MonoSISTR: Monocular 3D Object Detection via Staged Iterative Structure and Target Refinement

Genlin Zhou[1] , Cheng Feng[2] , Feng Lu[1] , Zige Wang[1] , Zhen Chen[1] , Congxuan Zhang[1,*]

[1] Nanchang Hangkong University, Nanchang, China

[2] Beihang University, Beijing, China

Abstract

Monocular 3D object detection aims to predict object category, position, size, and orientation from a single RGB image. Existing DETR-based monocular 3D detectors suffer from maintaining consistent high-confidence responses due to weak or incomplete target features, resulting in information loss for distant and occluded objects during encoding-decoding. Firstly, to address the core challenge of insufficient target feature perception in complex scenes, we propose a staged iterative monocular 3D detector that progressively refines targets from coarse to fine through multiple paired encoding-decoding stages, significantly improving both feature utilization and network convergence. Furthermore, each stage integrates a dynamic target iteration module that continuously enhances query representation by focusing on high-confidence regional features, thereby enhancing the model's perception of potential targets. Finally, we design a dual-branch depth estimator with parallel global and local processing for a comprehensive representation of the depth feature. Experimental results show that our method achieves superior performance over prior approaches on challenging scenarios (e.g., distant and occluded objects) in the KITTI dataset without auxiliary data, while maintaining competitive accuracy on the nuScenes benchmark under frontal-view settings.

1. Introduction

In hardware-constrained scenarios, early monocular 3D object detection methods mainly relied on hand-crafted geometric constraints [42][12][9][48] and prior knowledge [19][20][27][5] for depth reasoning. However, their limited feature representation capacity hindered performance in complex scenes. The advent of deep learning brought significant advances through CNN-based approaches that learned depth-aware visual features, although challenges persisted in depth estimation accuracy and multiscale feature fusion. Recently, frameworks based on DETR[4] architectures have enabled more flexible depth feature learning through attention mechanisms and query interactions.

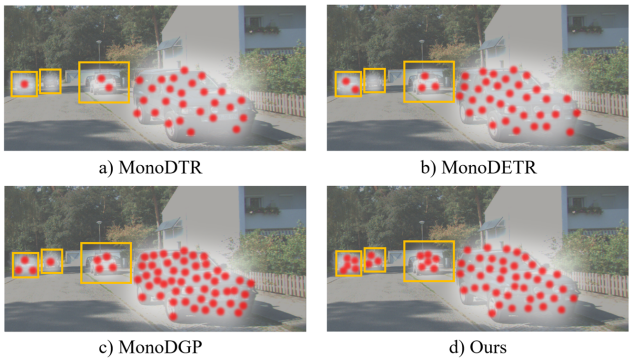

Figure 1. **Comparison of confidence response levels to weak target signals across different end-to-end detectors**, where red spots indicate the decoder's perception intensity of target features.

In order to achieve accurate estimation of 3D attributes, particularly for moderate and hard targets, numerous existing methods [19][46][43] have focused on optimizing object depth estimation to improve the accuracy of position prediction. However, depth prediction and its error estimation remain fundamentally constrained by the feature processing capability of the encoding-decoding stages. As illustrated in Fig.1, the serial encoder-decoder architecture represented by [45] struggles to maintain consistent high-confidence responses for targets with weak or incomplete features, while different encoded features exhibit varying contributions to 3D attribute learning. These issues collectively form the core bottleneck for performance improvement in monocular 3D detection, with their impact being particularly pronounced in challenging scenarios where robustness is most critical.

To address these fundamental challenges, we propose a staged iterative monocular 3D object detector that transforms traditional serial architectures into a unified iterative

*Corresponding author.

framework. Through multistage iterative processing, our method achieves progressive 3D detection from coarse to fine. To fully exploit the advantages of this staged iterative structure, we design a Dynamic Target Iteration Module that adaptively optimizes queries at each iteration stage, enabling continuous refinement and enhancement of query features throughout the process. Furthermore, recognizing the critical role of depth features in accurate 3D localization, we develop an adaptively fused dual-branch depth estimator that processes global contextual perception and local geometric details in parallel to achieve more comprehensive depth feature representation.

The proposed method addresses the challenge of feature utilization in DETR-based models through fundamental architectural innovations. Our work demonstrates that iterative optimization principles can be effectively integrated into end-to-end detection frameworks, establishing a new paradigm to tackle the inherent challenges of monocular 3D understanding. We summarize the contributions as follows.

- We propose a novel architecture that decomposes the task into global and local branches with spatially-aware weight adaptation, dynamically providing optimal feature extraction strategies for diverse scene regions.
- An efficient iterative framework with staged structure that shortens encoder-decoder distances to improve feature utilization while accelerating convergence, addressing the low training efficiency of existing approaches.
- We present a target refinement module that concentrates attention on high-confidence regions to amplify target representations, significantly boosting perception capability in challenging scenarios.
- Extensive experiments on the KITTI test set demonstrate the superiority of our approach, achieving the AP_{3D} metric of 25.98%, 18.96%, and 16.38% for the car category. In particular, the gains are most pronounced for distant objects, which are typically more error-prone.

2. Related Work

The 3D object detection task aims to identify and localize objects in a 3D space from 2D images or 3D data, delivering reliable observations for downstream tasks. However, the inherent depth ambiguity of single RGB images makes accurate inference of 3D positions and geometric properties from a single viewpoint exceptionally challenging. Consequently, precise depth prediction emerges as the primary challenge in the detection of monocular 3D object.

