

BEV-CAR: Enhancing Monocular Bird’s Eye View Segmentation with Context-Aware Rasterization

Yixin Xiong¹, Ke Wang², Tongtong Cheng¹, Chunhui Liu^{*3}, and Kai Liu^{*1}

¹College of Computer Science, Chongqing University, China

²National Elite Institute of Engineering, Chongqing University, China

³The Hong Kong Polytechnic University, China

Abstract

Bird’s Eye View (BEV) semantic segmentation is essential for autonomous driving and mobile robotics, yet it still faces significant challenges on accurate segmentation of foreground object and efficient estimation of layout categories obscured by objects. To address these issues, we propose BEV-CAR, a Context-Aware Rasterization method that rasterizes the BEV representation without any coordinate transformations. By optimising each ray and incorporating depth features, BEV-CAR effectively addresses the challenges posed by object occlusions and varying environmental conditions. It ensures robust performance across diverse scenarios, particularly improving the accuracy of foreground object segmentation and layout estimation in occluded areas. And extensive experiments on the nuScenes and Argoverse datasets demonstrate that BEV-CAR achieves state-of-the-art (SOTA) performance. More importantly, the rasterization technique in this paper does not introduce additional computational overhead during the inference process, making it suitable for practical deployment in real-world scenarios. Code is available at github.com/BEV-DAR/BEV-DAR.

1. Introduction

With the development of autonomous driving and mobile robot, BEV perception has emerged as a critical component of scenario perception [9, 11, 16]. BEV provides a top-down perspective that is essential for understanding the spatial layout of the environment, facilitating tasks such as path planning [6], obstacle detection, and map segmentation [3]. In particular, map segmentation plays a crucial role in accurately identifying different elements in the envi-

ronment. In industry, obtaining accurate semantic maps in the BEV often rely on expensive sensors such as lidar and radar [22, 34]. However, these methods are computationally demanding and require real-time inference on point cloud data. To address these limitations and reduce the reliance on lidar, many researchers are focusing on the use of monocular images to build BEV segmentation maps [3, 10, 17].

Most existing BEV segmentation methods [17, 18, 20, 31] employ a transformer-based encoder-decoder structure to capture global context and obtain comprehensive feature representations. Then, a decoding network similar to the generator in generative adversarial networks (GAN) [8] is adopted to predict BEV segmentation maps. While these approaches can overcome the limitations imposed by the flat-world assumption, it may lead to the decreasing of segmentation accuracy due to the lack of geometric prior features. To address this issue, some researchers derive geometric prior from intrinsic and extrinsic camera parameters and transform the feature map into BEV features, thus predicting the surrounding environment [5, 7, 24]. These methods are effective in handling perspective feature transformation in flat regions. However, they are not sufficient in predicting regions that are occluded by foreground objects. Since these regions are not visible, using prior features for viewpoint transformation can lead to objects being distributed over abnormally large regions on the BEV segmentation map. This enlargement can cause distortion. It may also result in the absence of content. This discrepancy in effectiveness highlights the need for a more robust approach that can handle both the complexities of cross-view transformations and the distortions caused by occlusions.

Several studies have focused on exploring alternative coordinate systems to address this challenge. With polar coordinate transformation, the BEV sensing area can be viewed as a circle with the camera’s optical center as the center and the sensing distance as the radius. This gives models the potential to perceive complex scenes. For instance, certain approaches transform BEV features from the Cartesian coordinate system to a polar coordinate system [12, 15, 26, 32],

*Corresponding authors: Chunhui Liu and Kai Liu

This work was supported by the National Natural Science Foundation of China under Grant No. 62472055, and Chongqing Key Laboratory of Big Data Intelligence and Privacy Computing.

and such transformation can simplify the processing of radial data and increase model interpretability. However, these methods will unavoidably duplicate the sampling of the nearer points, which leads to an inability to accurately segment the occluded invisible regions at the far end. Moreover, the transformation of the coordinates will introduce extra computation overhead.

To enhance spatial reasoning and occlusion awareness in BEV perception, We propose BEV Context-Aware Rasterisation (BEV-CAR), a framework that enhances occlusion reasoning and spatial awareness in BEV perception without relying on coordinate transformations. BEV-CAR extends the global encoder with a depth-aware branch to capture depth-related features, which are fused with the main BEV representation to improve depth sensitivity, particularly for foreground objects. During training, a context-aware rasterisation mechanism structures the decoder output along radial rays using a ray-wise 1D representation in Cartesian space, enabling the model to distinguish visible and occluded regions along the line of sight through a dedicated raster loss. Crucially, this mechanism is only active during training and is removed at inference, introducing no additional computational overhead while avoiding the over-sampling and distortion issues inherent in polar or other transformed coordinate systems.

To the best of our knowledge, this is the first attempt to address the occlusion problem in monocular BEV segmentation using context-aware rasterization. Our contributions are summarized as follows:

- We introduce a global BEV branch that fuses multi-layer features to generate distance-adaptive representations in BEV feature, addressing the limitations of traditional lift-splat methods that rely on fixed-depth projection. This enhancement improves the accuracy of the model on predicting the depth of foreground objects compared with traditional monocular BEV segmentation algorithms.
- We propose a plug-and-play context-aware rasterisation mechanism with a dedicated raster loss function. By structuring decoder outputs into depth-indexed, ray-wise 1D vectors and optimizing each ray individually, the mechanism enhances model reasoning in complex scenes such as overlapping and occluded regions. Importantly, this approach exhibits strong generalizability by being integrable into BEV segmentation methods.
- We propose a novel framework, BEV-CAR, which fuses global and depth BEV features in encoder and incorporates a context-aware raster mechanism after decoder. Extensive experiments on nuScenes and Argoverse datasets show that BEV-CAR achieves SOTA performance in layout and vehicle occupancy estimation.

2. Related Work

We divided BEV segmentation methods into two categories based on the underlying coordinate systems used. Our work utilizes Cartesian coordinate system to store the rays of polar coordinate system. Therefore, distinguishing between methods based on a Cartesian coordinate system [23, 25] and those based on a polar coordinate system [15, 26], provides a clearer understanding of the research and facilitates a more informed comparison.

