APPENDIX

Anonymous authors
Paper under double-blind review

A ADDITIONAL RELATED WORK

Object Detection with Transformers. 2D object detectors (Girshick, 2015; Ren et al., 2015; Lin et al., 2017a;b; Tian et al., 2019) have achieved excellent performance in recent years, but count on cumbersome non-maximum suppression (NMS) post-processing and rule-based label assignment. To circumvent it, the seminal work DETR (Carion et al., 2020b) constructs a novel framework by adapting the powerful transformer (Vaswani et al., 2017) in natural language processing into computer vision for 2D detection. DETR detects objects on the image by an encoder-decoder architecture and conducts set prediction aided by Hungary Matching Algorithm (Carion et al., 2020b). However, due to the quadratic computational complexity of attention, DETR requires the expensive 500 epochs to be fully trained. To accelerate the convergence, Deformable DETR (Zhu et al., 2020) designs sparse deformable attention mechanisms and achieves better performance with only 50-epoch training. ACT (Zheng et al., 2020) boosts the time efficiency by introducing adaptive clustering algorithms during inference. SMCA (Gao et al., 2021) proposes Gaussian-modulated co-attention mechanisms that refocus the attention of each query into object-centric areas. Besides, DETR is further enhanced by placing anchors (Wang et al., 2021), redesigning as two stages (Sun et al., 2021a;b), setting conditional attention (Meng et al., 2021), embedding dense priors (Yao et al., 2021), introducing query denoising (Li et al., 2022a) and so on (Dai et al., 2021; Misra et al., 2021). For image-based 3D object detection, DETR3D (Wang et al., 2022) and PETR (Liu et al., 2022a) adopt vanilla transformers with 3D object queries to aggregate surrounding visual features in an end-to-end way. BEVFormer (Li et al., 2022b) utilizes a spatiotemporal transformer to generate BEV representations from multi-view images. In contrast, our MonoDETR equip the vanilla transformer with depth guidance for adaptive scene-level depth understanding, and can tackle both single-view and multi-view circumstances.

B DETAILS OF ATTRIBUTE PREDICTION AND LOSS FUNCTIONS

After the depth-guided transformer, we adopt detection heads to estimate six attributes for each object query: object category, 2D size (l, r, t, b), projected 3D center (x_{3D}, y_{3D}), depth d_{reg}, 3D size (h_{3D}, w_{3D}, l_{3D}) and orientation α. All queries share the head weights for the same attribute. Specifically, we utilize one linear projection layer for the object category, and two-layer MLP for depth, 3D size and orientation, and three-layer MLP for 2D size and projected 3D center.

Projected 3D Center (x_{3D}, y_{3D}). We directly output the coordinate (x_{3D}, y_{3D}) of each query's projected 3D center on the image, which thus discards two types of widely-adopted offsets. The first is the 2D-to-3D offset for recovering the projected 3D center from the predicted 2D center. The other is the quantization offset caused by the downsampled heatmap, which is a requisite for existing center-guided methods. By this, we can obtain the projected 3D center of each object in one step without the error of intermediate offsets, contributing to better localization accuracy. We adopt L1 loss for the center estimation and denote it as $\mathcal{L}_{xy3D}$.

Object Category and 2D Size (l, r, t, b). We detect objects of three categories, car, pedestrian and cyclist, in KITTI (Geiger et al., 2012), and adopt Focal loss (Lin et al., 2017b) for optimization, denoted as $\mathcal{L}_{class}$. Referring to FCOS (Tian et al., 2019), we obtain the 2D bounding box of an object by predicting the distances from its four sides, l, r, t, b, to the projected 3D center (x_{3D}, y_{3D}). Both (l, r, t, b) and (x_{3D}, y_{3D}) are normalized from 0 to 1 by the image size. We apply L1 loss for the distances and GIoU loss (Rezatofighi et al., 2019) for the recovered 2D bounding box following DETR (Carion et al., 2020a), denoted as $\mathcal{L}_{lrtb}$ and $\mathcal{L}_{GIoU}$, respectively.