Monocular 3D Object Detection. Due to the ill-posed nature of mapping single RGB images to 3D space, existing approaches mainly adopt strategies that extend 2D detection frameworks into 3D domains, forming two dominant technical paradigms: single-stage and two-stage architectures. In single-stage approaches, M3D-RPN[2] pioneered the anchor-based 3D detection paradigm, with subsequent works [23][13] enhancing it through asymmetric attention modules and differentiable non-maximum suppression. CenterNet[47] established the anchor-free framework, which inspired multiple advancements: [21][37][46] refined depth estimation, [24][49] optimized loss functions and camera extrinsic prediction, respectively, while [17] introduced a keypoint-based detection approach. In two-stage approaches, ROI-10D[25] successfully adapted Faster R-CNN's[32] design to 3D detection, spawning variants including decoupled 2D-3D detection losses [33] and first-stage orientation prediction [16]. To enhance monocular detection accuracy, MonoDepth[8] and DORN[6] leveraged pre-trained depth estimation networks to provide geometric cues, while [38] improved per-pixel depth estimation through foreground-background separation. Complementary approaches [15][44][1] effectively incorporated shape priors.

DETR-based Methods. DETR[4] pioneered the integration of Transformers into object detection as the first fully end-to-end framework, eliminating traditional NMS post-processing and anchor-based priors. In the 2D detection domain, several advancements address DETR's slow convergence and high computational complexity: Deformable DETR[50] introduces a deformable attention mechanism that constrains queries to focus only on critical sampling points within images; Conditional DETR[26] accelerates convergence through conditional spatial queries; Anchor DETR[39] reintroduces anchor queries to guide the Transformer's attention toward specific regional patterns. For 3D detection, MonoDTR[11] uses LiDAR point clouds as auxiliary supervision. MonoPGC[40] designs a depth-space-aware Transformer with depth-graded positional encoding to improve spatial reasoning. MonoDETR[45], as the first fully DETR-based monocular 3D detector, develops a depth-guided decoder with deformable attention layers for local feature aggregation. MonoDGP[31] advances this through an error-compensated decoupled visual decoder that predicts depth offsets to correct geometric errors. MonoVQD[36] introduces variational query denoising and look-ahead distillation with masked self-attention for refined query control. MonoDSSMs[35] introduced the first application of state-space models for efficient feature extraction and fusion in monocular 3D detection.

Depth Estimation in Monocular 3D Detector. Accurate depth prediction remains the fundamental challenge in monocular 3D detection, as it directly determines the precision of 3D location. GUPNet[22] addresses this by modeling depth uncertainty through Laplacian distributions, effectively handling inherent depth estimation biases and noise. DEVIANT[14] proposes a depth-equivariant network that uses depth-invariant constraints to improve prediction robustness. MonoDDE[18] reveals the limitations

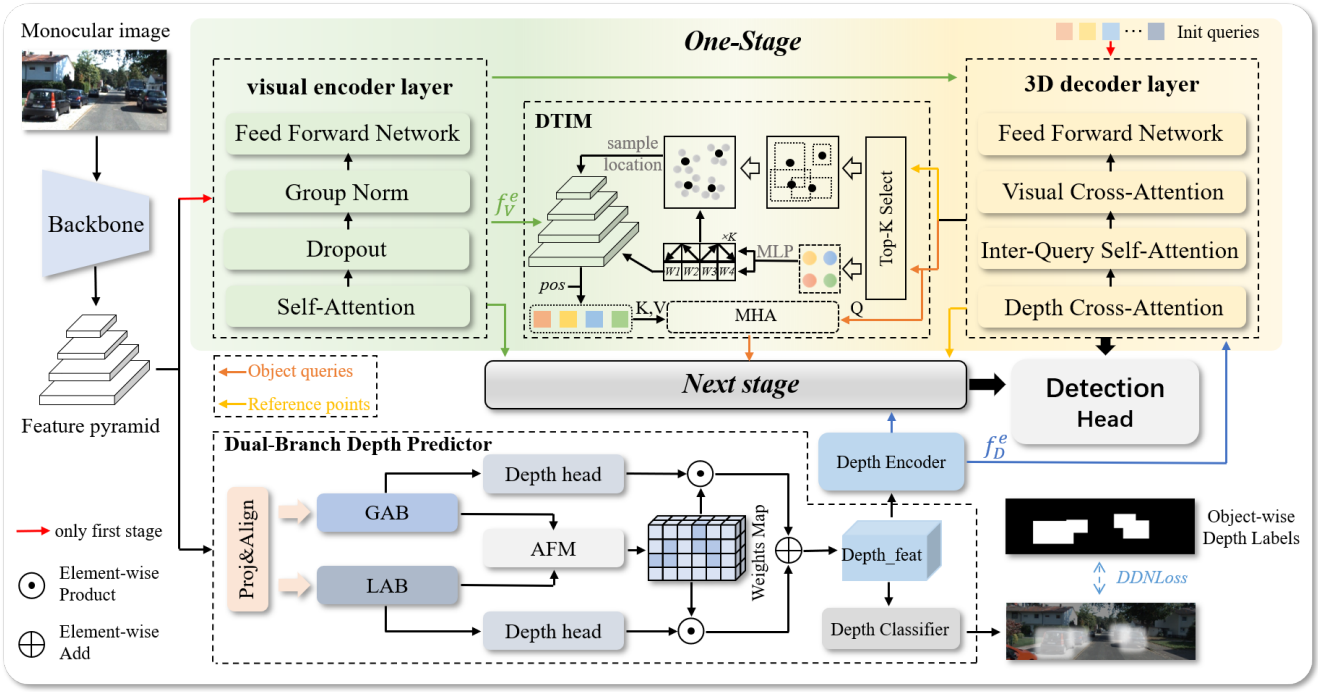

Figure 2. **Overall pipeline of MonoSISTR.** The network first distributes the image features to both the dual-branch depth estimator and the staged iterative structure. Initial queries interact bidirectionally with both visual and depth features within the decoder. These representations are subsequently refined through the DTIM before final 3D attribute prediction in the detection head.

of single-branch depth estimation through its multi-branch architecture with weighted fusion. MonoCD[43] introduces complementary depth estimation, significantly improving accuracy by combining global and local depth cues.