2.1. Cartesian-based BEV Segmentation

Influenced by 2D image semantic segmentation, early researchers [13, 17, 27] followed the pipeline of 2D image semantic segmentation to fit the correspondence between front view(FV) image and BEV coordinates. Lu [17] employs a variational encoder-decoder (VED) framework that learns to predict a top-view semantic-metric occupancy grid map, but using this approach directly does not work well for the cross view task. Because the image pixels do not correspond to the BEV segmentation map one-to-one, the pixels are subject to information confusion. Thus, Mono-Layout [18], PYVA [31], and VPN [20] have drawn inspiration from generative learning and proposed generator-like decoder structures to produce BEV segmentation results directly. However, these methods tend to result in object positions being predicted too closely together due to a lack of geometric a priori knowledge. Consequently, some works [7, 21, 35, 36] have introduced geometric priors, utilizing homography matrices to project FV images into the BEV perspective. These methods can easily handle perspective feature transformations in flat areas. However, it is hard to achieve accurate prediction for regions that are occluded.

2.2. Polar-based BEV Segmentation

Research on Polar-based BEV Segmentation has only begun to emerge in the last few years. Compared to the Cartesian coordinate system, polar coordinates are better able to represent depth features because they are closer to the real viewpoint. Thus, to improve depth representation, many polar-based methods [12, 15, 30, 32] are proposed. Specifically, within the realm of segmentation, several approaches leverage this advantage to enhance the efficiency of depth perception. Gitnet [7] corresponds the columns in the FV image to the polar lines in the BEV view, and propose a ray-based transformer. TIM [26] proposes an inter-plane attention mechanism to better represent polar information. TADE [33] transforms the BEV label map from Cartesian to polar coordinates to learn better alignment across perspective views and BEV. PCBEV[22] converted the image first to a polar coordinate system and then to a Cartesian coordinate system and fused the two features. However, transformation to polar coordinates entails a certain amount of computation, and secondly, direct transformation to polar

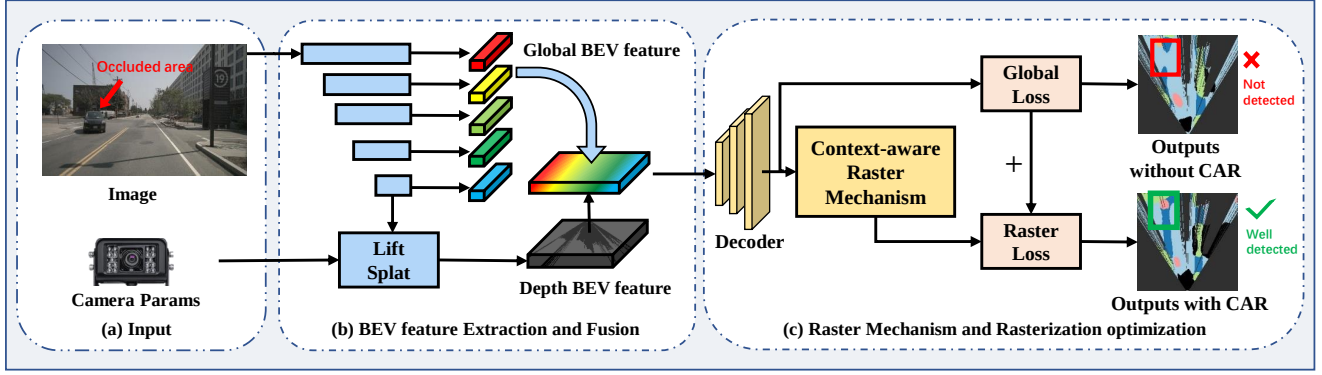


Figure 1. The overview of BEV-CAR. (a) Input Image and camera parameter. (b) A multi-scale global BEV feature and a depth-aware BEV feature are extracted in parallel and then fused to form a more complete and robust BEV representation. (c) The decoder output is rasterized and rays are sampled on the rasterized map. Subsequently, a composite loss function is computed as a weighted sum of the raster loss and the global loss to optimize the performance.

coordinates on the Cartesian coordinate system result in the pixels at the near point being repeatedly sampled multiple times.

In summary, Cartesian-based methods offer deployment-friendly outputs but lack explicit geometric supervision along viewing rays, while polar-based approaches better model depth structure at the cost of computational overhead and representation incompatibility. BEV-CAR bridges this gap: it retains a standard Cartesian BEV representation at inference for full compatibility and zero latency, yet enforces ray-wise semantic consistency during training via polar-sampled rays.

3. Method

The structure of BEV-CAR is shown in Figure. 1, the BEV features are composed of global BEV features and depth BEV features complementary to each other. After that, the initial BEV segmentation map is obtained by using the unified decoder, and then the BEV segmentation map is improved by the raster optimization technique to generate the final BEV segmentation result.

3.1. BEV Feature Extraction and Fusion

Our BEV feature extraction and fusion module designs a bifurcation encoder to generate BEV feature, which is jointly modeled by exploiting implicit depth and global information. The depth BEV features are generated using the lift-splat method from LSS [21], which maps image features into BEV space based on implicit depth prediction. However, this approach struggles to produce precise object boundaries and capture long-range spatial relationships.

To better preserve object boundaries and global context, multi-scale features are fused in a structured way. High-resolution feature maps are assigned to model distant regions, while low-resolution maps handle nearby areas. This

is because distant objects occupy fewer pixels and require more precise features. Each feature level is transformed into BEV using densenettransformer, resulting in scale-specific BEV features $G_k \in \mathbb{R}^{C_k \times Z_k \times X_k}$. These features are then combined through spatially-aligned concatenation:

$$G_{\text{BEV}} = \text{Concat}_{k=0}^K G_k \quad (1)$$

The integration of multi-scale global features with implicit depth feature enables the learning of richer and more robust BEV feature representations.

