capture the inter-proposal attention,

$$\mathbf{F}_{1 \times N \times d}^{geo-pt-roi} \xrightarrow{\text{Rearrange}} \mathbf{F}_{N \times d}^{geo-pt-roi} \xrightarrow{\text{Self-Attn}} \mathbb{F}_{N \times d}. \quad (7)$$

3.2.5 Proposal Verification

This verification is posed as a regression problem ,

$$\mathbb{F}_{N \times d} \xrightarrow{\text{MLPs}} (\mathbf{y}_{N \times L}, \Delta \mathbf{P}_N, \Delta \mathbf{D}_N), \quad (8)$$

where L is the total number of categories (e.g., $L = 3$ in the KITTI dataset), and $\mathbf{y}_{N \times L}$ is the classification scores (logits). $\Delta \mathbf{P}_N$ and $\Delta \mathbf{D}_N$ are the position and dimension residuals.

Consider a proposal \mathbb{B}_i (Eqn. 1), it is verified via,

$$\hat{\ell}_i = \arg \max_{l=1, \dots, L} \mathbf{y}_{i,l}, \quad (9)$$

$$\hat{\mathbf{P}}_i = \mathbf{P}_i + \Delta \mathbf{P}_i, \quad (10)$$

$$\hat{\mathbf{D}}_i = \mathbf{D}_i + \Delta \mathbf{D}_i, \quad (11)$$

where a proper unnormalization step will be done for $\hat{\mathbf{P}}_i$ to counter the normalization step used in the proposal representation (Sec. 3.2.1). Based on the score $\mathbf{y}_{i, \hat{\ell}_i}$, we can keep top- k proposals for each anchor position.

3.3. Details of Training and Testing

For simplicity, we evaluate the proposed MonoXiver in a two-stage setting where the backbone monocular 3D object detector is first trained and kept frozen. Then, the MonoXiver component is trained end-to-end. One of the reasons for this choice is that off-the-shelf backbone monocular 3D object detectors often involve multiple loss functions, and joint training with MonoXiver would require sophisticated tuning of the trade-off parameters for different loss terms. We leave the joint end-to-end training or iterative training,

as done in the early version of Faster RCNN [46] between the region proposal network and the region classification head network, for future work. Additionally, the separate two-stage training may also have advantages in leveraging the set of bottom-up anchor proposals merged from multiple backbone monocular 3D detectors, which we also leave for future work.

3.3.1 The Set Prediction Formulation in Training

Based on the separate two-stage setting, the proposed MonoXiver is trained with a fixed set of 3D bounding box proposals and a fixed set of ground-truth 3D bounding boxes. The ground-truth assignment for proposals is needed in training. Two straightforward methods are: using the maximum 3D IoU based assignment, or using the maximum 2D IoU assignment (after projected to the image plane). Due to the “crowding” issue in the proposal space, we observe that both of them do not work during our development of the MonoXiver since they will create difficult “decision boundaries” for the MonoXiver to learn or fit.

Since we have the two fixed set as input, we resort to the set prediction formulation used in the DETR framework [3]. Denote by $\Omega_{\mathbb{B}} = \{\mathbb{B}_i\}_{i=1}^N$ the set of generated proposals, by $\Omega_{\mathbb{B}^*} = \{\mathbb{B}_j^*\}_{j=1}^M \cup \{\emptyset_j\}_{j=M+1}^N$ the set of ground-truth 3D bounding boxes padded with $N - M$ dummy \emptyset elements. Given a permutation of N elements, denoted by σ which assigns the i -th element in the ground-truth set to the $\sigma(i)$ -the element in the proposal set, the loss for the one-to-one bipartite matching between \mathbb{B}_i^* and $\mathbb{B}_{\sigma(i)}$ is defined by,