3. Methodology

Overview. As shown in Fig.2, we propose MonoSISTR for the detection of monocular 3D objects. Our approach incorporates the following key innovations: Firstly, we propose a dual-branch depth estimator that extracts complementary depth information through a local geometric details branch and a global context branch, incorporating an adaptive fusion module to generate branch weight maps for optimal depth feature combination. Furthermore, we introduce a staged iterative structure that maximizes feature utilization and stabilizes gradient propagation through multiple progressive refinement stages. Finally, we integrate a target refinement module at each stage, which dynamically focuses on potential target regions to enhance feature-query interactions. The implementation details of each component will be elaborated in the following sections.

3.1. Dual-Branch Depth Predictor

Effective depth feature perception is key to achieving superior performance in monocular 3D object detection networks. Our analysis reveals that the global net, with their expansive receptive fields, excels at capturing comprehensive depth information and spatial distribution patterns of targets. In contrast, local net leverage their high spatial resolution to precisely extract fine-grained depth features and boundary details. This inherent complementarity forms the theoretical foundation for our dual-branch depth estimator architecture, as illustrated in Fig.3.

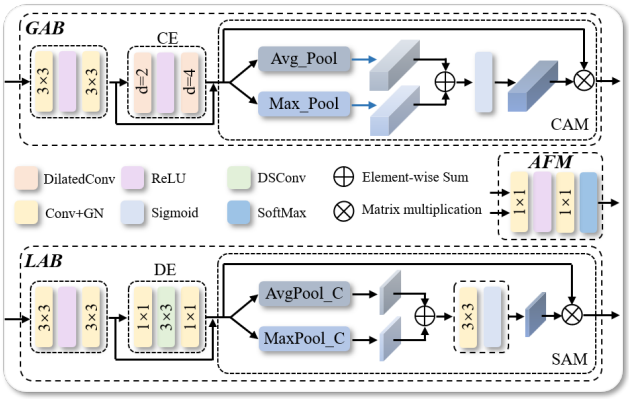

Figure 3. **Dual-Branch Depth Predictor.** Illustrating the proposed Global-Aware Branch (GAB), Local-Aware Branch (LAB), and the Adaptive Fusion Module (AFM).

Global-Aware Branch. Firstly, the network performs projection processing on multiscale features F^8, F^{16}, F^{32} from the backbone and uniformly aligns them to size $\frac{H}{16} \times$

$\frac{W}{16}$ as input for both branches. In this branch, we introduced a learnable parameter to attenuate features $F_{\downarrow}^{8}$ derived from high-resolution downsampling while emphasizing features $F_{\uparrow}^{32}$ obtained from low-resolution upsampling. The feature fusion obtained through different weights yields F_g:

$$F_g = Concat[\alpha F_{\downarrow}^{8}, F^{16}, (\frac{1}{\alpha})F_{\uparrow}^{32}] \tag{1}$$

where $\alpha \in (0, 1)$ is a learnable parameter. Second, we design a context enhancement module (CE) that captures multirange spatial dependencies and contextual features F_g^c by stacking dilated convolution layers with varying dilation rates. Next, we introduce the channel attention mechanism (CAM) to filter and enhance semantic information most relevant to depth estimation, helping the network understand spatial distribution of objects, formulated as:

$$W_{CAM} = \sigma(MLP(AP(F_g^c) + MP(F_g^c))) \tag{2}$$

$$F_g = F_g^c \otimes W_{CAM} \tag{3}$$

where σ denotes the sigmoid activation function, and W_{CAM} represents the channel attention weights. Ultimately, the weighted global features F_g are processed through the depth head module to generate the output F_g^d from the global perception branch.

Local-Aware Branch. In contrast to the global perception branch, we introduce a learnable parameter to suppress features $F_{\uparrow}^{32}$ derived from low-resolution upsampling while enhancing features $F_{\downarrow}^{8}$ obtained from high-resolution feature map downsampling. In this branch, we implement a detail enhancement module (DE) that incorporates separable convolutions in depth to refine local features to F_l^t while preserving fine-grained details. Then, the spatial attention mechanism(SAM) deployed learns a refined spatial weight distribution and applies the feature weighting to local features, producing the enhanced representation F_l,which is formulated as:

$$W_{SAM} = \sigma(Conv([AP(F_l^t), MP(F_l^t)])) \tag{4}$$

$$F_l = F_l^t \otimes W_{SAM} \tag{5}$$

where W_{SAM} represents the spatial attention weights. This design maximally preserves local detail information while maintaining computational efficiency. Before input into the depth head, residual connections are performed similarly to avoid performance degradation. The structure of the depth head remains consistent with the global branch, obtaining depth features F_l^d through two 3×3 convolutional layers.

Adaptive Fusion Module. This represents the core innovation of the dual-branch structure, which abandons simple addition or averaging strategies and instead learns an adaptive weight map to dynamically fuse the depth features output from both branches. In a specific implementation, the

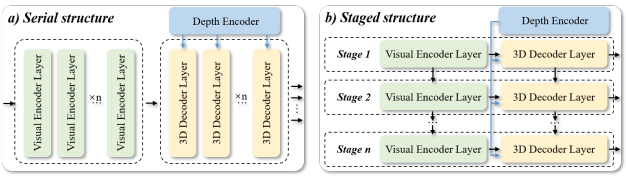

Figure 4. **Staged Iterative Structure.** a): The traditional serial structure[45]; b): Our proposed staged structure.

global features F_g and the local features F_l are first concatenated along the channel dimension, and weight maps W_f are obtained through the dynamic adaptive fusion module, with specific operations as follows:

$$W_f = DAFM(F_g, F_l) = [W_g, W_l] \in R^{\frac{H}{16} \times \frac{W}{16} \times 2} \tag{6}$$

where $W_g + W_l = 1$, W_g represents the weights for the global-aware branch, and W_l represents the weights for the local-aware branch. For the final prediction of depth features, we use the dual-branch local-global perception weights of the local branch to weight the output features and global branch output features, obtaining input features F_d for the depth encoder:

$$F_d = W_g \odot F_g + W_l \odot F_l \tag{7}$$

where $\odot$ denotes multiplication by element. This adaptive balancing strategy can more precisely handle the depth estimation challenges of multiscale targets in complex scenes, significantly improving the accuracy and robustness of depth prediction while maintaining computational efficiency.