3.2. Context-aware Raster Mechanism

Conventional BEV representations model the scene as a uniform Cartesian grid, which ignores the anisotropic resolution and occlusion structure inherent in perspective imaging. To better align the representation with the physical observation process, we propose to conceptualize the BEV plane as a set of discrete rays emanating from the ego-vehicle in polar coordinates. Along each ray, points are sampled at increasing depths, forming a 1D sequence where the index implicitly encodes relative distance. We then model contextual interactions locally within each ray, enabling the network to learn geometric priors such as depth continuity, occlusion transitions, and object boundary coherence without resorting to expensive global attention or dense neighborhood aggregation.

To efficiently instantiate these rays as connected pixel sequences on the discrete BEV grid, we employ the Bresenham line algorithm [1], which guarantees topologically continuous and gap-free rasterization. This yields an ordered set of spatial locations per ray, perfectly suited for sequential feature refinement along sightlines. We define each ray by its origin at the camera’s optical center (x_0, y_0) and its endpoint (x_1, y_1) , obtained by intersecting the viewing direction with the image boundary according to the hor-

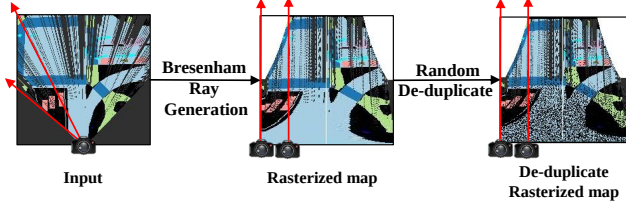


Figure 2. The illustration of context-aware raster mechanism, where rays generated by the Bresenham-Ray Generator are rearranged into a raster map. This map is then randomly de-duplicated to remove some resampled points, preventing the model from focusing excessively on near-point objects.

horizontal field of view. To rasterize this ray into a discrete sequence of BEV grid cells, we employ the Bresenham line algorithm [1]. Assuming $0 \leq m \leq 1$ and $x_0 < x_1$ (other octants are handled via symmetry), the algorithm uses a decision variable d , initialized as

$$d = 2(y_1 - y_0) - (x_1 - x_0), \quad (2)$$

and updated iteratively as follows:

$$(x, y)_{\text{next}} = \begin{cases} (x + 1, y), & \text{if } d \leq 0, \\ (x + 1, y + 1), & \text{if } d > 0, \end{cases} \quad (3)$$

$$d_{\text{next}} = \begin{cases} d + 2(y_1 - y_0), & \text{if } d \leq 0, \\ d + 2(y_1 - y_0) - 2(x_1 - x_0), & \text{if } d > 0. \end{cases} \quad (4)$$

This produces a gap-free, 8-connected pixel path that preserves the depth ordering along the sightline. The resulting sequence, in which each index corresponds to an increasing depth, serves as the structural basis for local contextual modeling and enables the network to capture geometric cues such as depth continuity and occlusion transitions.

Figure 2 illustrates the context-aware raster mechanism, where red rays are utilized to represent camera rays. By transforming the BEV map into line segments and rearranging them, we can see that the proximal points are resampled multiple times, which helps the model to more accurately segment the region of the proximal points, but it also makes the model less capable of segmenting the distal objects. It is worth noting that since we are storing the Cartesian coordinates of each line segment in pixels, it is straightforward to use a de-duplication operation similar to deletion to minimize some of the resampling, thus ensuring that the model performs well at both the proximal and distal ends.

3.3. Raster Optimization Strategy

Figure 3 shows the optimization process of the raster optimization strategy. Our loss function is divided into two parts: the raster loss function and the global loss function.

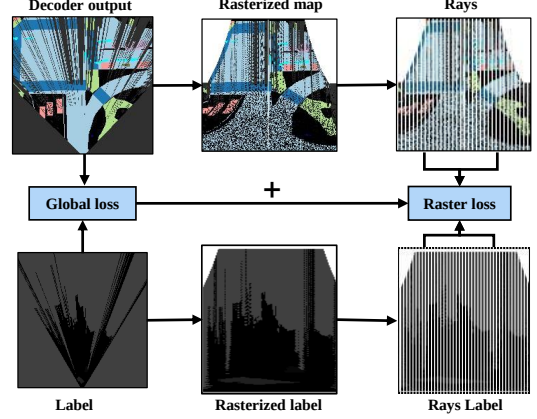


Figure 3. Illustration of raster optimization strategy. The raster loss function is computed for each ray of the rasterised logits, and later merged with the loss function computed directly from the global logits.

The raster loss function is computed directly for each ray to refine the prediction by addressing discrepancies in depth perception. To improve optimization efficiency and robustness, we randomly drop duplicate points along each ray, akin to dropout.

Random De-duplication. Given a segmentation map $\mathbf{P} \in \mathbb{R}^{B \times C \times W \times H}$, label $\mathbf{T} \in \mathbb{R}^{B \times W \times H}$, and a set of rays $\mathcal{R}$, for each ray $r \in \mathcal{R}$:

For each point $p = (p_x, p_y)$ on the ray r :

$$\mathbf{l}_r \leftarrow \mathbf{L}[:, :, p_y, p_x], \mathbf{t}_r \leftarrow \mathbf{T}[:, :, p_y, p_x] \quad (5)$$

Before computing the raster loss, we perform random removal of duplicate points along each ray. This is achieved by introducing a dropout-like [28] mechanism where points that are found to be duplicates are randomly removed with a certain probability p_{drop} . This helps in reducing redundancy and potentially improving the convergence of the training process, and we set $p_{\text{drop}} = 0.5$ in this paper. After the de-duplication, raster loss is calculated.

Raster Loss. The raster loss function for a single ray $\mathbf{l}_r$ can be denoted as:

$$L_r(\mathbf{l}_r, \mathbf{t}_r) = - \sum_{r=1}^N [\mathbf{t}_r \log(\mathbf{l}_r) + (1 - \mathbf{t}_r) \log(1 - \mathbf{l}_r)] \quad (6)$$

By incorporating this random de-duplicated step, we ensure that the optimization process focuses more on unique points, thereby enhancing the model's ability to generalize and reduce overfitting to redundant points. And the loss function for multiple rays is denoted as:

$$L_{\text{rasterloss}}(\mathbf{l}_r, \mathbf{t}_r) = \frac{\sum_{r=1}^k \text{top}_k(L(\mathbf{l}_r, \mathbf{t}_r))}{k} \quad (7)$$

where we adopt the TOP-K loss function to enhance model optimization by focusing on regions with significant prediction errors. The TOP-K strategy can be seen as online hard sample mining within a limited number of rays, enabling the model to effectively address challenges such as occlusion, lighting variations, and excessive distances.