3D Size (h_{3D}, w_{3D}, l_{3D}) **and Orientation** α**.** Instead of predicting the residuals to the mean shape values, we follow MonoDLE (Ma et al., 2021) to use the 3D IoU oriented loss for 3D sizes. We divide the heading angle into multiple bins with residuals and adopt MultiBin loss (Chen et al., 2020; Zhou et al., 2019) to optimize the prediction of orientation. The two losses are respectively denoted as $\mathcal{L}_{size3D}$ and $\mathcal{L}_{orien}$.

Depth d_{pred}**.** To estimate the final object depth d_{pred} more accurately, we average three predicted values: d_{reg} regressed by the detection head, d_{geo} converted by the predicted 2D and 3D sizes, and $d_{map}(x_{3D}, y_{3D})$ interpolated from D_{fg}. We formulate as

$$d_{geo} = f\,\frac{h_{3D}}{t+b}, \qquad d_{pred} = (d_{reg} + d_{geo} + d_{map}(x_{3D}, y_{3D}))/3, \tag{1}$$

where h_{3D} and $t+b$ denote the predicted heights of 3D and 2D sizes, and f denotes the camera focal length. We then adopt Laplacian aleatoric uncertainty loss (Chen et al., 2020) for the overall d_{pred}, formulated as

$$\mathcal{L}_{depth} = \frac{\sqrt{2}}{\sigma}\|d_{gt} - d_{pred}\|_1 + \log(\sigma), \tag{2}$$

where σ denotes the standard deviation predicted together with d_{reg}, and d_{gt} denotes the ground-truth depth label of the object.

Bipartite Matching. To correctly match each query with a ground-truth object, we calculate the loss for each query-label pair and utilize Hungarian algorithm (Carion et al., 2020a) to find the globally optimal matching. For each pair, we integrate the losses of six attributes into two groups. The first group contains object category, 2D size and the projected 3D center, since these attributes mainly concern 2D visual appearances of the image, formulated as

$$\mathcal{L}_{2D} = \lambda_1\mathcal{L}_{class} + \lambda_2\mathcal{L}_{xy3D} + \lambda_3\mathcal{L}_{lrtb} + \lambda_4\mathcal{L}_{GIoU}, \tag{3}$$

where we set $\lambda_{1\sim4}$ as 2, 10, 5, 2, respectively. The second group consists of the depth, 3D size and orientation, which are 3D spatial properties of the object, formylated as

$$\mathcal{L}_{3D} = \mathcal{L}_{size3D} + \mathcal{L}_{orien} + \mathcal{L}_{depth}. \tag{4}$$

As the network generally predicts less accurate 3D attributes than 2D attributes, especially at the beginning of training, the value of $\mathcal{L}_{3D}$ is unstable and would disturb the matching process. We only utilize $\mathcal{L}_{2D}$ as the matching cost for matching each query-label pair.

Overall Loss. After the matching, we obtain N_{gt} valid pairs out of N queries, where N_{gt} denotes the number of ground-truth objects. Then, the overall loss of a training image is formulated as

$$\mathcal{L}_{overall} = \frac{1}{N_{gt}} \cdot \sum_{n=1}^{N_{gt}} (\mathcal{L}_{2D} + \mathcal{L}_{3D}) + \mathcal{L}_{dmap}. \tag{5}$$

$\mathcal{L}_{dmap}$ represents the loss of the predicted categorical foreground depth map D_{fg}, for which we also utilize Focal loss (Lin et al., 2017b).

C Additional Results

Car Category on KITTI *val* Set. We list more results of the car category on KITTI *val* set under different IoU thresholds in Table 1, where our MonoDETR all achieves the highest detection accuracy. Compared to the second-best MonoDTR (Huang et al., 2022) that is a center-guided method with external depth supervision, our MonoDETR only requires object-wise depth labels and surpasses it by significant gains for the easy level, e.g., +4.53% AP_{BEV}@IoU=0.7 and +4.83% AP_{3D}@IoU=0.5.