3.2. Staged Iterative Structure

Traditional end-to-end detection models typically employ a serial structure - a strict 'encoder→decoder' unidirectional information flow pattern as shown in Fig.4 a). All encoder layers must process the visual features fully before passing them to the decoder. This design suffers from two fundamental limitations:

- Unidirectional information flow creates a bottleneck at the encoder's final output, making it difficult to maintain consistent high-confidence responses for distant targets with weak or incomplete features;
- The fixed feature-query interaction pattern in serial end-to-end decoders often leads to gradient instability and slow convergence during training.

As illustrated in Fig.4 b), we address these issues through our proposed staged iterative structure, which fundamentally transforms the encoder-decoder interaction paradigm. This innovative framework divides the entire encoding-decoding process into multiple iterative phases, each comprising a tightly coupled vision encoder and a 3D decoder forming an atomic iteration unit. Specifically, within each

stage: the visual encoder first enhances multiscale features through self-attention, followed by the decoder performing triple attention interactions - depth cross-attention, inter-query self-attention, and visual cross-attention - utilizing the enhanced visual features, depth embeddings, and current queries. The stage output feeds into the detection head for 3D property prediction, while the encoded visual features propagate to the next encoder layer. Meanwhile, the decoder's output queries and reference points advance to the next decoder layer.

This design enables queries to interact immediately with their corresponding visual features at each hierarchical level, achieving early information fusion and dynamic updates. Each iteration acts as a 'progressive refinement' of the preceding queries, realizing synergistic co-evolution of feature extraction and query optimization. The architecture maintains excellent configurability by adjusting iteration counts, optimal balances between computational cost and detection accuracy can be systematically achieved.

3.3. Dynamic Target Iteration Module

In the staged iterative structure, queries achieve more flexible target representation learning through multi-round interactions with both visual features and depth features. However, for low signal-to-noise ratio scenarios, the risk of feature dilution during deep encoding still exists. To address this, we design the dynamic target iteration module (DTIM), as illustrated in Fig.2.

Focus on key queries. At each iterative decoding stage, we design a dynamic Top-K selection-based query selection mechanism. Specifically, a lightweight linear classification head first evaluates all queries to obtain target confidence scores S_n:

$$S_n = \max_{c\in\{1,2,3\}} \sigma(W_{cls} \times q_n + b_{cls}) \quad \text{for } n = 1, 2, \ldots, N_q \tag{8}$$

where q_n denotes the query vector, W_{cls} is the classification weight matrix, and b_{cls} represents the bias vector. The confidence score reflects the probability that a query contains valid target information. Then, we need to obtain indices of target queries and reference points based on ranked confidence scores and a pre-set dynamic target filtering rate, which helps us obtain corresponding queries and reference points during the filtering stage. Subsequently, we select the K_{top} most promising queries Q^{key} and their associated reference points P^{key} from the full set of queries Q_N. This is achieved by ranking the queries based on their confidence scores and retaining only the Top-K candidates:

$$K_{top} = \text{TopK}(\{S_n\}_{n=1}^{N_q}, K) \tag{9}$$

where $K = \lfloor N_q \times \alpha \rfloor$, α is the dynamic filtering rate. After filtering, we obtain the key query set $Q^{key} = \{q_i | i \in K_{top}\}$ and the corresponding reference points $P^{key} = \{p_i | i \in K_{top}\}$. These reference points serve as centers for subsequent adaptive sampling, while filtered high-quality queries guide the dynamic generation of sampling points, ensuring sufficiently rich representations of target features.

Adaptive Multi-Scale feature sampling. Based on the selected target queries Q^{key} and reference points P^{key} from the previous step, we predict a fixed number M of offset vectors and scale weights for each reference point p_i from its corresponding query q_i:

$$[\Delta p_{i,1}, \ldots, \Delta p_{i,M}, W_{i,1}, \ldots, W_{i,L}] = \text{MLP}(q_i) \tag{10}$$

where $\Delta p_{i,j} \in \mathbb{R}^2$ is the relative offset and $W_{i,l}$ is the corresponding weight for the feature of the l-th layer. This query-driven offset prediction mechanism dynamically adjusts sampling locations according to the target's specific shape, pose, and scale. During the sampling phase, the final sampling coordinates are first computed based on the selected reference points and predicted offsets:

$$p_{i,j} = c_i + s_i \odot \tanh(\Delta p_{i,j}) \tag{11}$$

where $c_i = [p_i^x, p_i^y]$ is the reference center, $s_i = [p_i^w, p_i^h]$ is the scale factor, $\odot$ denotes element-wise multiplication, and the tanh function ensures that offsets remain within reasonable bounds. For multiscale encoded features $\{F^l\}_{l=0}^{L}$, we perform bilinear sampling on each scale to obtain sampled features $f_{i,j}^{(l)}$. To account for the different information characteristics on different feature scales, we introduce an adaptive scale weighting mechanism that generates learnable scale weights α from the predicted feature-level weights. This mechanism adaptively assigns varying importance to feature points from different scales:

$$\alpha^{(l)} = \text{Softmax}([W_{i,1}, \ldots, W_{i,L}]) \quad \text{for } i = 1, 2, \ldots, K \tag{12}$$

where l represents the index of the feature layer. Finally, we used scale weights to perform a weighted fusion of sampled features to obtain the final target-aware features.

$$f_{i,j} = \sum_{l=0}^{L} \alpha^{(l)} \cdot f_{i,j}^{(l)} \tag{13}$$

Adaptively sampled target features $f_{i,j}$ represent the most informative components of encoded features, providing both a purer signal representation for target queries and improved overall feature utilization. This mechanism proves particularly crucial for improving the detection accuracy of low-confidence responses, especially in challenging scenarios involving distant or occluded objects.