$$\begin{aligned} \mathcal{L}_{match}(\mathbb{B}_i^*, \mathbb{B}_{\sigma(i)}) &= \mathbf{1}_{\ell_i^* \neq \emptyset} \cdot (-\lambda_1 \cdot \hat{p}_{\sigma(i)}(\ell_i^*) + \\ &\quad \lambda_2 \cdot \mathcal{L}_{bbox}^{2D} + \lambda_3 \cdot \mathcal{L}_{iou}^{2D} + \lambda_4 \cdot \mathcal{L}_{iou}^{3D}), \end{aligned} \quad (12)$$

where $\hat{p}_{\sigma(i)}(\ell_i^*)$ is computed using $\mathbf{y}_{\sigma(i)}$ (Eqn. 8) via Softmax. \mathcal{L}_{bbox}^{2D} represents the normalized L1 2D bounding box, and \mathcal{L}_{iou}^{2D} and \mathcal{L}_{iou}^{3D} the IoU loss in 2D and 3D respectively. λ_1 to λ_4 are the trade-off parameters.

3.3.2 Loss Functions for Proposal Verification

For the three outputs in Eqn. 8, we have three loss terms as follows,

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{size} + \lambda \cdot \mathcal{L}_{loc} \quad (13)$$

where \mathcal{L}_{cls} is the Focal Loss [29] with the α value of 0.5 to balance the number of positive and negative samples. The Focal loss is used due to the imbalance introduced by the explicit top-down proposal generation. \mathcal{L}_{size} are \mathcal{L}_{loc} are the loss functions for the 3D size and the 3D center respectively. λ is an trade-off hyper-parameter. We use ℓ_1 loss for \mathcal{L}_{size} and \mathcal{L}_{loc} . We set $\lambda = 5$ to induce the model to focus more on the 3D localization refinement task.

3.3.3 Training

MonoXiver is trained using a total of 8 GPUs, with a batch size of 64 for 24 epochs on KITTI (12 epochs on Waymo) using the AdamW optimizer. The optimizer is set with $(\beta_1, \beta_2) = (0.95, 0.99)$ and weight decay of 0.0001, excluding feature normalization layers and bias parameters. The initial learning rate is set to $2.25e-5$, and it is reduced by a factor of 10 at the 16th and 22nd epoch. Notably, the entire second-stage training process takes merely 1.5 hours on KITTI [14] using a single Nvidia RTX5000 GPU, indicating the low computational overhead of our proposed MonoXiver method. The training time on Waymo [51] varies depending on the training data and training recipes of off-the-shelf monocular 3D detectors. Overall, the second-stage training on Waymo is much faster compared to training the backbone monocular 3D object detector, leading to about a 3/4 reduction in training time.

3.3.4 Testing

During testing, we keep at most top-3 predictions per initial anchor proposal with a score threshold of 0.03 (the score is predicted from the proposed MonoXiver module). The final prediction score is the product of the classification probability predicted from the backbone monocular 3D object detector and the score computed by the MonoXiver module.

4. Experiments

We evaluate our MonoXiver on the well-established KITTI [14] dataset and the large-scale Waymo [51] dataset. We first show that our improved baseline detector outperforms previous SOTA methods on the KITTI benchmark by a large margin. Then we show that we consistently improved pretrained monocular 3D object detectors with limited computation overhead on both datasets. We further analyze the contribution of each part of our MonoXiver in the ablation studies.

4.1. Setup

Dataset. The KITTI dataset [14] consists of 7,481 training images and 7,518 testing images. In KITTI’s experiments, we split the training data into a training subset with 3,712 images and a validation subset with 3,769 images following [5–7]. We conduct ablation studies on the defined split and report the test set results evaluated by the official KITTI benchmark. The Waymo dataset [51] contains 52,386 training and 39,848 validation images from the front camera. In Waymo experiments, we use a subset of its training set by sampling every third frame from the training sequences following [22, 45].

