This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Spe-BEVHead: Rethinking the Detection Head Design for Bird's-Eye-View Object Detection

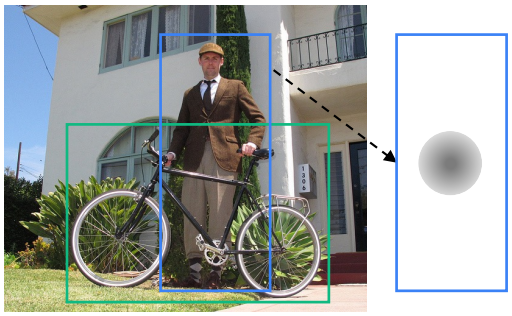
Junshu Zhang[1] Sicheng Zhao[2*] Xin Zhao[3] Fan Yang[1] Ruike Chen[1]
Jungong Han[4] Guiguang Ding[1*]
[1]School of Software [2]Department of Psychological and Cognitive Sciences
[3]BNRist [4]Department of Automation
Tsinghua University
junshu-z24@mails.tsinghua.edu.cn {schzhao, dinggg}@tsinghua.edu.cn

Abstract

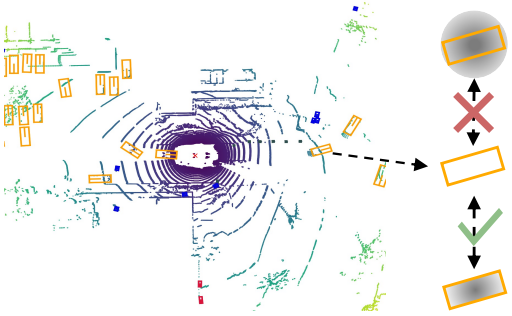
Bird's-Eye-View (BEV) detection has become a dominant paradigm for 3D object detection in autonomous driving, due to its strong perception capability. However, most existing methods mainly focus on constructing high-quality BEV feature representations, while neglecting the design of task-specific detection heads. In practice, they directly adopt the center-based head originally developed for 2D detection, without any specific optimization. This leads to three inherent limitations: (i) a geometric mismatch between the Gaussian kernel used for classification and the real BEV object, (ii) degraded end-to-end performance without Non-Maximum Suppression (NMS), and (iii) sparse supervisory signals. To address these issues, we propose Spe-BEVHead, a detection head specifically tailored for BEV 3D object detection. Spe-BEVHead introduces three BEV-specific adaptations: (1) a Rotated Box Kernel that generates geometry-aligned classification weights, (2) a Local Response Refinement Module (LRRM) that suppresses non-peak responses and improves end-to-end performance, and (3) a dual-branch architecture that provides richer supervisory signals to promote more robust learning while inherently preserving the performance for end-to-end inference. Extensive experiments show that Spe-BEVHead can be seamlessly integrated into existing BEV backbones, delivering direct performance gains while retaining competitive performance under the challenging end-to-end setting.

1. Introduction

3D object detection is a core task in autonomous driving, enabling reliable perception for safe navigation and decision-making. Among existing paradigms, Bird's-Eye-View (BEV) detection [6, 12–14] has become dominant for its unified representation that facilitates multi-sensor fusion and holistic scene understanding. Unlike monocular 3D methods [18, 23, 24, 26, 38, 39, 43, 44], BEV approaches provide full 360° environmental perception. Recent advances largely attribute performance gains to stronger encoders and more effective feature aggregation for accurate BEV representation learning.



(a) Objects and Weighting Kernel in 2D Images



(b) Objects and Weighting Kernels in BEV

Figure 1. Comparison between 2D and BEV object distributions and the effects of different weighting kernels on them. The gradient gray circle denotes the Gaussian kernel, and the gradient gray rectangle represents the proposed Rotated Box kernel.

*Corresponding authors.

Early Transformer-based approaches [14, 19, 33] introduced a paradigm that learns to construct BEV representations from multi-view 2D image features through attention mechanisms. Rather than explicitly projecting pixels into the BEV plane, these methods employ cross-view and spatial attention to implicitly learn correspondences between image features and BEV queries, enabling the integration of multi-camera context into a unified representation. Although achieving strong perception accuracy, their heavy computational cost and slow inference speed limit their applicability in real-time autonomous driving systems.

Recent studies have focused on more efficient Lift-Splat-Shoot (LSS)-based frameworks [5, 6, 12, 13, 25], which lift multi-view image features into discretized 3D frustum volumes where depth—either estimated or explicitly predicted—guides feature distribution along the depth axis. The lifted 3D features are then aggregated onto a 2D BEV plane to form a unified representation integrating information from all cameras. Advances in LSS-based models have improved both efficiency and accuracy through enhanced depth estimation [12], faster view-to-BEV transformation [13], and more effective pooling strategies [42]. Despite these improvements in BEV feature generation, existing LSS-based detectors still employ the 2D center-based head [48] originally designed for image-plane tasks. While widely adopted, this head is not specifically optimized for BEV detection and not all its modules are inherently suited for BEV perception. In this work, we analyze the limitations of center-based detection head [48] in BEV 3D detection and introduce corresponding improvements.