Global Loss. Based on the previous work [24], we adopt the cross entropy as the global loss function, the L_{global} is shown below.

$$\mathcal{L}_{global} = \frac{1}{N} \left[- \sum_i^N w_i y_i \log c_i - \sum_i^{N_{neg}} (1 - w_i) (1 - y_i) \log(1 - c_i) \right] \quad (8)$$

where we use weight w_i to balance the uneven distribution of categories, and for clarity, c_i represents the predicted probability of the category outcome, while w_i signifies the true label for the corresponding instance.

Optimization Strategy. Combining the aforementioned two parts, the whole loss function can be represented as

$$Loss = \lambda L_{raster} + L_{global} \quad (9)$$

where λ is a hyperparameter used to balance raster loss with global loss.

4. Experiment

4.1. Experimental Setup

Datasets. To evaluate our methodology, we conducted experiments on two public datasets: the nuScenes dataset and the Argoverse dataset. These datasets provide rich and diverse data for evaluating our approach in real-world scenarios. The nuScenes dataset comprises 1,000 scenes captured by six cameras in Boston and Singapore. Its detailed lidar segmentation annotations enable easy acquisition of the ground truth for monocular BEV segmentation. The Argoverse dataset includes 65 training sequences and 24 validation sequences recorded in Miami and Pittsburgh. Similar to the nuScenes dataset, Argoverse provides 3D object annotations across 15 categories. In order to verify the fairness of the experimental comparisons, many methods [7, 33] use the data partitioning method provided by PON [24] to split the training and test sets, so our work follows this method, and we chose to perform the ablation experiments on the nuScenes dataset due to the greater diversity.

Implementation Details. We fix BEV perspective on a grid 50 meters in front of the given camera, with a lateral coverage of 25 meters on each side and a resolution of 25 cm per pixel, and end up with a 200×198 groundtruth, which is consistent with other methods. The model is

trained using one A100 card. We optimize the network using the Adam strategy. The initial learning rate is set to 0.00018, weight decay to 0.01, and batch size to 12. The λ in raster loss is set to 10. We use the intersection over union (IoU) metric across different background categories to showcase our results. Additionally, we compute the mean IoU across categories to evaluate overall performance. During the assessment process, we disregard grid cells that are not visible due to the inability of lidar beams to penetrate these areas, for which groundtruth is unavailable.

4.2. Quantitative Comparison

We conducted experiments utilizing all available surround cameras on nuScenes [2] and Argoverse [4] datasets. We first compared our method with classic methods [20, 24, 35], and then with SOTA methods from the recent two years [14, 19, 26, 29, 33, 36]. The results on the nuScenes dataset are presented in Table 1, while those on the Argoverse dataset are shown in Table 2.

NuScenes. Table 1 presents the performance of BEV-CAR and other SOTA methods on the nuScenes dataset. It shows that our BEV-CAR method achieves the best overall performance in terms of average IoU across all classes, with a score of 31.5%, surpassing recent SOTA approaches.

In particular, BEV-CAR performs exceptionally well on layout classes, achieving IoUs of 70.9% for drivable areas, 48.3% for pedestrian crossings, 49.2% for walkways, and 34.4% for car parks, demonstrating its strong capability in modeling large-scale scene structures. For object segmentation, BEV-CAR also delivers competitive results, especially for challenging categories such as trailers (29.1%) and trucks (29.7%), which are often underrepresented in semantic segmentation benchmarks. Additionally, it achieves the highest IoU for buses (38.9%), showing strong detection performance for large objects. While BEV-CAR does not achieve the top performance for construction vehicles, barriers, motorcycles, or traffic cones, its consistently strong and balanced performance across different object and layout categories demonstrates its effectiveness and robustness in complex autonomous driving scenarios. **Argoverse.** Table 2 presents the performance metrics of BEV-CAR alongside other algorithms on the Argoverse validation set. BEV-CAR achieves a mean IoU of 29.9%, outperforming all previous methods and setting a new state-of-the-art result. It demonstrates particularly strong performance in drivable area segmentation (81.2%) and vehicle segmentation (43.1%), indicating its effectiveness in capturing both road structure and object-level details. While BEV-CAR does not achieve the highest IoU for large vehicles (15.9%) or motorcycles (3.3%), it performs competitively across other categories. Notably, it significantly improves pedestrian (9.3%) and bike (12.9%) segmentation, highlighting its robustness in small objects.

Method	Lay Out				Object										Mean
	Drivable	Crossing	Walkway	Carpark	Bus	Bike	Car	C.V.	Motor	Trailer	Truck	Ped.	Cone	Barrier	
VPN [20]	58.0	27.3	29.4	12.9	20.0	4.4	25.5	4.9	5.6	16.6	17.3	7.1	4.6	10.8	17.5
PON [24]	60.4	28.0	31.0	18.4	20.8	9.4	24.7	12.3	7.0	16.6	16.8	8.2	5.7	8.1	19.1
HFT [35]	55.9	35.6	35.4	23.2	30.6	19.1	27.3	18.6	5.8	7.9	11.1	11.7	5.1	11.4	21.3
TIM [26]	<u>74.5</u>	36.6	35.9	31.3	32.8	14.7	39.7	14.2	7.6	13.9	26.3	9.5	7.6	14.7	25.7
DiffBEV [36]	65.4	41.3	41.1	28.4	33.7	13.2	38.9	8.4	14.4	21.1	23.1	9.6	7.5	16.7	25.9
GitNet [7]	65.1	41.6	42.1	<u>31.9</u>	35.4	13.8	43.4	9.7	15.0	22.5	25.5	14.1	11.6	18.6	27.9
FTVP [14]	75.7	34.0	39.2	30.5	31.9	11.3	34.4	0.8	10.4	17.2	20.4	6.3	6.2	16.5	23.9
TaDe [33]	65.9	40.9	<u>42.3</u>	30.7	<u>38.5</u>	14.2	42.8	11.4	18.2	26.3	26.9	14.0	<u>15.0</u>	21.4	29.2
MoBEV [19]	59.8	30.1	32.3	24.4	35.5	14.8	49.3	9.4	17.2	27.6	<u>28.9</u>	8.1	12.2	11.7	25.8
OEBEV [29]	68.2	<u>41.7</u>	36.9	29.7	31.2	<u>18.6</u>	<u>45.2</u>	<u>18.4</u>	<u>16.5</u>	<u>28.3</u>	25.4	10.8	16.1	18.4	<u>29.3</u>
BEV-CAR(ours)	70.9	48.3	49.2	34.4	38.9	17.9	42.5	17.1	<u>17.7</u>	29.1	29.7	14.1	13.1	<u>18.7</u>	31.5