Evaluation Metrics. The KITTI vehicle benchmark evaluates detection results by the 40-point interpolated av-

Methods	Venues	Extra Info.	$AP_{BEV R40} \uparrow$			$AP_{3D R40} \uparrow$		
			Easy	Mod.	Hard	Easy	Mod.	Hard
Kinematic3D [2]	ECCV20	Temporal	26.69	17.52	13.10	19.07	12.72	9.17
AutoShape [34]	ICCV21	CAD	30.66	20.08	15.95	22.47	14.17	11.36
DCD [25]	ECCV22	CAD	32.55	21.50	18.25	23.81	15.90	13.21
MonoDistill [10]	ICLR22	LiDAR	31.87	22.59	19.72	22.97	16.03	13.60
DID-M3D [43]	ECCV22	LiDAR	<u>32.95</u>	22.76	19.83	<u>24.40</u>	16.29	13.75
DD3D [42]	ICCV21	Depth	30.98	22.56	20.03	23.22	16.34	14.20
DFM [55]	ECCV22	Temporal + LiDAR	31.71	22.89	19.97	22.94	16.82	14.65
Pseudo-Stereo [9]	CVPR22	Depth + LiDAR	32.84	<u>23.67</u>	<u>20.64</u>	23.74	<u>17.74</u>	<u>15.14</u>
MonoFlex [59]	CVPR21	None	28.23	19.75	16.89	19.94	13.89	12.07
GUPNet [35]	ICCV21		30.29	21.19	18.20	20.11	14.20	11.77
DEVIANT [22]	ECCV22		29.65	20.44	17.43	21.88	14.46	11.89
Homography [15]	CVPR22		29.60	20.68	17.81	21.75	14.94	13.07
DimEmbedding [61]	CVPR22		32.82	21.98	18.70	23.62	16.10	13.41
MonoCon [30]	AAAI22	None	31.12	22.10	19.00	22.50	16.46	13.95
Our MonoXiver + MonoCon	-		34.14	25.37	22.20	25.24	19.04	16.39

Table 2: **Comparisons with state-of-the-art methods on the Car category on the KITTI test set.** According to the KITTI protocol, methods are ranked based on their performance under the moderate difficulty setting. We highlight the best results in **bold** and the second-best results in underline.

Methods	$AP_{3D R40} \uparrow$			Relative Improvement
	Easy	Moderate	Hard	
SMOKE [33]	10.43	7.09	5.57	
SMOKE + Ours	11.58	9.40	7.75	
<i>Improvement</i>	+1.15	+2.31	+2.18	11%-33%
MonoDLE [39]	17.94	13.72	12.10	
MonoDLE + Ours	21.15	16.19	14.75	
<i>Improvement</i>	+3.21	+2.47	+2.65	18%-22%
MonoCon [30]	26.33	19.01	15.98	
MonoCon + Ours	30.48	22.40	19.13	
<i>Improvement</i>	+4.15	+3.39	+3.15	16%-20%

Table 3: **The effectiveness of MonoXiver based on different methods on the KITTI validation dataset.** In order to demonstrate the ability of the proposed MonoXiver to generalize across different detectors, we utilized three distinct base detectors with varying levels of detection accuracy. We observe that MonoXiver consistently yields significant improvements across all three base detectors.

erage precision (AP_{R40}) of 3D bounding boxes in 3D space ($AP_{3D|R40}$) and bird eye's view ($AP_{BEV|R40}$) at $IoU_{3D} \geq 0.7$. The prediction results are evaluated based on three difficulty settings, *easy*, *moderate* and *hard*, according to the 2D box height, occlusion and truncation levels of objects. **The vehicle detection results on Waymo** are evaluated on two levels of difficulty including *Level_1* and *Level_2* at $IoU_{3D} \geq 0.5$. The level is assigned based on the number of LiDAR points included in each 3D box. Besides the AP_{3D} metric, Waymo uses the APH_{3D} metric to incorporate heading information in AP_{3D} .