First, center-based detection head [48] employs Gaussian kernels to assign soft classification weights, reducing the loss contribution from near-center negative samples to encourage precise localization. However, this approach struggles with the substantial scale disparity between the image and BEV planes. As shown in Fig. 1, 2D objects are often large, typically confining the Gaussian kernel within the ground-truth box. In contrast, BEV representations are geometrically tied to the physical world, rendering targets much smaller. Consequently, directly applying the Gaussian radius computation from 2D tasks to BEV leads to an overly large radius, causing the resulting Gaussian kernel to extend beyond the object boundary. This incorrectly down-weights background pixels, leading to suboptimal supervision. To address this, as shown in Fig. 1b, we propose a Rotated Box kernel that replaces the conventional Gaussian kernel, generating geometry-aligned classification weights better suited for BEV detection.

Second, the center-based detection head enforces a strict one-to-one matching between positive samples and ground truths to enable end-to-end detection without Non-Maximum Suppression (NMS), which cannot be efficiently parallelized or executed on GPUs. In practice, NMS is still required to remove redundant boxes, primarily because convolutional feature extractors struggle to produce sharp responses. In 2D detection, as shown in Fig. 1a, variations in feature-map scales, perspective distortion, overlapping boxes, and the need for fine-grained features to classify numerous categories make it difficult to design modules that explicitly suppress non-central responses within a local region. In contrast, BEV objects have fixed sizes and ground truth boxes rarely overlap, allowing each object to be processed locally without affecting others. Motivated by these properties, we propose a Local Response Refinement Module (LRRM) to suppress non-central responses, which directly improves the end-to-end performance.

Third, regression in BEV detection is inherently more challenging than in 2D tasks, as it involves more attributes. However, the center-based head supervises only the center pixels of the objects through regression, resulting in extremely sparse supervision signals. Another challenge is the inherent spatial misalignment between the predicted heatmap peak and the ground-truth centroid, often caused by downsampling or feature fuzziness. This necessitates the incorporation of a localization refinement capability to accurately compensate for these near-center offsets. To address the above issues, we redesign the detection head into a dual-branch structure consisting of a main and an auxiliary branch, which enriches the supervision while maintaining the one-to-one matching required for end-to-end detection. We integrate all the proposed improvements into a unified detection head, named **Spe-BEVHead**, specifically designed for BEV detection. We conduct extensive experiments on the challenging nuScenes [1] benchmark to validate its effectiveness. Spe-BEVHead can be seamlessly plugged into any LSS-based models, yielding immediate performance gains. Moreover, under an end-to-end setting, it achieves competitive results. In summary, our contributions are as follows:

- We analyze the inherent limitations of directly applying 2D detection's center-based head to the BEV detection paradigm.
- We propose Spe-BEVHead, a novel design specifically tailored for the unique challenges of BEV detection tasks.
- We evaluate our method on the nuScenes dataset, where Spe-BEVHead demonstrates its effectiveness in improving the performance of existing networks and maintains competitive performance within an end-to-end setting.

2. Related Work

2.1. Transformer Based BEV Perception

With the success of Transformers [29], attention mechanisms have been widely adopted in BEV detection. BEVFormer [14] introduces deformable attention to transform image features into BEV space, jointly modeling spatial

and temporal cues. BEVFormerV2 [37] adds perspective-view supervision and a two-stage design for faster convergence and better compatibility with modern backbones. OCBEV [27] employs object-centric queries to more effectively capture temporal and spatial dynamics of moving targets. BEVDiffuser [41] extends BEVFormer with a diffusion model to denoise BEV feature maps, while PolarFormer [8] leverages a polar coordinate representation for more precise localization. CorrBEV [36] incorporates visual–language prototypes to recover object features degraded by occlusion.

To reduce computational overhead, some methods avoid explicitly constructing the BEV features. DETR3D [33] directly manipulates object queries in 3D space to associate multi-view image features, eliminating the need for depth prediction and post-processing. PETR [19] encodes the position information of 3D coordinates into image features, producing 3D position-aware features. PETRv2 [20] extends PETR to temporal modeling and BEV semantic segmentation tasks. StreamPETR [32] leverages sparse queries as intermediate representations to model moving objects and effectively propagate temporal information. RayDN [17] improves detection accuracy by strategically sampling along camera rays to construct hard negative examples. However, these Transformer-based models often suffer from high computational costs and slow inference speeds, hindering their deployment in real-time autonomous driving systems.