Table 1. Quantitative results of IoU (%) on the nuScenes [2] validation set. Best scores are boldfaced and the second-best are underlined. “Drivable”, “C.V.”, “Motor.”, “Ped.”, and “Cone” denote drivable area, construction vehicle, motorcycle, pedestrian, and traffic cone.

Method	Drivable	Vehicle	Ped.	L.V.	Bike	Bus	Trailer	Motor.	Mean
VPN [20]	64.9	23.9	6.2	9.7	0.9	3.0	0.4	1.9	13.9
PON [24]	65.4	31.4	7.4	11.1	3.6	11.0	0.7	5.7	17.0
TADE [33]	68.3	34.5	9.3	14.7	4.4	37.8	3.1	<u>6.4</u>	22.3
TIM [26]	75.9	35.8	5.7	14.9	3.7	30.2	12.2	2.6	22.6
FTVP [14]	83.5	37.5	4.9	12.0	4.1	0.8	60.9	3.1	25.9
HFT [35]	79.0	<u>38.8</u>	<u>9.0</u>	23.1	<u>7.7</u>	<u>34.7</u>	38.1	7.5	<u>29.7</u>
BEV-CAR(ours)	<u>81.2</u>	43.1	9.3	<u>15.9</u>	12.9	16.7	<u>57.4</u>	3.3	29.9

Table 2. Quantitative results of IoU (%) on the Argoverse validation set. “Mean” indicates IoU across all classes. “Motor.”, “L.V.” denote motorcycle and large vehicle, respectively.

	Flops	Params	FPS
BEV-CAR	125.82G	34.54	43.1
HFT [35]	122.02G	43.97	34.6
PON [24]	135.67G	37.42	35.3

Table 3. Comparison of computational overhead on nuScenes. We compare the Flops, Params and FPS of CAR and two open source algorithms.

p_{drop}	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Layout IoU (%)	51.5	51.3	50.9	50.9	50.7	50.6	49.8	48.5
Object IoU (%)	21.0	21.8	22.4	22.9	23.4	23.9	23.2	22.1

Table 4. Ablation on p_{drop} . Best performance at $p_{\text{drop}} = 0.5$.

Distance Analysis. For further verification of BEV-CAR, which is claimed to be more effective in segmenting distant objects, we have posted experiments on Figure. 5 for different distance ranges.

At close range (0–15 m), BEV-CAR achieves notable improvements in “Carpark”, “Crossing”, and “Drivable” categories, thanks to better modeling of spatial context and surface continuity. These enhancements support more reliable path planning and safer low-speed maneuvers, such as parking or navigating crowded urban intersections. Gains for “Bus” and “Pedestrian” further demonstrate the model’s ability to capture fine-grained contextual cues, including ac-

curate detection of pedestrians partially occluded by parked vehicles, which helps reduce collision risks at crosswalks. In the medium range (15–30 m), the model shows strong performance gains across both common and rare object categories. Improvements in “Walkway” and “Truck”, as well as challenging classes like “Cone” and “Barrier”, indicate effective reasoning under partial visibility or occlusion. For example, more robust cone detection in construction zones enables safer lane-keeping, while better truck recognition supports appropriate following distances and overtaking decisions. Even at long range (30–56 m), where data sparsity and occlusion are most severe, BEV-CAR maintains consistent advantages in difficult categories such as “Construction Vehicle” and “Trailer”. This robustness is attributed to the optimized raster loss function, which enhances sensitivity to depth-related context and improves depth estimation.

Generalization Experiments. To demonstrate the CAR and the plug-and-play performance of the raster loss function, we conducted experiments on two classical models [24, 31]. Table 5 shows the experimental results of the generalisation experiment. The results demonstrate that the model is significantly more capable of segmenting the layout, highlighting the plug-and-play capability of CAR.

Adverse Conditions. As shown in Table 6, CAR consistently improves perception performance under adverse conditions such as rain and low-light scenarios on nuScenes dataset, yielding higher mIoU scores across both Layouts and Objects compared to the variant without CAR. Addi-