Methods	Level_1		Level_2	
	$AP_{3D} \uparrow$	$APH_{3D} \uparrow$	$AP_{3D} \uparrow$	$APH_{3D} \uparrow$
PatchNet [36]	2.92	2.74	2.42	2.28
PCT [54]	4.20	4.15	4.03	4.15
GUPNet [35]	10.02	9.94	9.39	9.31
GUPNet + ours	11.47	11.35	10.67	10.56
<i>Improvement</i>	+1.45	+1.41	+1.28	+1.25
DEVIANT [22]	10.98	10.89	10.29	10.20
DEVIANT + ours	11.88	11.75	11.06	10.93
<i>Improvement</i>	+0.90	+0.86	+0.77	+0.73

Table 4: **The effectiveness of MonoXiver based on different methods on Waymo validation dataset.** In order to showcase the generalization capabilities of the proposed MonoXiver on large-scale datasets, we conducted testing on the challenging Waymo [51] validation set, utilizing two SOTA methods. We observed consistent improvements in performance.

Methods	FLOPs	Latency (ms)	FPS
SMOKE [33]/ + ours	42.82/48.19	23/31	43/32
MonoDLE [39]/ + ours	77.44/82.80	22/31	45/32
MonoCon [30]/ + ours	56.22/61.50	18/25	55/40

Table 5: **Computation overhead analysis.** The GFLOPs is computed based on an input size of (384, 1248), while the latency and FPS are evaluated on a NVIDIA RTX5000 GPU.

4.2. Experimental Results

Comparison with SOTA methods on the KITTI dataset.

We present the results of our proposed MonoXiver method on the challenging KITTI vehicle benchmark in Table 2. Notably, our method achieves the best performance across different evaluation metrics while only using image-level information, surpassing previous state-of-the-art methods by a large margin. Specifically, we observe a significant and consistent improvement from **2.44 AP** (hard settings) to **2.74 AP** (easy settings) **absolute increase** in AP_{3D} for the boosted MonoCon approach compared to the vanilla MonoCon approach. These consistent improvements demonstrate the effectiveness of our method. More qualitative and quantitative results are provided in the [Supplementary Materials](#).

Generalization abilities across backbone monocular 3D object detectors and datasets. We evaluate our method with various state-of-the-art backbone monocular 3D object detection methods on the KITTI and the Waymo validation set, reported in Table 3 and Tabel 4 respectively. Monoxiver consistently demonstrates significant improvements across different methods and different difficult levels. Specifically, on the well-established KITTI dataset, our MonoXiver is able to enhance various backbone detectors with varying levels of detection accuracy by up to 33% relative improvement. On the challenging large-scale Waymo dataset, our MonoXiver consistently improves the performance of strong baseline methods GUPNet and DEVIANT by up to 1.45 AP_{3D} on Level_1 and 1.28 AP_{3D} on Level_2. These results validate the effectiveness and robustness of our approach.

Computation overhead. As shown in Table 5, although our MonoXiver causes about an average of 8 ms overhead compared with baseline detectors, it still achieves real-time detection. We note that its inference speed could be improved by optimizing the detailed implementation (e.g., we still have for-loops in our current code, which can be easily paralleled for better efficiency).

Qualitative Comparison. We show the visualization comparisons with MonoCon in Fig. 2. It shows that MonoXiver achieves better 3D box center localization.

4.3. Ablation Studies

In this section, we report ablation studies of MonoXiver structure and different bounding box branches on the KITTI validation set. More ablation studies are provided in the [Supplementary Materials](#).

Effectiveness of MonoXiver structures. In this study, we explore the importance of ROI feature (Table 6 a.), the context information provided by geometric embeddings (Table 6 b.), and our complete MonoXiver model (Table 6 c.). In experiments a. and b., we replace the Perciver with a MLP layer to fuse the appearance feature and geometry feature to keep similar parameter size.

	Appearance	Geometry	Perciver	Easy/Mod./Hard
MonoCon [30]	-	-	-	26.33/19.01/15.98
a.	✓	-	-	29.30/21.04/18.22
b.	✓	✓	-	29.33/21.67/18.46
c.	✓	✓	✓	30.48/22.40/19.13

Table 6: Ablation studies of MonoXiver structure on KITTI validation set.