2.2. LSS Based BEV Perception

Lift-Splat-Shoot (LSS) [25] introduces a differentiable view transformation pipeline that lifts multi-view image features into 3D space via predicted depth distributions and then splats them onto a BEV feature map. BEVDet [6] first applies LSS to 3D object detection and enhances performance through a data augmentation strategy. BEVDet4D [5] extends BEVDet by incorporating historical frames to improve temporal perception. BEVDepth [12] leverages camera intrinsics and extrinsics for explicit depth supervision and introduces a depth correction sub-network to reduce projection noise. BEVStereo [11] further improves depth accuracy via temporal stereo across adjacent frames. MatrixVT [46] employs convolution and matrix multiplication to efficiently generate BEV features, while GeoBEV [42] uses In-Box Labels to recover object geometry in BEV space. M^2BEV [35] transforms multi-view 2D features into ego-centric 3D BEV representations with a shared encoder for multi-task learning, and FastBEV [13] accelerates this process via a look-up table. BEVNeXT [15] further strengthens LSS-based frameworks with improved architectures and training strategies. However, most LSS-based methods still directly adopt the 2D center-based detection head without task-specific optimization for BEV detection.

2.3. End-to-end Object Detection

As Non-Maximum Suppression (NMS) introduces additional computational overhead and is not amenable to GPU parallelization, achieving an elegant end-to-end architecture has been a consistent pursuit in object detection algorithms. DETR [2] formulates object detection as a direct set prediction problem, leveraging a Transformer-based architecture to achieve end-to-end detection without the need for post-processing. A series of DETR variants [7, 10, 22, 34, 40, 45, 50, 51] have been proposed to further enhance the performance of DETR. Some fully convolutional end-to-end models have also emerged in 2D object detection tasks. YOLOv10 [30] and DATE [3] adopt a dual-label assignment strategy to enable end-to-end detection while providing rich supervisory signals. OneNet [28] and DeFCN [31] design unique one-to-one label assignment strategies to suppress redundant boxes. CenterNet [48] generates a unique bounding box for each object by predicting its center. PSS [47] employs a positive sample selector to identify the unique for prediction. Taking advantage of the Transformer architecture, BEVFormer [14] naturally functions as an end-to-end 3D detection model. Despite adopting the CenterNet [48] head, fully convolutional and LSS-based BEV models suffer severe performance degradation under an end-to-end setting.

3. Method

This section details our proposed Spe-BEVHead, a detection head specifically designed and optimized for the BEV-based 3D object detection task.

3.1. Architecture of BEV detection model

The overall workflow of BEV detection networks is illustrated in Fig. 2 (left). First, the multi-view images are processed by an image backbone [4, 9] to extract multi-view features. Second, a feature transformation module converts the image features into a BEV representation. Third, a detection head predicts the final bounding boxes. The feature transformation module can be categorized into Transformer-based methods [14, 41] and LSS-based methods [6, 12], which respectively employ a Transformer decoder head [50] and a center-based detection head [48] for object prediction. Our innovations are focused on the BEV detection head.

3.2. Rotated Box Kernel

In the center-based detection head, a Gaussian kernel is used to reduce penalties for near-center negatives, guiding the network to predict keypoints closer to object centers. However, the scale relationship between object size and feature resolution differs drastically between 2D and BEV domains. For example, in FastBEV [13], the BEV feature map