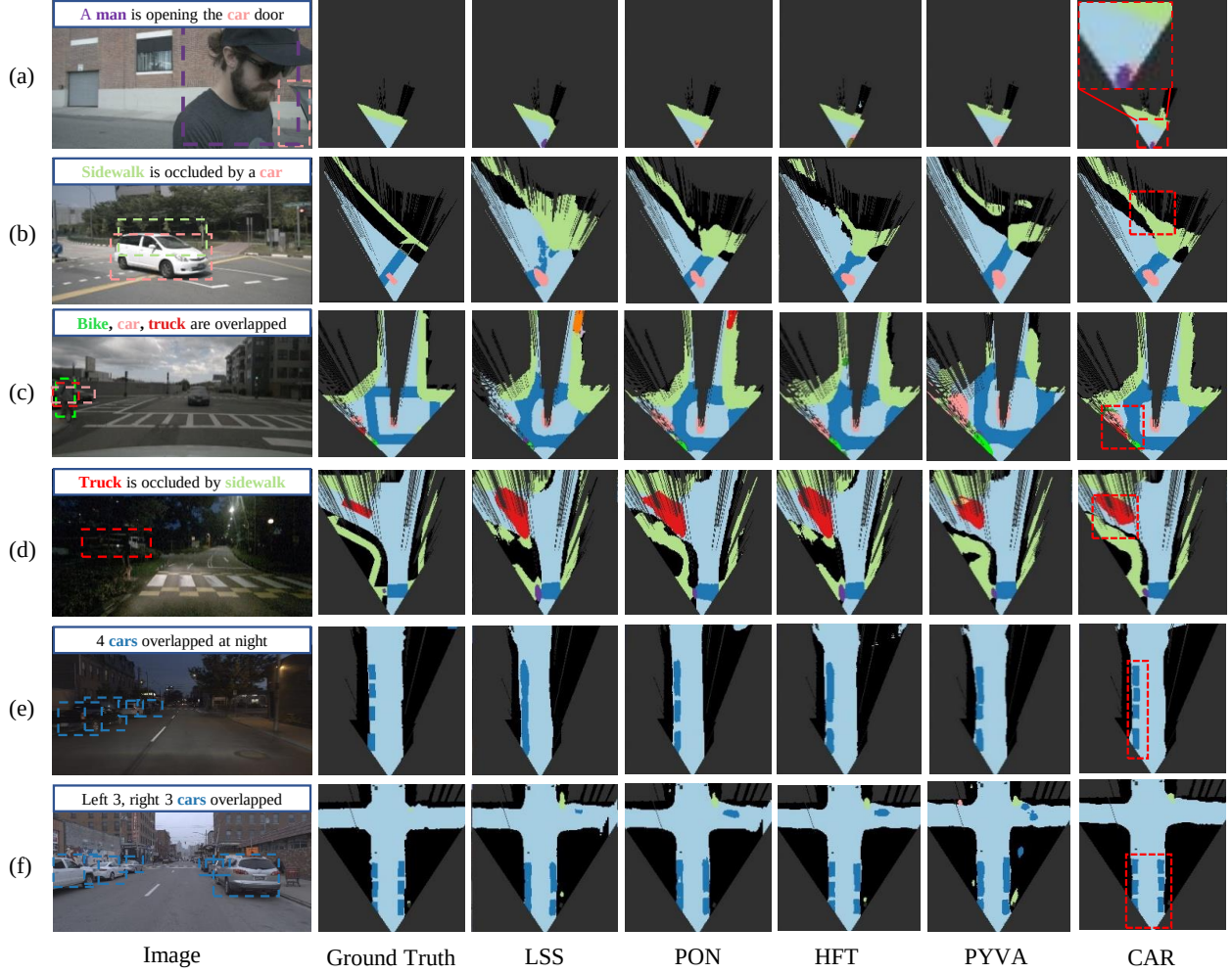


Figure 4. Qualitative results on nuScenes [2] and Argoverse [4] datasets. (a)-(d) are the visualization results for the nuScenes dataset and (e), (f) are the results for the Argoverse dataset. We compare with other methods following the color scheme used in HFT [35].

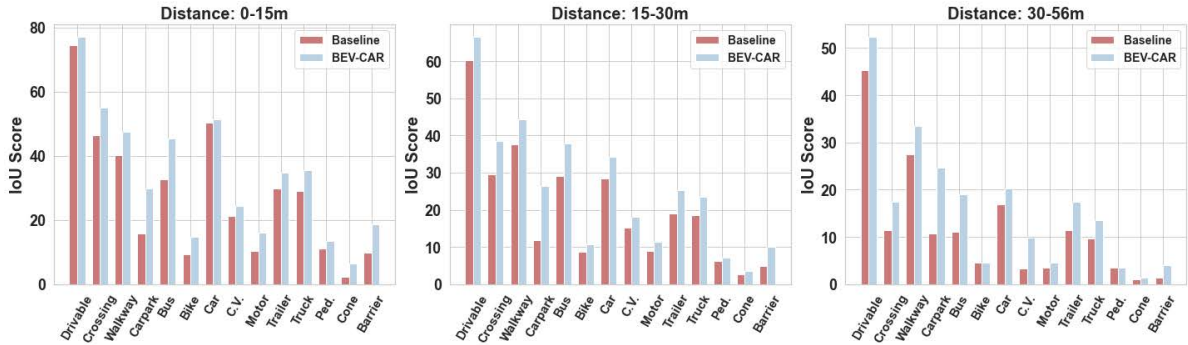


Figure 5. Comparison of IoU between the Baseline method and our proposed BEV-CAR approach across three different distance ranges on the nuScenes validation set. Each pair of bars represents the IoU performance for one category at a specific distance range.

tional qualitative results and visual comparisons are provided in the supplementary material.

Sensitivity Analysis of p_{drop} . As shown in Table 4, we tested $p_{\text{drop}} \in \{0, 0.1, \dots, 0.7\}$ on nuScenes val: low val-

Method	Lay Out	Objects	Mean
PYVA [31]	34.02	10.23	22.13
PYVA - CAR	46.25	9.74	27.99
PON [24]	34.45	12.96	23.08
PON - CAR	40.59	10.34	25.47

Table 5. Results of IoU(%) experiments on the generalisability of context-aware rasterization mechanisms with raster loss.

Condition	Layouts	Objects	Mean
Rain (CAR)	49.6	22.0	35.8
Rain (NoCAR)	43.7	21.1	32.4
Night (CAR)	47.5	19.6	33.6
Night (NoCAR)	41.8	18.8	30.3

Table 6. Ablation study of the CAR module in adverse conditions (rainy and night), where CAR is our full model and NoCAR is the variant without it.