	Rescore	Res _{Loc}	Res _{Dim}	Easy	Mod.	Hard
MonoCon [30]	-	-	-	26.33	19.01	15.98
a.	✓	-	-	29.66	21.41	18.38
b.	✓	✓	-	30.32	22.30	19.04
c.	✓	✓	✓	30.48	22.40	19.13

Table 7: Ablation studies of the effect of different branches on KITTI validation set

Results show that the appearance feature plays the most important role in performance improvement. When using the appearance feature only, the Moderate AP is improved by 2.39. The performance is further improved by 0.6 AP and 0.7 AP after fusing with geometric embeddings and Perciver. It shows that Perceiver effectively fuses information between appearance and geometric embeddings.

Effectiveness of Different Head Branches. In Table 7, we present the impact of different branches on the detection performance of our MonoXiver. Since our method generates 25 proposals per each initial proposal, the rescoring head branch plays a crucial role in filtering out false detections, leading to significant improvements by up to 3.3 AP. The localization residual branch and the dimension residual branch further enhance the performance by about 1 AP, and we employ all branches to achieve a new state-of-the-art result.

5. Conclusion

This paper explores the task of monocular 3D object detection. It begins with the strong observation that significant improvements in detection performance can be achieved by a local grid search based on initially detected 3D bounding boxes. This leads to a novel denoising-based approach called MonoXiver. The approach utilizes the Perceiver I/O model to produce a unified feature embedding, leverages the self-attention mechanism for proposal verification, and ultimately delivers high-quality 3D box predictions. MonoXiver can serve as a lightweight second-stage processing module to significantly improve the accuracy of detection results while only requiring 1.5 hours of training on a single GPU card for the KITTI dataset. Comprehensive evaluation experiments on the well-established KITTI benchmark and the large-scale challenging Waymo dataset demonstrate the effectiveness of our design, with a new state-of-the-art