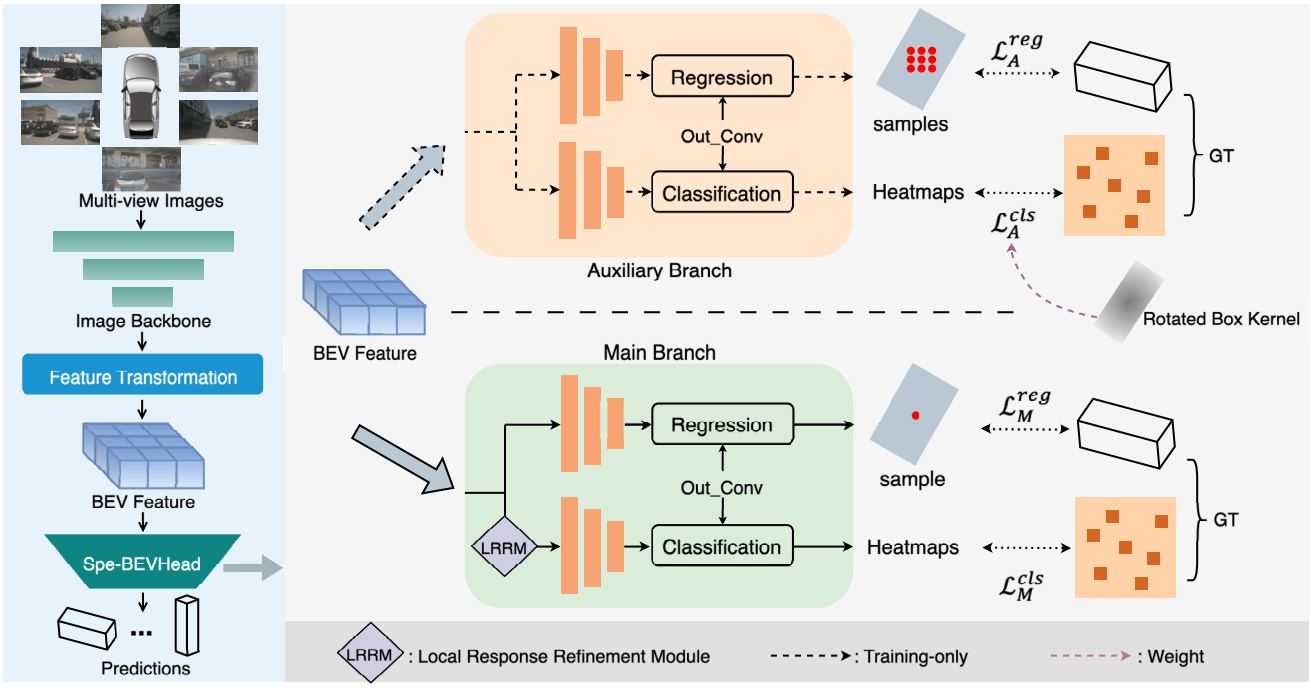


Figure 2. The left part of the figure illustrates the workflow of BEV detection, which consists of an image backbone, a feature transformation module, and a detection head. Our innovation lies in the design of the detection head, as shown in the right part of the figure. Specifically, we propose the Spe-BEVHead tailored for BEV detection, featuring a dual-branch architecture composed of a main branch (right bottom) and an auxiliary branch (right top).

is 128×128, while a typical vehicle covers fewer than 20 pixels. In contrast, 2D objects often occupy a large portion of the image. Consequently, a small Gaussian kernel in BEV space may extend beyond object boundaries, mistakenly down-weighting background pixels. To address this, we propose the Rotated Box Kernel (RBK), a geometry-aligned weighting mechanism tailored for BEV detection.

To simultaneously represent the weights of both object and background locations, we splat all ground-truth keypoints (object centers) onto a heatmap $\mathbf{H} \in [0,1]^{W \times H \times C}$ using the Rotated Box kernel, where W and H denote the width and height of the heatmap (identical to the BEV feature map), and C represents the number of object categories (10 for the nuScenes dataset). For each ground-truth bounding box (projected onto the BEV coordinate system) with center $\mathbf{k} = (k_x, k_y)$, size (w, l), yaw rotation angle θ and class c, we firstly transform pixel coordinates from the global frame to the box's local coordinate system. For each pixel $\mathbf{p} = (x, y)$ on the heatmap of class c, the local coordinates are computed via rotation and translation:

$$\begin{bmatrix} x_\ell \\ y_\ell \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x - k_x \\ y - k_y \end{bmatrix}, \tag{1}$$

where the rotation matrix aligns the coordinate axes with the box's principal directions. Next, we compute the normalized elliptical distance from each pixel to the target center:

$$d(x, y) = \sqrt{\left(\frac{2x_\ell}{w}\right)^2 + \left(\frac{2y_\ell}{l}\right)^2}. \tag{2}$$

This distance metric naturally adapts to different aspect ratios through the elliptical formulation. Finally, for pixels within the target bounding box (where $d \leq 1$), the corresponding heatmap values are computed as follow:

$$\mathbf{H}_{xyc} \leftarrow \max\Big(\mathbf{H}_{xyc},\, K \cdot \mathrm{clip}\big(1 - \gamma d^2,\, v_{\min},\, 1\big)\Big), \tag{3}$$

where $K = 1$ represents the center value, $\gamma = 1 - v_{\min}$ controls the decay rate and $v_{\min}$ denotes the boundary value (0.1 by default) of the target range. The max operation ensures that overlapping boxes preserve their peak responses. For pixels outside the target bounding box (where $d > 1$), their values are set to 0. Within the coverage area of the Rotated Box kernel, the values decay with increasing distance from the center. Moreover, the kernel explicitly accounts for the object's rotation and aspect ratio in BEV space, thereby providing more precise weighting for the loss calculation.

3.3. Local Response Refinement Module

Although center-based detection heads establish a strict one-to-one correspondence between positive samples and

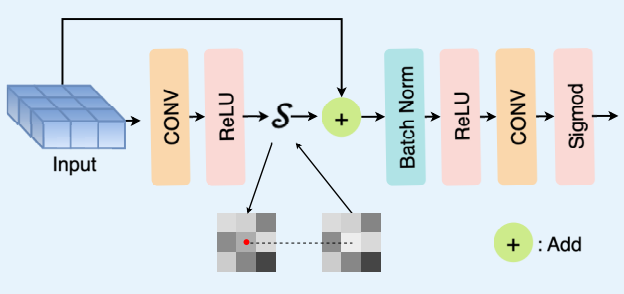


Figure 3. Architecture of Local Response Refinement Module. $\mathcal{S}$ denotes the Adaptive Mean Attenuation operator.