DB	GB	RL	Layout	Objects	Mean
✓			41.2	13.8	27.5
✓	✓		44.8	23.7	34.4
✓	✓	✓	50.6	23.9	37.3

Table 7. Ablation experiments of BEV-CAR on nuScenes. DB and GB denote the depth-based and global BEV features, respectively, both using a unified decoder, and RL denotes the raster loss.

ues (≤ 0.4) overfit near points (high layout, low object IoU); high values (> 0.5) cause excessive “point dropping”. $p_{\text{drop}} = 0.5$ achieves the best trade-off.

Computational Overhead. To evaluate computational efficiency, we report the number of parameters, FLOPs, and FPS of BEV-CAR with 1024×1024 input on an RTX 4090 GPU. CAR introduces no additional parameters or inference overhead; only the BEV feature extraction and fusion branches add a minimal parameter count. As shown in Table 3, BEV-CAR achieves real-time performance at 43.1 FPS, outperforming HFT [35] and PON [24], while maintaining competitive model size and computational cost, making it suitable for real-world deployment.

4.3. Qualitative Comparison

Figure 4 shows the visual comparison results of BEV-CAR with other methods on the nuScenes and Argoverse datasets. To make the figure more understandable, the colors of the bounding boxes in the figure are aligned with the segmentation classes colors, and the figure includes six representative scenes, each of which highlights challenging real-world scene, especially those involving occlusions.

lidar Blind Zone (Figure 4(a)). A person is opening a car door in an area not fully captured by lidar. Due to missing points, most methods fail to detect or segment this per-

son and car door. BEV-CAR successfully segments the car, showing its robustness under partial visibility.

Occlusion by Vehicle (Figure 4(b)). The pedestrian crossing and walkway are partially blocked by a vehicle. Many methods produce incomplete or incorrect predictions due to occlusion. BEV-CAR maintains accurate and continuous segmentation, demonstrating its ability to handle occlusion.

Multiple Overlapping Objects (Figure 4(c)). This scene contains a bike, truck, and car that are heavily overlapped. Some methods struggle to separate them clearly. BEV-CAR accurately separate each object while preserving the layout, showing strong performance in complex occlusion cases.

Low-light Conditions (Figure 4(d)). At night with limited visibility, shadows and reflections make it hard to detect objects. Despite these challenges, BEV-CAR still produces clear and accurate segmentation of both layout and objects.

Dense Parking Scenarios (Figures 4(e)–(f)). These images depict dense parking scenarios with multiple closely spaced vehicles. Other methods tend to merge nearby vehicles into one region. BEV-CAR uses rasterization to better separate adjacent objects.

4.4. Ablation Experiments.

In this section, we conduct ablation studies on the nuScenes dataset to evaluate the contributions of each key component in BEV-CAR: the depth BEV feature branch (DBF), global BEV feature branch (GBF), and raster loss (RL). The results are summarized in Table 7, where “DB” denotes DBF, “GB” denotes GBF, and “RL” denotes the raster loss.

Baseline. We define our baseline model as the architecture that only includes the depth BEV feature branch with a unified decoder. This baseline achieves an average IoU of 21.8%, with 41.2% for layout and 13.8% for objects.

Feature Fusion. By adding the global BEV feature to the baseline, we observe a significant improvement in both layout and object segmentation performance. The average IoU increases to 28.1%, with layout IoU reaching 44.8% and object IoU rising to 23.7%. This indicates that incorporating global contextual features enhances the model’s ability to capture scene structure and better recognize objects.

Raster Loss. Further introducing the raster loss leads to a substantial improvement in layout while maintaining similar performance on objects. Specifically, the layout IoU increases from 44.8% to 50.6%, showing that raster loss effectively regularizes the prediction and enhances spatial coherence in the BEV space. Although the object IoU is only 0.2% higher, the overall average IoU reaches 31.5%, indicating that raster loss contributes more to layout understanding without harming object segmentation.

5. Conclusion

In this paper, we extract depth features and fuse them with global BEV features to enrich representation. Dur-

ing training, we use rasterization to refine BEV depth without adding inference overhead. Experiments on two datasets validate the effectiveness of BEV-CAR, and its strong performance in monocular BEV segmentation suggests promising extensions to related tasks like end-to-end autonomous driving.