Pedestrian and Cyclist Categories. In Table 2, we report the scores for pedestrian and cyclist categories on KITTI *test* set both under the IoU threshold of 0.5. As these two categories contain much fewer training samples than car, it is more challenging for the network to accurately detect them. As shown, MonoDETR achieves superior AP_{3D} to other methods without additional data, indicating our superior generalization ability on other categories.

Table 1: **Performance of the car category on KITTI *val* sets under different IoU thresholds.** We utilize bold numbers to highlight the best results, and blue for the second-best ones.

Method	AP_{BEV}@IoU=0.7			AP_{3D}@IoU=0.5			AP_{BEV}@IoU=0.5		
	Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
SMOKE (Liu et al., 2020)	19.99	15.61	15.28	-	-	-	-	-	-
MonoPair (Chen et al., 2020)	24.12	18.17	15.76	55.38	42.39	37.99	61.06	47.63	41.92
MonoRCNN (Shi et al., 2021)	25.29	19.22	15.30	-	-	-	-	-	-
MonoDLE (Ma et al., 2021)	24.97	19.33	17.01	55.41	43.42	37.81	60.73	46.87	41.89
IAFA (Zhou et al., 2020)	22.75	19.60	19.21	-	-	-	-	-	-
MonoGeo (Zhang et al., 2021a)	27.15	21.17	18.35	56.59	43.70	39.37	61.96	47.84	43.10
RTM3D (Li et al., 2020)	24.74	22.03	18.05	52.59	40.96	34.95	56.90	44.69	41.75
GUPNet (Lu et al., 2021)	31.07	22.94	19.75	57.62	42.33	37.59	61.78	47.06	40.88
MonoDTR (Huang et al., 2022)	33.33	25.35	21.68	64.03	47.32	42.20	69.04	52.47	45.90
MonoDETR (Ours)	**37.86**	**26.95**	**22.80**	**68.86**	**48.92**	**43.57**	**72.30**	**53.10**	**46.62**
Improvement	+4.53	+1.60	+1.12	+4.83	+1.60	+1.37	+3.26	+0.63	+0.72

Table 2: **Performance of the pedestrian and cyclist categories on KITTI *test* set.** We utilize bold numbers to highlight the best results, and blue ones for the second-best ones.

Method	Pedestrian, AP_{3D}			Cyclist, AP_{3D}		
	Easy	Mod.	Hard	Easy	Mod.	Hard
Movi3D (Simonelli et al., 2020)	8.99	5.44	4.57	1.08	0.63	0.70
MonoGeo (Zhang et al., 2021a)	8.00	5.63	4.71	4.73	2.93	2.58
MonoFlex (Zhang et al., 2021b)	9.43	6.31	5.26	4.17	2.35	2.04
MonoDLE (Ma et al., 2021)	9.64	6.55	5.44	4.59	2.66	2.45
MonoPair (Chen et al., 2020)	10.02	6.68	5.53	3.79	2.12	1.83
MonoDETR (Ours)	**12.54**	**7.89**	**6.65**	**7.33**	**4.18**	**2.92**
Improvement	+2.52	+1.21	+1.12	+2.60	+1.25	+0.34

D Additional Ablation Study

Depth Discretization. We explore different depth discretization methods for the foreground depth map d_{fg} in Table 3. 'UD', 'SID' and 'LID' denote uniform, spacing-increasing, and linear-increasing discretizations, respectively. Instead of the weighted summation of depth bins, 'LID + argmax' outputs the depth value of the most confident bin. For 'Continuous Rep.', we directly regress the continuous depth value and optimize it by L1 loss. As reported, 'LID' performs the best than other discretization methods, since the linear-increasing intervals can suppress the larger estimation errors of farther objects. Also, 'LID' with weighted summation outperforms 'LID + argmax' for aggregating more depth cues from the predicted confidence of other depth bins.