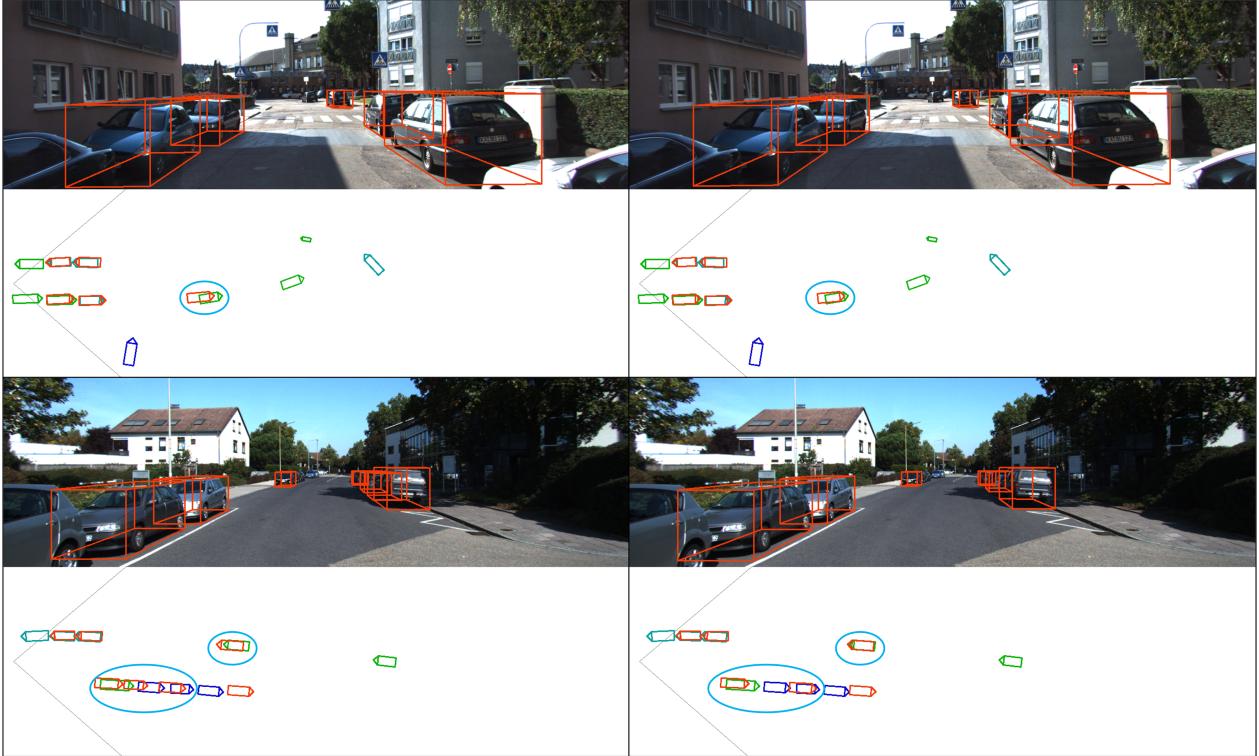


Figure 2: **Qualitative comparison of our MonoXiver with MonoCon on KITTI validation set [5].** The left column is MonoCon’s prediction result; the right column is our MonoXiver’s prediction result. The ground truth is shown in green and blue. The prediction result is shown in red.

achieved. Furthermore, our strong observation should encourage the community to further study refinement modules for 3D object detection.

Limitations. Our work keeps at most top-3 predictions per bottom-up proposal for evaluation. Future work might be reducing the number of top-3 prediction by incorporating stronger priors for helping denoising proposals in 3D, i.e. temporal modeling, 3D geometric priors, large foundation models, etc. Our work has only been comprehensively tested on monocular settings with significant improvement, and it has not been tested on multiview settings.

Acknowledgements

X. Liu and T. Wu were partly supported by ARO Grant W911NF1810295, NSF IIS-1909644, ARO Grant W911NF2210010, NSF IIS-1822477, NSF CMMI-2024688 and NSF IUSE-2013451. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ARO, NSF, DHHS or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any

copyright annotation thereon. The authors are grateful for the constructive comments by anonymous reviewers and area chairs.

References

- [1] Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *ICCV*, pages 9287–9296, 2019. [2](#)
- [2] Garrick Brazil, Gerard Pons-Moll, Xiaoming Liu, and Bernt Schiele. Kinematic 3d object detection in monocular video. In *ECCV*, pages 135–152. Springer, 2020. [2, 7](#)
- [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. [3, 6](#)
- [4] Hansheng Chen, Yuyao Huang, Wei Tian, Zhong Gao, and Lu Xiong. Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In *CVPR*, pages 10379–10388, 2021. [2](#)
- [5] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *CVPR*, pages 2147–2156, 2016. [1, 6, 9](#)
- [6] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Andrew G Berneshawi, Huimin Ma, Sanja Fidler, and Raquel Urtasun.