ground truths to enable end-to-end detection without Non-Maximum Suppression (NMS), in practice, additional post-processing such as pooling or NMS is still required to eliminate redundant predictions. This limitation primarily stems from the inability of convolutional layers to produce sufficiently sharp and discriminative responses [31]. Designing a dedicated non-peak suppression module for 2D detection networks is particularly challenging, as 2D bounding boxes frequently overlap and rely on fine-grained texture cues for accurate classification across a large number of categories. Integrating such a module may disrupt critical object representations and inadvertently suppress central peaks corresponding to nearby targets. In the BEV space, objects rarely overlap and the number of object categories is relatively small. These characteristics motivate the design of the Local Response Refinement Module (LRRM) to suppress non-peak responses. As illustrated in Fig. 3, the LRRM consists of several convolutional layers, nonlinear activation functions, and an Adaptive Mean Attenuation (AMA) operator denoted by $\mathcal{S}$. The AMA operator is formulated as:

$$\mathbf{F}'(x,y) = \begin{cases} \mathbf{F}(x,y) \\ \quad \text{if } \mathbf{F}(x,y) = \max\limits_{(i,j)\in\mathcal{N}_k(x,y)} \mathbf{F}(i,j); \\ \mathbf{F}(x,y) - \dfrac{1}{k^2} \sum\limits_{(i,j)\in\mathcal{N}_k(x,y)} \mathbf{F}(i,j) \\ \quad \text{otherwise}, \end{cases} \tag{4}$$

where $\mathbf{F}(x,y)$ denotes the input feature map at position (x,y), and $\mathbf{F}'(x,y)$ represents the output after suppression. $\mathcal{N}_k(x,y)$ indicates a neighborhood of size $k \times k$ centered at (x,y). If the central pixel $\mathbf{H}(x,y)$ is the maximum within $\mathcal{N}_k(x,y)$, its response is preserved; otherwise, the neighborhood mean is subtracted to suppress non-peak activations. LRRM can be integrated into the network for end-to-end training; during inference, it suppresses responses at non-peak locations, enhancing end-to-end performance.

3.4. Dual Branch Architecture of Spe-BEVHead

The center-based head enforces a strict one-to-one matching, yielding extremely sparse supervisory signals. This sparse supervision is suboptimal for the complex BEV regression task, which includes center offsets $(x_{\text{offset}}, y_{\text{offset}})$, height-above-ground h_g, yaw rotation $((\sin(\theta), \cos(\theta))$, 3D dimensions (w, l, h), and velocity v. Furthermore, network downsampling and feature blurring cause inevitable spatial misalignment between the peak response and the ground-truth centroid. Consequently, the network must incorporate a localization refinement capability to compensate for predicted offsets at these near-center locations. To address the aforementioned issues while preserving the end-to-end detection paradigm of the center-based head, we design the proposed Spe-BEVHead as a dual-branch architecture, consisting of an auxiliary branch and a main branch, as illustrated in Fig. 2 (right). Each branch follows a similar structural design but differs in several details, which will be elaborated in the following.

Main branch. As illustrated in the lower part of Fig. 2 (right), the main branch consists of a classification head and a regression head. Both classification and regression strictly follow a one-to-one label assignment strategy, where each object is associated with only one positive sample. To further enhance the end-to-end inference capability, the features are first processed by the LRRM proposed in Sec. 3.3 before entering the classification head.

Auxiliary branch. As shown in the upper part of Fig. 2 (right), the auxiliary branch also consists of a classification head and a regression head. The key difference lies in the regression process, where the auxiliary branch abandons the strict one-to-one matching mechanism and instead increases the number of positive samples. Specifically, the regression head of the auxiliary branch selects nine samples(the target center and its eight neighboring points) to participate in loss computation. This fixed sampling strategy is adopted for three reasons: (1) the large spatial size of the BEV feature makes it impractical to dynamically search [28, 31] for the optimal matching samples across the entire feature map; (2) to avoid including pixels outside the ground-truth bounding box, we do not further enlarge the sampling radius; and (3) assigning all pixels within the bounding box as regression samples may harm training stability, since for larger targets, edge pixels and central pixels should not share identical regression objectives. For the classification head, we continue to follow the one-to-one label assignment principle. The motivation behind this design is to avoid introducing label ambiguity, particularly preventing a scenario where a specific grid location is treated as a negative sample in the main branch but as a positive one in the auxiliary branch. We argue that such conflicting supervisory signals are detrimental to the convergence of the classification task. However, unlike the main branch, the auxiliary classification head differs

in how the loss is computed — it incorporates the Rotated Box Kernel introduced in Sec. 3.2 to weight the classification loss. The kernel serves to alleviate the penalties on negative samples near the object center, acting as a softened one-to-one matching mechanism analogous to the principle adopted in the center-based head [48].

During training, both branches provide gradients to jointly optimize the network parameters. However, the auxiliary branch is discarded during inference and only the main branch is utilized for the final prediction.