Dynamic Target Refinement. To fully exploit the advantages of our staged iterative structure, we design a dynamic target refinement mechanism based on object-focused attention. Using the target-aware features $f_{i,j}$ obtained through

Method	Reference	Extra data	Test, AP_{3D}			Test, AP_{BEV}			Val, AP_{3D}		
			Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
MonoDTR[11]	CVPR 22	LiDAR	21.99	15.39	12.73	28.59	20.38	17.14	24.52	18.57	15.51
DID-M3D[29]	ECCV 22		24.40	16.29	13.75	32.95	22.76	19.83	22.98	16.12	14.03
DD3Dv2[28]	ICRA 23		**26.36**	17.61	15.32	**35.70**	24.67	21.73	–	–	–
OccupancyM3D[30]	CVPR 24		25.55	17.02	14.79	35.38	24.18	21.37	26.87	19.96	17.15
MonoPGC[40]	ICRA 23	Depth	24.68	17.17	14.14	32.50	23.14	20.30	25.67	18.63	15.65
OPA-3D[34]	RAL 23		24.60	17.05	14.25	33.54	22.53	19.22	24.97	19.40	16.59
GUPNet[22]	ICCV 21	None	20.11	14.20	11.77	–	–	–	22.76	16.46	13.72
MonoDTR[11]	CVPR 22		21.99	15.39	12.73	28.59	20.38	17.14	24.52	18.57	15.51
MonoCon[19]	AAAI 22		22.50	16.46	13.95	31.12	22.10	19.00	26.33	19.01	15.98
DCD[18]	ECCV 22		23.81	15.90	13.21	32.55	21.50	18.25	–	–	–
DEVIANT[14]	ECCV 22		21.88	14.46	11.89	29.65	20.44	17.43	–	–	–
MonoDETR[45]	ICCV 23		25.00	16.47	13.58	33.60	22.11	18.60	28.84	20.61	16.38
DVDET[10]	RAL 23		23.19	15.44	13.07	32.05	22.15	19.32	–	–	–
MonoUNI	NeurIPS 23		24.75	16.73	13.49	–	–	–	24.66	17.18	14.06
MonoCD[43]	CVPR 24		25.53	16.59	14.53	33.41	22.81	19.57	26.45	19.37	16.38
FD3D[41]	AAAI 24		25.38	17.12	14.50	34.20	23.72	20.76	28.22	20.23	17.04
MonoDGP[31]	CVPR 25		26.35	18.72	15.97	35.24	25.23	22.02	**30.76**	22.34	19.02
Ours		None	25.98	**18.96**	**16.38**	35.15	**25.42**	**22.38**	30.28	**22.74**	**19.60**

Table 1. Comparison on the performance of the Car category on KITTI test and val set. For all results, we use $AP|R_{40}$ metric with IoU threshold equal to 0.7. We **bold** the best results and underline the second-best results.

the aforementioned adaptive sampling, we construct a specialized iterative refinement model using multi-head attention. Using the sampled features $f_{i,j}$ as key-value pairs (K, V) and taking the decoder output Q^{s-1} of the previous stage as the module input query to perform the target attention computation, formulated as:

$$K = V = f_{i,j} + pos\_embed \tag{14}$$

$$Q^s = Norm(Q^{s-1} + Dropout(MHA(Q^{s-1}, K, V))) \tag{15}$$

where pos_embed represents the sinusoidal positional encoding generated based on sampling coordinates to provide spatial positional information for the sampled features, and Q^s serves as input to the next stage decoder layer. Through the introduction of dropout regularization and residual connections, we effectively prevent overfitting risks and ensure training stability.

4. Experiments

4.1. Settings

Datasets. We evaluate the performance of our model on two authoritative benchmarks: KITTI[7] and nuScenes[3]. The KITTI dataset contains 7,481 training and 7,518 test images, with 3,769 images split from the training set for validation following standard protocols. It features three categories (Car, Pedestrian, Cyclist) with targets stratified into three difficulty levels (easy, moderate, hard) based on the height of the bounding box, the occlusion, and the truncation ratios, allowing for a comprehensive evaluation in challenging scenarios. The nuScenes dataset provides a comprehensive autonomous driving benchmark with multimodal data from six cameras, one LiDAR, and five radars offering 360° coverage, divided into 700 training, 150 validation, and 150 test scenes.

Method	IOU_{3d}	AP_{3D}		AP_{BEV}	
		Easy	Mod.	Easy	Mod.
MonoDETR[45]	0.7	9.53	8.19	16.39	14.41
MonoDGP[31]		**10.04**	8.78	16.55	14.53
Ours		9.86	**9.02**	**16.75**	**15.11**
MonoDETR[45]	0.5	31.81	28.35	35.70	31.96
MonoDGP[31]		29.56	26.17	32.67	29.44
Ours		**32.07**	**28.85**	**36.12**	**32.10**

Table 2. nuScenes val results for car. Our method achieves a certain improvement in IOU_{3d} thresholds at 0.5 and 0.7.

Evaluation Metrics. For the KITTI dataset, we adopt the standard evaluation metrics of 3D average precision (AP_{3D}) and Bird's Eye View average precision(AP_{BEV}) for the car category, with a 3D bounding box IoU threshold of 0.7 and precision calculations averaged across 40 recall positions. For nuScenes, we adopt the same metrics (AP_{3D} and AP_{BEV}) and evaluate the levels of easy and moderate difficulty.