- 3d object proposals for accurate object class detection. In *NeurIPS*, pages 424–432. Citeseer, 2015. 1, 6
- [7] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, pages 1907–1915, 2017. 1, 6
- [8] Yongjian Chen, Lei Tai, Kai Sun, and Mingyang Li. Monopair: Monocular 3d object detection using pairwise spatial relationships. In *CVPR*, pages 12093–12102, 2020. 2
- [9] Yi-Nan Chen, Hang Dai, and Yong Ding. Pseudo-stereo for monocular 3d object detection in autonomous driving. In *CVPR*, 2022. 3, 7
- [10] Zhiyu Chong, Xinzhu Ma, Hong Zhang, Yuxin Yue, Haojie Li, Zhihui Wang, and Wanli Ouyang. Monodistill: Learning spatial features for monocular 3d object detection. In *ICLR*, 2022. 2, 7
- [11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3
- [12] 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
- [13] Simon Doll, Richard Schulz, Lukas Schneider, Viviane Benzin, Markus Enzweiler, and Hendrik Lensch. Spatialdetr: Robust scalable transformer-based 3d object detection from multi-view camera images with global cross-sensor attention. In *European Conference on Computer Vision*, pages 230–245. Springer, 2022. 3
- [14] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2, 6
- [15] Jiaqi Gu, Bojian Wu, Lubin Fan, Jianqiang Huang, Shen Cao, Zhiyu Xiang, and Xian-Sheng Hua. Homography loss for monocular 3d object detection. In *CVPR*, 2022. 7
- [16] Xiaoyang Guo, Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Liga-stereo: Learning lidar geometry aware representations for stereo-based 3d detector. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3153–3163, 2021. 3
- [17] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 4
- [18] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022. 3
- [19] Kuan-Chih Huang, Tsung-Han Wu, Hung-Ting Su, and Winston H Hsu. Monodtr: Monocular 3d object detection with depth-aware transformer. In *CVPR*, 2022. 2
- [20] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Kopputala, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. *ICLR*, 2022. 1, 2, 3, 4
- [21] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *ICML*, pages 4651–4664. PMLR, 2021. 3
- [22] Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parhami, and Xiaoming Liu. Deviant: Depth equivariant network for monocular 3d object detection. In *ECCV*, 2022. 6, 7
- [23] Abhinav Kumar, Garrick Brazil, and Xiaoming Liu. Groomed-nms: Grouped mathematically differentiable nms for monocular 3d object detection. In *CVPR*, pages 8973–8983, 2021. 2
- [24] Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object key-points for autonomous driving. In *ECCV*, pages 644–660. Springer, 2020. 2
- [25] Yingyan Li, Yuntao Chen, Jiawei He, and Zhaoxiang Zhang. Densely constrained depth estimator for monocular 3d object detection. *ECCV*, 2022. 2, 7
- [26] Zhuoling Li, Zhan Qu, Yang Zhou, Jianzhuang Liu, Haoqian Wang, and Lihui Jiang. Diversity matters: Fully exploiting depth clues for reliable monocular 3d object detection. In *CVPR*, 2022. 2
- [27] Zhiqi Li, Wenhui Wang, Hongyang Li, Enze Xie, Chong-hao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers. *ECCV*, 2022. 3
- [28] Qing Lian, Peiliang Li, and Xiaozhi Chen. Monojsg: Joint semantic and geometric cost volume for monocular 3d object detection. In *CVPR*, 2022. 2
- [29] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 6
- [30] Xianpeng Liu, Nan Xue, and Tianfu Wu. Learning auxiliary monocular contexts helps monocular 3d object detection. In *AAAI*, 2022. 1, 2, 7, 8
- [31] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. *ECCV*, 2022. 3
- [32] 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
- [33] Zechen Liu, Zizhang Wu, and Roland Tóth. Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *CVPR Workshops*, pages 996–997, 2020. 2, 7
- [34] Zongdai Liu, Dingfu Zhou, Feixiang Lu, Jin Fang, and Liangjun Zhang. Autoshape: Real-time shape-aware monocular 3d object detection. In *ICCV*, pages 15641–15650, 2021. 2, 7
- [35] Yan Lu, Xinzhu Ma, Lei Yang, Tianzhu Zhang, Yating Liu, Qi Chu, Junjie Yan, and Wanli Ouyang. Geometry uncertainty projection network for monocular 3d object detection. *arXiv:2107.13774*, 2021. 2, 7