3.5. Loss

The overall loss is formulated as shown in Eq. (5):

$$\mathcal{L} = \lambda_{\text{cls}}\mathcal{L}_M^{\text{cls}} + \lambda_{\text{reg}}\mathcal{L}_M^{\text{reg}} + \lambda_{\text{cls}}\mathcal{L}_A^{\text{cls}} + \lambda_{\text{reg}}\mathcal{L}_A^{\text{reg}}, \quad (5)$$

where $\mathcal{L}_M^{\text{cls}}$ and $\mathcal{L}_M^{\text{reg}}$ denote the classification loss and regression loss of the main branch, respectively, while $\mathcal{L}_A^{\text{cls}}$ and $\mathcal{L}_A^{\text{reg}}$ represent the classification loss and regression loss of the auxiliary branch, respectively. λ_{cls} and λ_{reg} denote the loss weights for classification and regression, respectively, and are typically set as $\lambda_{\text{cls}} = 4\lambda_{\text{reg}}$. We use L1 loss for regression. For the classification task, our objective is to predict a keypoint heatmap $\hat{\mathbf{Y}} \in [0,1]^{W\times H\times C}$. In the ground-truth heatmap $\mathbf{Y} \in \{0,1\}^{W\times H\times C}$, only the pixel corresponding to the projected location of each object center on its respective class channel is assigned a value of 1, while all other positions are set to 0. We use focal loss [16] for classification of the main branch. The classification loss of the auxiliary branch is calculated by:

$$\mathcal{L}_A^{\text{cls}} = -\frac{1}{N}\sum_{x,y,c}\begin{cases}(1-\hat{Y}_{xyc})^{\alpha}\log(\hat{Y}_{xyc}) & \\ & \text{if } Y_{xyc}=1;\\ (1-H_{xyc})^{\beta}(\hat{Y}_{xyc})^{\alpha}\log(1-\hat{Y}_{xyc}) & \\ & \text{if } Y_{xyc}<1,\end{cases} \quad (6)$$

where H_{xyc} denotes the weight assigned by the Rotated Box kernel to each pixel in the feature map, and α and β are hyper-parameters of the focal loss.

4. Experiments

4.1. Experimental Settings

Dataset. We conduct experiments on the large-scale nuScenes dataset [1], a widely used benchmark for autonomous driving perception. NuScenes consists of 1,000 20-second driving scenes recorded in Boston and Singapore, with keyframes annotated at 2 Hz. Each keyframe provides synchronized data from six RGB cameras and a LiDAR sensor covering a full 360-degree field of view, along with accurately annotated 3D bounding boxes for objects of interest. The dataset contains approximately 1.4 million 3D bounding boxes categorized into 10 object classes.

Metrics. We follow the nuScenes protocol for 3D detection evaluation. The benchmark reports mean Average Precision (mAP), where true positives are determined based on the 2D center distance between predicted and ground-truth boxes instead of IoU, making it more suitable for 3D perception. In addition, five complementary metrics—Average Translation Error (ATE), Average Scale Error (ASE), Average Orientation Error (AOE), Average Velocity Error (AVE), and Average Attribute Error (AAE)—are employed to evaluate localization, shape, orientation, motion, and attribute estimation accuracy, respectively. The nuScenes Detection Score (NDS) aggregates these metrics to provide a comprehensive weighted evaluation.

Implementation Details. To enable a fair comparison, we strictly match the experimental environment and all training hyperparameters to those used by the baseline methods. When compared to other methods, we train these models for 20 epochs with CBGS strategy [49]. We use AdamW [21] as the optimizer with a learning rate set to $2e-4$ and batch size set to 64. Unless otherwise specified, we use ResNet-50 [4] as the image backbone and the image size is processed to 256×704.

4.2. Main Results

Our baselines comprise FastBEV [13], BEVDet [6], BEVDet4D [5], BEVStereo [11] and GeoBEV[42], broadly spanning the prevailing LSS-based BEV detection models. FastBEV replaces the complex BEV feature transformation process with a lookup-table operation, making it the fastest BEV detector to date. BEVDet serves as the first BEV detection framework, while BEVDet4D is an enhanced version that incorporates temporal information. BEVDepth builds upon BEVDet by introducing explicit depth estimation and supervision to strengthen the BEV feature. BEVStereo further introduces temporal stereo to enhance depth estimation. All models follow the LSS architecture and employ a center-based detection head. In our experiments, we directly replace their original detection heads with the proposed Spe-BEVHead, without modifying any other network components. We report the 3D object detection performance of all models achieved on nuScenes val dataset in Tab. 1. Equipping the models with Spe-BEVHead leads to broadly consistent improvements on the primary metrics (NDS, mAP, mATE); the effect is most notable on FastBEV, where we observe $+1.7\%$ NDS, $+1.1\%$ mAP, and a -5.2% mATE. By integrating Spe-BEVHead with the previous state-of-the-art, GeoBEV, we achieve a new state-of-the-art.