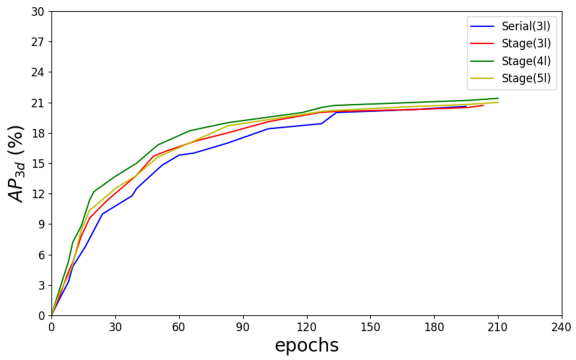

Figure 5. Comparative analysis of model convergence speed across different architectures and decoder layer counts.

4.2. Main Results

Comparisons on KITTI test set. In Tab.1, we compare the performance of our proposed method with three categories of approaches on the KITTI test set: LiDAR-assisted, depth-assisted, and extra-data-free methods.

Following [31], we similarly eliminate the contribution of the depth map in estimating the value of the center depth. Compared to LiDAR-assisted methods (DD3Dv2[28]), our method shows marginal decreases of -0.38%(AP_{3D}) and -0.55%(AP_{BEV}) on easy level targets. In particular, our method maintains robust high-confidence responses when processing valid but weak feature representations through staged iterative optimization, achieving consistent performance gains in our proposed methods: +0.24%(Mod.) and +0.41%(Hard) in AP_{3D}, along with +0.19%(Mod.) and +0.36%(Hard) in AP_{BEV}, and exceeds MonoDETR[45] in all metrics. In summary, these results definitively validate the efficacy of our method, particularly highlighting its advantage in handling challenging environmental conditions without relying on additional sensor input.

Comparisons on nuScenes val set. We further evaluate our method on the nuScenes val set. As presented in Tab.2, comparative results with other DETR-based methods in frontal view demonstrate that our approach achieves optimal performance at IoU threshold 0.5, with AP_{3D}/AP_{BEV} metrics reaching 32.07%/28.85% and 36.12%/32.10% respectively. At IoU 0.7, our method maintains superior performance in all metrics except AP_{3D} on Easy targets, confirming its advantage in complex scenarios. The slightly lower AP_{3D} (-0.18%) for easy targets at IoU 0.7 compared to MonoDGP[31] may stem from feature over-smoothing during iterative refinement, a hypothesis requiring dedicated experimental validation in future work.

4.3. Ablation Study

In the KITTI val set, we conducted extensive ablation studies using $AP_{3D}|R_{40}$(IOU=0.7) as the evaluation metric to analyze the individual components of our model optimization, with the results presented below:

Depth Predictor	Easy	Mod.	Hard
Baseline[45]	28.84	20.61	16.38
Global-Branch	28.83	20.66	16.55
Local-Branch	28.96	20.85	17.12
Dual-Branch	**29.02**	**20.94**	**17.16**

Table 3. Efficacy of the Dual-Branch design, showing the performance gains contributed by each branch to the network.

Dual-branch depth predictor. As shown in Tab.3, both the global and local branches individually provide marginal performance gains, while the adaptively fused dual-branch network contributes substantially to the model, achieving 20.94% and 17.16% accuracy on moderate and hard level targets, respectively. These results conclusively demonstrate the efficacy of the dual-branch architecture.

Enc-Dec.Arch	epochs to AP_{3d} of 15%	18%	20%	Mod.	Runtime
Serial(3l)	54	97	134	20.61	24ms
Stage(3l)	45	83	124	20.64	24ms
Stage(4l)	**39**	**63**	**112**	**21.43**	26ms
Stage(5l)	46	75	120	20.83	27ms

Table 4. The design of the Staged Iterative Structure. Impact of structure variants and stage counts on model convergence speed, where 'l' denotes the number of decoder layers.

Staged iterative structure. Results in Tab.4 demonstrate the superiority of our staged iterative structure. For the convergence speed evaluation, we record the minimum training epochs required to reach the average precision 15%, 18%, and 20% as key metrics, and each configuration was tested three times to obtain the median value. As evidenced by Fig.5, our structure achieves target accuracy levels with fewer epochs while maintaining equivalent precision and inference speed, confirming that the proposed architecture accelerates convergence without compromising performance.

Filtering rate	Easy	Mod.	Hard
0.2	28.89	22.18	18.27
0.4	**30.28**	**22.74**	**19.60**
0.6	29.84	22.58	19.13
0.8	29.16	22.07	18.21
w/o	29.34	21.89	18.43

Table 5. The design of dynamic target filtering rate. Impact of different filtering ratios on model performance. 'w/o' denotes the configuration without DTIM, while retaining both the dual-branch design and iterative structure.