- [36] Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. In *ECCV*, pages 311–327. Springer, 2020. [2](#), [7](#)
- [37] Xinzhu Ma, Wanli Ouyang, Andrea Simonelli, and Elisa Ricci. 3d object detection from images for autonomous driving: a survey. *arXiv preprint arXiv:2202.02980*, 2022. [1](#)
- [38] Xinzhu Ma, Zhihui Wang, Haojie Li, Pengbo Zhang, Wanli Ouyang, and Xin Fan. Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. In *ICCV*, pages 6851–6860, 2019. [2](#)
- [39] Xinzhu Ma, Yinmin Zhang, Dan Xu, Dongzhan Zhou, Shuai Yi, Haojie Li, and Wanli Ouyang. Delving into localization errors for monocular 3d object detection. In *CVPR*, pages 4721–4730, 2021. [1](#), [2](#), [7](#)
- [40] Yuexin Ma, Tai Wang, Xuyang Bai, Huitong Yang, Yuenan Hou, Yaming Wang, Yu Qiao, Ruigang Yang, Dinesh Manocha, and Xinge Zhu. Vision-centric bev perception: A survey. *arXiv preprint arXiv:2208.02797*, 2022. [3](#)
- [41] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learning and geometry. In *CVPR*, pages 7074–7082, 2017. [2](#)
- [42] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *ICCV*, pages 3142–3152, 2021. [2](#), [7](#)
- [43] Liang Peng, Xiaopei Wu, Zheng Yang, Haifeng Liu, and Deng Cai. Did-m3d: Decoupling instance depth for monocular 3d object detection. In *ECCV*, 2022. [7](#)
- [44] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, pages 194–210. Springer, 2020. [3](#)
- [45] Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *CVPR*, pages 8555–8564, 2021. [2](#), [3](#), [6](#)
- [46] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. [6](#)
- [47] Xuepeng Shi, Qi Ye, Xiaozhi Chen, Chuangrong Chen, Zhixiang Chen, and Tae-Kyun Kim. Geometry-based distance decomposition for monocular 3d object detection. *arXiv:2104.03775*, 2021. [2](#)
- [48] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Peter Kortschieder, and Elisa Ricci. Are we missing confidence in pseudo-lidar methods for monocular 3d object detection? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3225–3233, 2021. [2](#)
- [49] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kortschieder. Disentangling monocular 3d object detection. In *ICCV*, pages 1991–1999, 2019. [2](#)
- [50] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. ViL-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019. [3](#)
- [51] Pei Sun, Henrik Kretschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. [2](#), [6](#), [7](#)
- [52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5988–6008, 2017. [3](#), [5](#)
- [53] Li Wang, Liang Du, Xiaoqing Ye, Yanwei Fu, Guodong Guo, Xiangyang Xue, Jianfeng Feng, and Li Zhang. Depth-conditioned dynamic message propagation for monocular 3d object detection. In *CVPR*, pages 454–463, 2021. [2](#)
- [54] Li Wang, Li Zhang, Yi Zhu, Zhi Zhang, Tong He, Mu Li, and Xiangyang Xue. Progressive coordinate transforms for monocular 3d object detection. *Advances in Neural Information Processing Systems*, 34, 2021. [2](#), [7](#)
- [55] Tai Wang, Pang Jiangmiao, and Lin Dahua. Monocular 3d object detection with depth from motion. In *ECCV*, 2022. [3](#), [7](#)
- [56] 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 *CoRL*, pages 180–191, 2022. [3](#)
- [57] Yu Xiang, Wongun Choi, Yuanqing Lin, and Silvio Savarese. Data-driven 3d voxel patterns for object category recognition. In *CVPR*, pages 1903–1911, 2015. [2](#)
- [58] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *CVPR*, pages 2403–2412, 2018. [4](#)
- [59] Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *CVPR*, pages 3289–3298, 2021. [2](#), [7](#)
- [60] Yinmin Zhang, Xinzhu Ma, Shuai Yi, Jun Hou, Zhihui Wang, Wanli Ouyang, and Dan Xu. Learning geometry-guided depth via projective modeling for monocular 3d object detection. *arXiv:2107.13931*, 2021. [2](#)
- [61] Yunpeng Zhang, Wenzhao Zheng, Zheng Zhu, Guan Huang, Dalong Du, Jie Zhou, and Jiwen Lu. Dimension embeddings for monocular 3d object detection. In *CVPR*, 2022. [7](#)
- [62] Zhihang Zou, Xiaoqing Ye, Liang Du, Xianhui Cheng, Xiao Tan, Li Zhang, Jianfeng Feng, Xiangyang Xue, and Er-rui Ding. The devil is in the task: Exploiting reciprocal appearance-localization features for monocular 3d object detection. In *ICCV*, pages 2713–2722, 2021. [2](#)