Bipartite Matching. Our best solution only utilizes $\mathcal{L}_{2D}$ as the matching cost for each query-label pair. We investigate how it performs to append more 3D losses into the matching cost. As reported in Table 4, adding $\mathcal{L}_{size3D}$ or $\mathcal{L}_{orien}$ would adversely influence the performance due to their unstable prediction during training. Further, adding $\mathcal{L}_{depth}$ or the whole $\mathcal{L}_{3D}$ even leads to training collapse, which is caused by the ill-posed depth estimation from monocular images.

Transformer Blocks and FFN Channels. In Table 5, we experiment different block numbers of the visual encoder and depth-guided decoder, along with the latent channels of feed-forward neural network (FFN). As reported, MonoDETR achieves the best performance with the 3-block visual encoder, 3-block depth-guided decoder, and 256-channel FFN. Different from DETR's (Carion et al., 2020a) 6-block encoder, 6-block decoder, and 1024-channel FFN for COCO (Lin et al., 2014) dataset, MonoDETR adopts a lighter-weight transformer architecture because of the limited training samples in KITTI (Geiger et al., 2012) dataset.

Table 3: **The design of depth discretization in the foreground depth map.** 'Continuous Rep.' denotes the continuous representation of depth.

Settings	Easy	Mod.	Hard
LID	**28.84**	**20.61**	**16.38**
UD	25.61	18.90	15.49
SID	26.05	18.95	15.59
LID + argmax	21.61	15.21	12.13
Continuous Rep.	24.36	17.24	14.48

Table 4: **The design of bipartite matching.** 'w' denotes adding the loss to the matching cost. '-' denotes training collapse.

Matching Cost	Easy	Mod.	Hard
$\mathcal{L}_{2D}$	**28.84**	**20.61**	**16.38**
w $\mathcal{L}_{size3D}$	27.13	19.21	15.93
w $\mathcal{L}_{orien}$	25.78	18.63	15.12
w $\mathcal{L}_{depth}$	-	-	-
w $\mathcal{L}_{3D}$	-	-	-

Table 5: **Transformer blocks and FFN channels.** FFN denotes the feed-forward neural network.

	Set.	Easy	Mod.	Hard
Visual Encoder Blocks	2	26.72	18.73	15.43
	3	**28.84**	**20.61**	**16.38**
	4	27.37	20.04	16.09
Depth-guided Decoder Blocks	2	25.55	18.58	15.41
	3	**28.84**	**20.61**	**16.38**
	4	25.31	18.29	15.11
FFN Channels	256	**28.84**	**20.61**	**16.38**
	512	27.24	18.93	15.54
	1024	26.77	19.07	15.87

E Implementation Details

Monocular Experiments on KITTI (Geiger et al., 2012). We adopt ResNet-50 (He et al., 2016) as our feature backbone. To save GPU memory, we apply deformable attention mechanisms (Zhu et al., 2020) for the visual encoder and visual cross-attention layers, and utilize the vanilla attention (Carion et al., 2020a) to better capture global spatial structures for the depth encoder and depth cross-attention layers. We utilize 8 heads for all attention modules and set the number of queries N as 50. We set the channel C and all MLP's latent feature dimension as 256. For the foreground depth map, we set $[d_{min}, d_{max}]$ as $[0m, 60m]$ and the number of bins k as 80. On a single GeForce RTX 3090 GPU, we train MonoDETR for 195 epochs with batch size 16 and the learning rate 2×10^{-4}. We adopt AdamW (Loshchilov & Hutter, 2018) optimizer with weight decay 10^{-4} and decrease the learning rate at 125 and 165 epochs by 0.1. For data augmentation on KITTI *test* set, we adopt random flip and photometric distortion following previous works (Zhang et al., 2021b; Ma et al., 2021; Zhou et al., 2019), but for the *val* set, we also use random crop to further boost the performance. For training stability, we discard the training samples with depth labels larger than 65 meters or smaller than 2 meters. During inference, we simply filter out the object queries with the category confidence lower than 0.2 without NMS post-processing, and recover the 3D bounding box using the predicted six attributes following previous works.