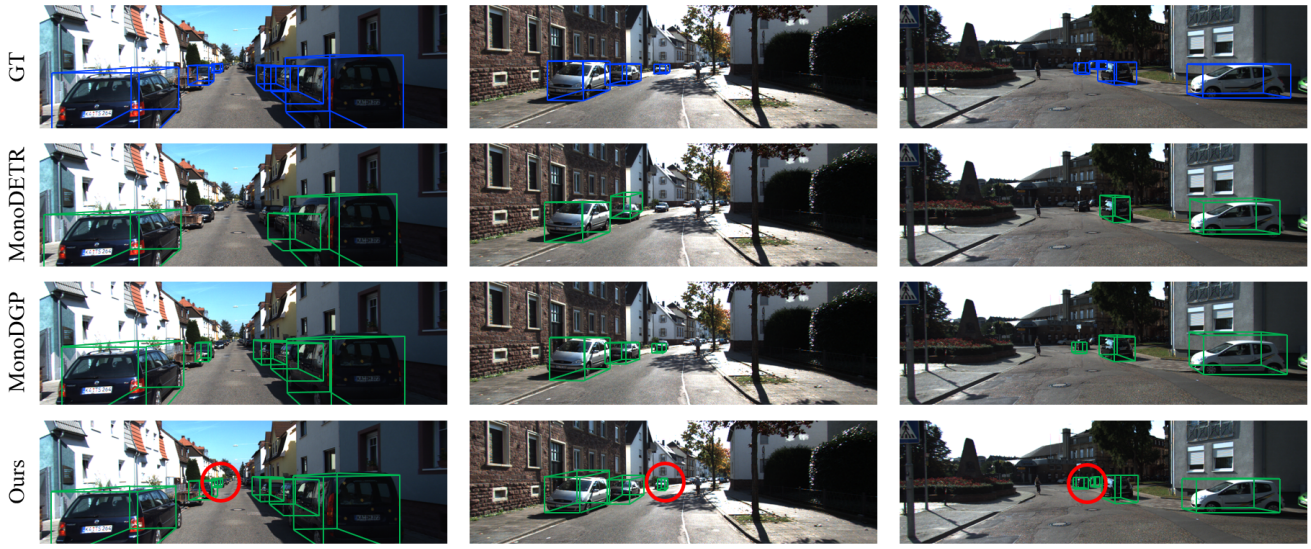

Figure 6. **Visualization of comparative results on the KITTI val set.** Blue boxes: ground truth annotations; Green boxes: detection results from different methods; Red circles: performance advantages of our method over competing approaches.

Dual.Bra	Staged.Str	DTIM	Easy	Mod.	Hard
-	-	-	28.84	20.61	16.38
✓	-	-	29.02	20.94	17.16
-	✓	-	29.11	21.43	18.22
-	✓	✓	29.54	22.38	19.18
✓	✓	✓	**30.28**	**22.74**	**19.60**
Improvement			+1.44	+2.13	+3.22

Table 6. The main ablation study. 'Dual.Bra' denotes the Dual-Branch Depth Predictor design, 'Staged.Str' represents the staged iterative structure.

Dynamic target filtering rate. To validate the impact of fixed reference point selection in our DTIM, we conduct comprehensive experiments with four filtering ratios (0.2, 0.4, 0.6 and 0.8) in five experimental configurations. Tab.5 demonstrates that peak performance occurs at a 0.4 filtering ratio (achieving highest average precision), while ratios of 0.2, 0.6 and 0.8 lead to degraded results. Our analysis reveals that lower ratios improve noise robustness at the cost of potential missed detections, whereas higher ratios preserve more target information but reduce feature interaction efficiency.

Overall performances. As shown in Tab.6, our comprehensive improvements - including dual-branch net, staged iterative structure and DTIM - collectively raise the baseline performance on the KITTI val set. The AD_{3d} metric demonstrates consistent gains of +1.44% (easy), +2.13% (moderate) and +3.22% (hard), conclusively validating the advantages of our proposed approach.

4.4. Qualitative Results

We present quantitative visualizations of our method's performance on the KITTI val set. As demonstrated in Fig.6, comparative analysis with two baseline approaches[45][31] reveals our method's superior capability in perceiving and localizing 3D attributes of distant small targets (marked by red circles) - particularly for objects that remain undetected by other methods. This empirically validates that our staged iterative structure successfully maintains high-confidence responses to weak target signals, significantly improving detection accuracy for challenging low-visibility targets.

5. Conclusion

In this paper, we point out that existing models improve performance by enhancing the prediction of target depth, but overlook the impact of the encoder-decoder architecture on the utilization of target features. To address this limitation, we present the staged iterative structure with target optimization mechanisms, coupled with a dual-branch depth estimator to enhance depth feature perception. Experimental results on the KITTI and nuScenes benchmarks demonstrate that our method achieves robust detection performance for challenging targets.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China(62222206, U25A20480, U2441241, 62272209, and 62401243), the Natural Science Foundation Youth Fund Program of Jiangxi Province (20242BAB212003), and the Jiangxi Province Natural Science Foundation (20242BAB20048, 20252BAC240010).

References

[1] Deniz Beker, Hiroharu Kato, Mihai Adrian Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Monocular differentiable rendering for self-supervised 3d object detection. In *European conference on computer vision*, pages 514–529. Springer, 2020. 2

[2] Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9287–9296, 2019. 2

[3] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 6

[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*, pages 213–229. Springer, 2020. 1, 2

[5] Yongjian Chen, Lei Tai, Kai Sun, and Mingyang Li. Monopair: Monocular 3d object detection using pairwise spatial relationships. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12093–12102, 2020. 1

[6] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2002–2011, 2018. 2

[7] 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*, pages 3354–3361. IEEE, 2012. 6

[8] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 270–279, 2017. 2

[9] Ente Guo, Zhifeng Chen, Susanto Rahardja, and Jingjing Yang. 3d detection and pose estimation of vehicle in cooperative vehicle infrastructure system. *IEEE Sensors Journal*, 21(19):21759–21771, 2021. 1

[10] Yue Hu, Shaoheng Fang, Weidi Xie, and Siheng Chen. Aerial monocular 3d object detection. *IEEE Robotics and Automation Letters*, 8(4):1959–1966, 2023. 6

[11] Kuan-Chih Huang, Tsung-Han Wu, Hung-Ting Su, and Winston H Hsu. Monodtr: Monocular 3d object detection with depth-aware transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4012–4021, 2022. 2, 6

[12] Jason Ku, Alex D Pon, and Steven L Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11867–11876, 2019. 1

[13] Abhinav Kumar, Garrick Brazil, and Xiaoming Liu. Groomed-nms: Grouped mathematically differentiable nms for monocular 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8973–8983, 2021. 2

[14] Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parchami, and Xiaoming Liu. Deviant: Depth equivariant network for monocular 3d object detection. In *European Conference on Computer Vision*, pages 664–683. Springer, 2022. 2, 6

[15] Abhijit Kundu, Yin Li, and James M Rehg. 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3559–3568, 2018. 2

[16] Buyu Li, Wanli Ouyang, Lu Sheng, Xingyu Zeng, and Xiaogang Wang. Gs3d: An efficient 3d object detection framework for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1019–1028, 2019. 2