References

- [1] J. E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4(1):25–30, 1965. [3](#), [4](#)
- [2] 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 multi-modal dataset for autonomous driving. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. [5](#), [6](#), [7](#)
- [3] Yigit Baran Can, Alexander Liniger, Danda Pani Paudel, and Luc Van Gool. Structured bird’s-eye-view traffic scene understanding from onboard images. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15641–15650, 2021. [1](#)
- [4] Ming-Fang Chang, Deva Ramanan, James Hays, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, and Simon Lucey. Argoverse: 3d tracking and forecasting with rich maps. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. [5](#), [7](#)
- [5] Isht Dwivedi, Srikanth Malla, Yi-Ting Chen, and Behzad Dariush. Bird’s eye view segmentation using lifted 2d semantic features. In *British Machine Vision Conference*, 2021. [1](#)
- [6] Jingjing Fan, Lili Fan, Qinghua Ni, Junhao Wang, Yi Liu, Ren Li, Yutong Wang, and Sanjin Wang. Perception and planning of intelligent vehicles based on bev in extreme off-road scenarios. *IEEE Transactions on Intelligent Vehicles*, 9(4):4568–4572, 2024. [1](#)
- [7] Shi Gong, Xiaoqing Ye, Xiao Tan, Jingdong Wang, Er-rui Ding, Yu Zhou, and Xiang Bai. Gitnet: Geometric prior-based transformation for birds-eye-view segmentation. In *ECCV*, pages 396–411, Cham, 2022. Springer Nature Switzerland. [1](#), [2](#), [5](#), [6](#)
- [8] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 27 (NIPS 2014)*, pages 2672–2680, 2014. 28th Conference on Neural Information Processing Systems (NIPS), Montreal, CANADA, DEC 08-13, 2014. [1](#)
- [9] Chenhang He, Hui Zeng, Jianqiang Huang, Xian-Sheng Hua, and Lei Zhang. Structure aware single-stage 3d object detection from point cloud. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11870–11879, 2020. [1](#)
- [10] Anthony Hu, Zak Murez, Nikhil Mohan, Sofia Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: Future instance prediction in bird’s-eye view from surround monocular cameras. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15253–15262, 2021. [1](#)
- [11] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9223–9232, 2023. [1](#)
- [12] Yanqin Jiang, Li Zhang, Zhenwei Miao, Xiatian Zhu, Jin Gao, Weiming Hu, and Yu-Gang Jiang. Polarformer: Multi-camera 3d object detection with polar transformer. 2022. [1](#), [2](#)
- [13] Abhijit Kundu, Yin Li, Frank Dellaert, Fuxin Li, and James M. Rehg. *Joint Semantic Segmentation and 3D Reconstruction from Monocular Video*, pages 703–718. Springer International Publishing, Cham, 2014. [2](#)
- [14] Wenxi Liu, Qi Li, Weixiang Yang, Jiaxin Cai, Yuanlong Yu, Yuexin Ma, Shengfeng He, and Jia Pan. Monocular bev perception of road scenes via front-to-top view projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(9):6109–6125, 2024. [5](#), [6](#)
- [15] Zhi Liu, Shaoyu Chen, Xiaojie Guo, Xinggang Wang, Tianheng Cheng, Hongmei Zhu, Qian Zhang, Wenyu Liu, and Yi Zhang. Vision-based uneven bev representation learning with polar rasterization and surface estimation. In *Proceedings of The 6th Conference on Robot Learning*, pages 437–446. PMLR, 2023. [1](#), [2](#)
- [16] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L. Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird’s-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2774–2781, 2023. [1](#)
- [17] Chenyang Lu, Marinus Jacobus Gerardus van de Molengraft, and Gijs Dubbelman. Monocular semantic occupancy grid mapping with convolutional variational encoder-decoder networks. *IEEE Robotics and Automation Letters*, page 445–452, 2019. [1](#), [2](#)
- [18] Kaustubh Mani, Swapnil Daga, Shubhika Garg, N. Sai Shankar, J. Krishna Murthy, and K. Madhava Krishna. Monolayout: Amodal scene layout from a single image. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020. [1](#), [2](#)
- [19] Shiyu Meng and Yuxiang Sun. Semantic-moseg: Semantics-assisted moving-obstacle segmentation in bird-eye-view for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 26(7):9251–9262, 2025. [5](#), [6](#)
- [20] B. Pan, J. Sun, H. Y. T. Leung, A. Andonian, and B. Zhou. Cross-view semantic segmentation for sensing surroundings. *IEEE Robot. Autom. Lett.*, 2020. [1](#), [2](#), [5](#), [6](#)
- [21] Jonah Philion and Sanja Fidler. *Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D*, page 194–210. 2020. [2](#), [3](#)
- [22] Shunming Qiu, Xingyu Li, Xiaochao Xue, and Jie Pu. PC-BEV: An Efficient Polar-Cartesian BEV Fusion Framework for LiDAR Semantic Segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6612–6620, 2025. [1](#), [2](#)

- [23] Lennart Reiher, Bastian Lampe, and Lutz Eckstein. A sim2real deep learning approach for the transformation of images from multiple vehicle-mounted cameras to a semantically segmented image in bird's eye view. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 2020. [2](#)
- [24] Thomas Roddick and Roberto Cipolla. Predicting semantic map representations from images using pyramid occupancy networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. [1](#), [5](#), [6](#), [8](#)
- [25] T. Roddick, A. Kendall, and R. Cipolla. Orthographic feature transform for monocular 3d object detection. *arXiv preprint arXiv:1811.08188*, 2018. [2](#)
- [26] Avishkar Saha, Oscar Mendez, Chris Russell, and Richard Bowden. Translating images into maps. In *2022 International Conference on Robotics and Automation (ICRA)*, 2022. [1](#), [2](#), [5](#), [6](#)
- [27] Samuel Schuster, Menghua Zhai, Nathan Jacobs, and Manmohan Chandraker. Learning to look around objects for top-view representations of outdoor scenes. In *Computer Vision – ECCV 2018*, pages 815–831, Cham, 2018. Springer International Publishing. [2](#)
- [28] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, 2014. [4](#)
- [29] Jian Sun, Yuqi Dai, Chi-Man Vong, Qing Xu, Shengbo Eben Li, Jianqiang Wang, Lei He, and Keqiang Li. Oe-bevseg: An object informed and environment aware multimodal framework for bird's-eye-view vehicle semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–14, 2025. [5](#), [6](#)
- [30] Huitong Yang, Xuyang Bai, Xinge Zhu, and Yuexin Ma. One training for multiple deployments: Polar-based adaptive bev perception for autonomous driving. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5602–5609, 2023. [2](#)
- [31] Weixiang Yang, Qi Li, Wenxi Liu, Yuanlong Yu, Yuexin Ma, Shengfeng He, and Jia Pan. Projecting your view attentively: Monocular road scene layout estimation via cross-view transformation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. [1](#), [2](#), [6](#), [8](#)
- [32] Zichen Yu, Quanli Liu, Wei Wang, Liyong Zhang, and Xiaoguang Zhao. Polarbevdet: Exploring polar representation for multi-view 3d object detection in bird's-eye-view. 2024. [1](#), [2](#)
- [33] Tianhao Zhao, Yongcan Chen, Yu Wu, Tianyang Liu, Bo Du, Peilun Xiao, Shi Qiu, Hongda Yang, Guozhen Li, Yi Yang, and Yutian Lin. Improving bird's eye view semantic segmentation by task decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15512–15521, 2024. [2](#), [5](#), [6](#)
- [34] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9934–9943, 2021. [1](#)
- [35] Jiayu Zou, Zheng Zhu, Junjie Huang, Tian Yang, Guan Huang, and Xingang Wang. Hft: Lifting perspective representations via hybrid feature transformation for bev perception. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7046–7053, 2023. [2](#), [5](#), [6](#), [7](#), [8](#)
- [36] Jiayu Zou, Zheng Zhu, Yun Ye, and Xingang Wang. Diffbev: Conditional diffusion model for bird's eye view perception. In *2024 Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2024. [2](#), [5](#), [6](#)