Multi-view Experiments on nuScenes (Caesar et al., 2019). For fair comparison with existing multi-view methods, MonoDETR-MV follows most of the settings in (Liu et al., 2022a;b), including VoVNetV2 (Lee & Park, 2020) feature backbone, 3D object queries, 3D position embeddings, temporal information, loss functions and data augmentation. We utilize 2 blocks for the depth encoder to better encode multi-view depth embeddings, and apply the depth cross-attention layer at the end of each decoder block for training stability. The number of queries N for 6-view images is set as 900, which predict 10 object categories. The configurations of depth predictor, e.g., $[d_{min}, d_{max}]$ and k are the same as monocular experiments. We train MonoDETR-MV for 24 epochs (2x schedule) on 8 NVIDIA A100 GPUs with a batch size of 8. We adopt AdamW (Loshchilov & Hutter, 2018) optimizer with weight decay 10^{-2} and utilize the learning rate 2×10^{-4} with the cosine scheduler.

F ADDITIONAL VISUALIZATION

In Figure 1, we show the detection results of our MonoDETR and the variant without the depth-guided transformer on KITTI *val* set. Benefited from the depth guidance, MonoDETR obtains a global understanding of the scene-level spatial structure and the inter-object relations. This enables MonoDETR to well detect the objects occluded by others or truncated by images, and filter out the objects of ignored categories, e.g., van and truck.