[17] 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*, pages 644–660. Springer, 2020. 2

[18] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2791–2800, 2022. 2, 6

[19] Xianpeng Liu, Nan Xue, and Tianfu Wu. Learning auxiliary monocular contexts helps monocular 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1810–1818, 2022. 1, 6

[20] Yuxuan Liu, Yuan Yixuan, and Ming Liu. Ground-aware monocular 3d object detection for autonomous driving. *IEEE Robotics and Automation Letters*, 6(2):919–926, 2021. 1

[21] Zechen Liu, Zizhang Wu, and Roland Tóth. Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 996–997, 2020. 2

[22] 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*, pages 3111–3121, 2021. 2, 6

[23] Shujie Luo, Hang Dai, Ling Shao, and Yong Ding. M3dssd: Monocular 3d single stage object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6145–6154, 2021. 2

[24] Yuexin Ma, Xinge Zhu, Sibo Zhang, Ruigang Yang, Wenping Wang, and Dinesh Manocha. Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6120–6127, 2019. 2

[25] Fabian Manhardt, Wadim Kehl, and Adrien Gaidon. Roi-10d: Monocular lifting of 2d detection to 6d pose and met-

ric shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2069–2078, 2019. 2

[26] 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*, pages 3651–3660, 2021. 2

[27] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learning and geometry. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 7074–7082, 2017. 1

[28] Dennis Park, Jie Li, Dian Chen, Vitor Guizilini, and Adrien Gaidon. Depth is all you need for monocular 3d detection. *arXiv preprint arXiv:2210.02493*, 2022. 6, 7

[29] Liang Peng, Xiaopei Wu, Zheng Yang, Haifeng Liu, and Deng Cai. Did-m3d: Decoupling instance depth for monocular 3d object detection. In *European Conference on Computer Vision*, pages 71–88. Springer, 2022. 6

[30] Liang Peng, Junkai Xu, Haoran Cheng, Zheng Yang, Xiaopei Wu, Wei Qian, Wenxiao Wang, Boxi Wu, and Deng Cai. Learning occupancy for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10281–10292, 2024. 6

[31] Fanqi Pu, Yifan Wang, Jiru Deng, and Wenming Yang. Monodgp: Monocular 3d object detection with decoupled-query and geometry-error priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6520–6530, 2025. 2, 6, 7, 8

[32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016. 2

[33] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1991–1999, 2019. 2

[34] Yongzhi Su, Yan Di, Guangyao Zhai, Fabian Manhardt, Jason Rambach, Benjamin Busam, Didier Stricker, and Federico Tombari. Opa-3d: Occlusion-aware pixel-wise aggregation for monocular 3d object detection. *IEEE Robotics and Automation Letters*, 8(3):1327–1334, 2023. 6

[35] Kiet Dang Vu, Trung Thai Tran, and Duc Dung Nguyen. Monodssms: Efficient monocular 3d object detection with depth-aware state space models. In *Proceedings of the Asian Conference on Computer Vision*, pages 3883–3900, 2024. 2

[36] Kiet Dang Vu, Trung Thai Tran, and Duc Dung Nguyen. Monovqd: Monocular 3d object detection with variational query denoising and self-distillation. *arXiv preprint arXiv:2506.14835*, 2025. 2

[37] Tai Wang, ZHU Xinge, Jiangmiao Pang, and Dahua Lin. Probabilistic and geometric depth: Detecting objects in perspective. In *Conference on Robot Learning*, pages 1475–1485. PMLR, 2022. 2

[38] Xinlong Wang, Wei Yin, Tao Kong, Yuning Jiang, Lei Li, and Chunhua Shen. Task-aware monocular depth estimation for 3d object detection. In *Proceedings of the AAAI conference on artificial intelligence*, pages 12257–12264, 2020. 2

[39] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based detector. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2567–2575, 2022. 2

[40] Zizhang Wu, Yuanzhu Gan, Lei Wang, Guilian Chen, and Jian Pu. Monopgc: Monocular 3d object detection with pixel geometry contexts. *arXiv preprint arXiv:2302.10549*, 2023. 2, 6

[41] Zizhang Wu, Yuanzhu Gan, Yunzhe Wu, Ruihao Wang, Xiaoquan Wang, and Jian Pu. Fd3d: Exploiting foreground depth map for feature-supervised monocular 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6189–6197, 2024. 6

[42] Chen Yan and Emre Salman. Mono3d: Open source cell library for monolithic 3-d integrated circuits. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(3): 1075–1085, 2017. 1

[43] Longfei Yan, Pei Yan, Shengzhou Xiong, Xuanyu Xiang, and Yihua Tan. Monocd: Monocular 3d object detection with complementary depths. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10248–10257, 2024. 1, 3, 6

[44] Sergey Zakharov, Wadim Kehl, Arjun Bhargava, and Adrien Gaidon. Autolabeling 3d objects with differentiable rendering of sdf shape priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12224–12233, 2020. 2

[45] Renrui Zhang, Han Qiu, Tai Wang, Ziyu Guo, Ziteng Cui, Yu Qiao, Hongsheng Li, and Peng Gao. Monodetr: Depth-guided transformer for monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9155–9166, 2023. 1, 2, 4, 6, 7, 8

[46] 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*, pages 3289–3298, 2021. 1, 2

[47] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 2

[48] Xichuan Zhou, Yicong Peng, Chunqiao Long, Fengbo Ren, and Cong Shi. Monet3d: Towards accurate monocular 3d object localization in real time. In *International conference on machine learning*, pages 11503–11512. PMLR, 2020. 1

[49] Yunsong Zhou, Yuan He, Hongzi Zhu, Cheng Wang, Hongyang Li, and Qinhong Jiang. Monocular 3d object detection: An extrinsic parameter free approach. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7556–7566, 2021. 2

[50] 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. 2