4.3. End-to-End Performance

In our end-to-end experiments, we select the top-150 predictions ranked by classification scores and discard those with scores below 0.1 as final outputs. As shown in Tab. 2,

Table 1. Comparison with the baselines on the nuScenes val set. "4D" indicates that temporal information from preceding frames is incorporated. Odd-numbered rows report the results of the baseline model with the center-based head, whereas even-numbered rows present those of the baseline model equipped with the proposed Spe-BEVHead. ↑ means higher is better; ↓ means lower is better.

Method	Frames	BEV Size	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
FastBEV [13]	1	128 × 128	38.4	29.5	74.9	0.277	0.573	0.827	0.213
+ Spe-BEVHead	1	128 × 128	**40.1**	**30.6**	**69.7**	0.277	0.515	0.828	0.207
BEVDet [6]	1	128 × 128	39.7	30.5	70.6	0.275	0.554	0.812	0.209
+ Spe-BEVHead	1	128 × 128	**40.6**	**30.7**	**70.2**	0.278	0.520	0.768	0.207
FastBEV4D [13]	2	128 × 128	**44.7**	31.3	69.1	0.276	0.519	0.423	0.188
+ Spe-BEVHead	2	128 × 128	44.5	**31.8**	**68.7**	0.282	0.518	0.454	0.200
BEVDet4D [5]	2	128 × 128	44.4	31.5	69.2	0.279	0.552	0.404	0.199
+ Spe-BEVHead	2	128 × 128	**45.3**	**32.6**	**69.1**	0.283	0.471	0.424	0.226
BEVDepth4D [12]	2	128 × 128	47.8	35.4	62.5	0.279	0.491	0.387	0.209
+ Spe-BEVHead	2	128 × 128	**48.6**	**35.6**	**61.1**	0.279	0.431	0.394	0.209
BEVStereo4D [11]	2	128 × 128	49.5	**38.1**	58.9	0.278	0.505	0.385	0.204
+ Spe-BEVHead	2	128 × 128	**49.9**	37.6	**58.2**	0.277	0.450	0.381	0.195
GeoBEV4D [42]	2	256 × 256	54.0	**42.9**	55.0	0.261	0.424	0.295	0.212
+ Spe-BEVHead	2	256 × 256	**54.6**	42.7	**54.8**	0.259	0.371	0.289	0.210

Table 2. Comparison with baselines on the nuScenes val set under an end-to-end setting. "Post" denotes the post-processing method.

Method	Post	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
FastBEV [13]	None	34.5	21.9	77.3	0.277	0.563	0.824	0.210
	NMS	38.4	**29.5**	74.9	0.277	0.573	0.827	0.213
+ Spe-BEVHead	None	37.9	26.2	71.0	0.276	0.504	0.827	0.204
	Pooling	**39.9**	28.3	**66.6**	0.278	0.482	0.778	0.222
BEVDet4D [5]	None	40.8	24.2	71.6	0.279	0.541	0.394	0.198
	NMS	44.4	**31.5**	69.2	0.279	0.552	0.404	0.199
+ Spe-BEVHead	None	43.2	28.3	70.5	0.283	0.457	0.419	0.225
	Pooling	**44.4**	29.8	**65.9**	0.282	0.453	0.426	0.229

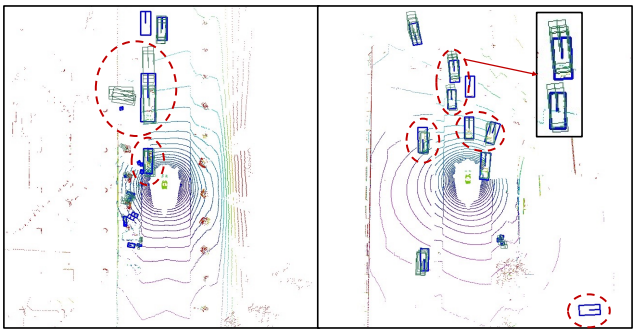
under this setting, Spe-BEVHead surpasses the center-based head by $+3.4\%$ NDS and $+7.6\%$ mAP on the FastBEV baseline. In 2D detection, the center-based head enforces a strict one-to-one matching between ground-truth objects and positive samples, producing strong responses only at object centers to achieve NMS-free detection. However, when directly transferred to BEV tasks, this design suffers a substantial performance drop under the end-to-end setting. Removing the post-processing step from the baseline results in a decline of 7.6% mAP and 3.9% NDS for FastBEV, and 7.3% mAP and 3.6% NDS for BEVDet4D. In contrast, when replacing the baseline detection head with our proposed Spe-BEVHead, the model maintains reliable performance even without any post-processing. Moreover, applying max-pooling NMS further boosts the results to match those obtained with standard NMS, demonstrating the robustness and adaptability of our design. Fig. 4 visualizes the predictions of FastBEV equipped with the center-based head and the proposed Spe-BEVHead without any post-processing. Compared to the center-based head, Spe-BEVHead clearly suppresses redundant predictions.