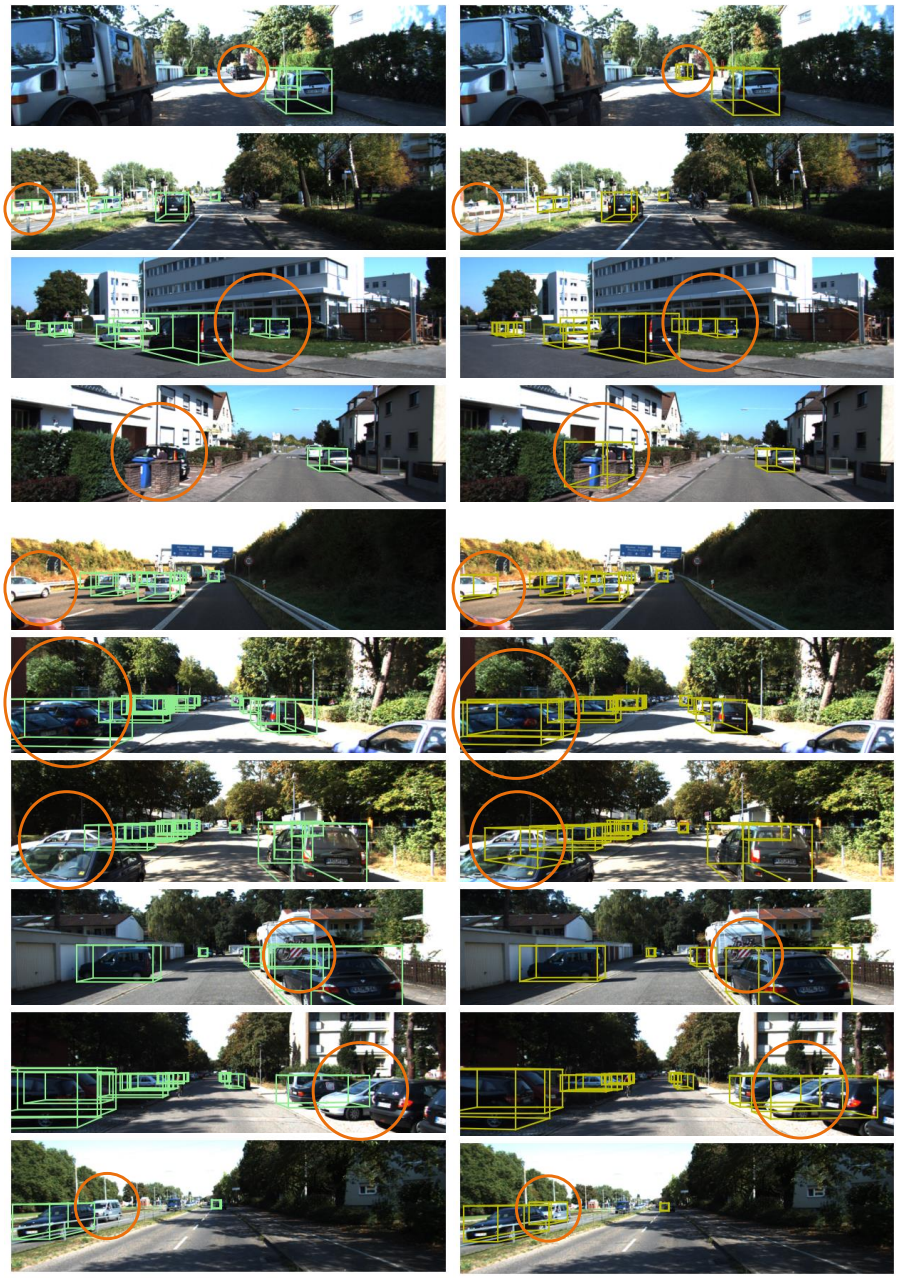

Figure 1: **Visualization of detection results.** We utilize green boxes for the variant without depth-guided transformer (Left) and yellow boxes for MonoDETR (Right). We use red circles to emphasize the detection difference.

Figure 2: **Depth errors for different variants of MonoDETR.** The x axis and y axis denote the AP_{3D} under the moderate level and the mean depth errors on KITTI *val* set, respectively.

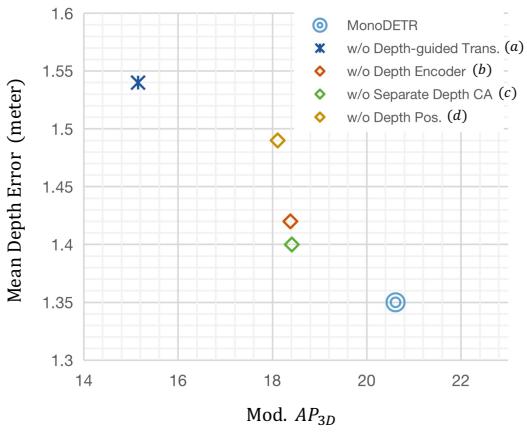

Table 6: **Quantitative results of depth errors.** We construct four network variants of MonoDETR by removing one of the components at a time. We respectively remove the depth-guided transformer, depth encoder, separate depth cross-attention layer, and depth positional encodings, denoted as '(a), (b), (c), (d)'. We show their AP_{3D} under the moderate level and the mean depth errors with standard deviations.

Architecture	AP_{3D} ↑	Depth Error ↓
MonoDETR	**20.61**	**1.35±2.07**
(a)	15.15	1.54±2.29
(b)	18.38	1.42±2.10
(c)	18.41	1.40±2.11
(d)	18.11	1.49±2.29

G Depth Error Analysis

To demonstrate the effectiveness of our depth-guided design, we show the depth error comparison for different variants of MonoDETR. We utilize four network variants, denoted as '(a), (b), (c), (d)' in Figure 2 and Table 6. We calculate their predicted mean depth errors and standard deviations on KITTI *val* set. With our depth-guided transformer, the depth estimation can be well benefited, which reduces the mean error from 1.54 meters to 1.35 meters and improves the AP_{3D} by +5.46% under the moderate level. In addition, our best solution of 20.61% AP_{3D} performs lower error variance of ±2.07 than others, indicating our depth-guided transformer can produce more stable depth estimation of objects.

H Anonymous Code Release

For reproducibility, we anonymously release our codes in https://anonymous.4open.science/r/MonoDETR_anonymous-FFC0/.

REFERENCES

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. *CoRR*, abs/1903.11027, 2019. URL http://arxiv.org/abs/1903.11027. 4

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, 2020a. 1, 2, 3, 4

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020b. 1

Yongjian Chen, Lei Tai, Kai Sun, and Mingyang Li. Monopair: Monocular 3d object detection using pairwise spatial relationships. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 3

Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1601–1610, 2021. 1

Peng Gao, Minghang Zheng, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fast convergence of detr with spatially modulated co-attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3621–3630, October 2021. 1

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, 2012. doi: 10.1109/CVPR.2012.6248074. 1, 3, 4

Ross Girshick. Fast r-cnn. In *IEEE International Conference on Computer Vision*, 2015. 1

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 4

Kuan-Chih Huang, Tsung-Han Wu, Hung-Ting Su, and Winston H Hsu. Monodtr: Monocular 3d object detection with depth-aware transformer. *arXiv preprint arXiv:2203.10981*, 2022. 2, 3

Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13906–13915, 2020. 4

Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13619–13627, 2022a. 1

Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. In *European Conference on Computer Vision*, 2020. 3

Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022b. 1

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014. 3

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017a. 1

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017b. 1, 2

Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. *arXiv preprint arXiv:2203.05625*, 2022a. 1, 4

Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Qi Gao, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petrv2: A unified framework for 3d perception from multi-camera images. *arXiv preprint arXiv:2206.01256*, 2022b. 4

Zechen Liu, Zizhang Wu, and Roland Tóth. SMOKE: single-stage monocular 3d object detection via keypoint estimation. *CoRR*, abs/2002.10111, 2020. URL https://arxiv.org/abs/2002.10111. 3

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 4

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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3111–3121, October 2021. 3

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4721–4730, June 2021. 2, 3, 4

Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional detr for fast training convergence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3651–3660, 2021. 1

Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2906–2917, 2021. 1

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. 1

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 658–666, 2019. 1

Xuepeng Shi, Qi Ye, Xiaozhi Chen, Chuangrong Chen, Zhixiang Chen, and Tae-Kyun Kim. Geometry-based distance decomposition for monocular 3d object detection. In *IEEE International Conference on Computer Vision*, 2021. 3

Andrea Simonelli, Samuel Rota Bulò, Lorenzo Porzi, Elisa Ricci, and Peter Kontschieder. Towards generalization across depth for monocular 3d object detection. In *Proceedings of the European Conference on Computer Vision*, 2020. 3

Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14454–14463, 2021a. 1

Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris M Kitani. Rethinking transformer-based set prediction for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3611–3620, 2021b. 1

Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9627–9636, 2019. 1

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1

Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based detector. *arXiv preprint arXiv:2109.07107*, 2021. 1

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*, pp. 180–191. PMLR, 2022. 1

Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient detr: Improving end-to-end object detector with dense prior. *arXiv preprint arXiv:2104.01318*, 2021. 1

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 preprint arXiv:2107.13931*, 2021a. 3

Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3289–3298, June 2021b. 3, 4

Minghang Zheng, Peng Gao, Xiaogang Wang, Hongsheng Li, and Hao Dong. End-to-end object detection with adaptive clustering transformer. *arXiv preprint arXiv:2011.09315*, 2020. 1

Dingfu Zhou, Xibin Song, Yuchao Dai, Junbo Yin, Feixiang Lu, Miao Liao, Jin Fang, and Liangjun Zhang. Iafa: Instance-aware feature aggregation for 3d object detection from a single image. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 3

Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *CoRR*, abs/1904.07850, 2019. URL https://arxiv.org/abs/1904.07850. 2, 4

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 1, 4