4.4. Ablation Study

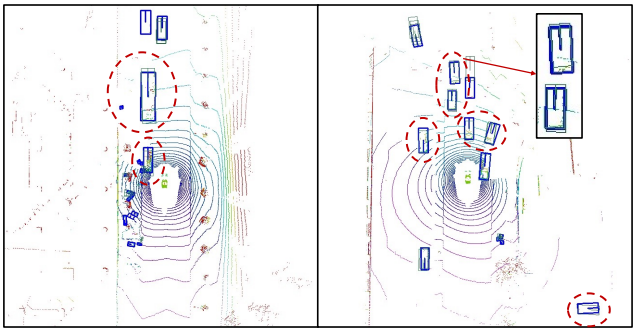
Table 3. Ablation of Spe-BEVHead components. DB: dual-branch architecture; RBK: Rotated Box Kernel; LRRM: Local Response Refinement Module.

DB	RBK	LRRM	NDS↑	mAP↑
			34.5	21.9
✓			36.6	26.0
✓	✓		37.3	26.0
✓	✓	✓	**37.9**	**26.2**

We conduct end-to-end ablation studies on the FastBEV



(a) FastBEV with Center-based Head



(b) FastBEV with Spe-BEVHead

Figure 4. Visualization of predictions without post-processing. Blue boxes denote ground truths, and green boxes denote predictions. Zoom in for better details.

Table 4. Ablation of Rotated Box Kernel. "L" and "Q" denote linear decay and quadratic decay, respectively.

Decay Function	Boundary Value	NDS↑	mAP↑
L	0.1	37.5	26.2
L	0.3	37.2	26.0
L	0.5	37.3	25.9
Q	0.01	37.4	26.0
Q	0.1	**37.9**	**26.2**
Q	0.3	37.4	25.7
Q	0.5	37.2	25.7

baseline to evaluate the impact of Spe-BEVHead components and hyperparameters.

Component Analysis. To validate the contribution of each component in the proposed Spe-BEVHead, we progressively introduce its modules. As depicted in Tab. 3, dual-branch architecture boosts the performance by 2.1% NDS and 4.1% mAP. Rotated Box Kernel boosts the performance by 0.7% NDS. Local Response Refinement Module boosts the performance by 0.6% NDS and 0.2% mAP.

Rotated Box Kernel. We conduct experiments to evaluate different configurations when applying the Rotated Box Kernel as shown in Tab. 4. As the boundary value($v_{\min}$ in Eq. (3)) within the RBK coverage range increases from 0.1 to 0.3 and 0.5, both mAP and NDS exhibit a slight decrease.

Table 5. Ablation of Local Response Refinement Module. "Repeat" denotes the number of repetitions of the LRRM, while "Local Size" denotes the receptive field size of the AMA operator in each LRRM.

Repeat	Local Size	NDS↑	mAP↑
×1	(3,)	37.2	25.9
×2	(3, 3)	**37.9**	**26.2**
×2	(3, 5)	37.3	25.4
×2	(5, 5)	37.3	25.8
×3	(3, 3, 3)	37.8	25.8

Table 6. Ablation of different local suppression operators.

Φ	Repeat	NDS↑	mAP↑
Max Pooling	×1	37.0	25.7
Max Pooling	×2	37.4	25.9
AMA	×1	37.2	25.9
AMA	×2	**37.9**	**26.2**

Nevertheless, no performance gain was observed when $v_{\min}$ was further reduced to 0.01. The model achieves optimal performance when the boundary value is set to 0.1 and a quadratic decay function is employed.

Local Response Refinement Module. We conduct experiments to analyze how the number of stacked Local Response Refinement Modules and the receptive scope of the Adaptive Mean Attenuation operator affect the performance. As shown in Tab. 5, the best performance is achieved when the LRRM is stacked twice and the kernel size of the AMA is set to 3×3. The 2D object detection network DeFCN [31] utilizes max pooling to promote the learning of sharper features. To demonstrate the advantage of our AMA operator, we substitute the AMA operator in the LRRM with max pooling. As shown in the Tab. 6, the LRRM yields greater performance improvements than max pooling, primarily because it enforces stronger suppression on local non-maximum responses.

5. Conclusion

Conclusion. In this work, we analyze the limitations of the commonly used center-based head in LSS-based BEV detection models and propose Spe-BEVHead, a detection head specifically tailored for BEV detection. Simply replacing the original center-based head with Spe-BEVHead yields direct performance improvements across existing BEV detection baselines.

Limitations. Although Spe-BEVHead mitigates the performance drop in end-to-end detection, it still lags behind NMS-based results due to the intrinsic challenges of achieving fully end-to-end detection within convolutional architectures. Future work will focus on enhancing this capability to eliminate the GPU-unfriendly NMS stage.

Acknowledgments

This work was supported by National Science and Technology Major Project 2022ZD0119401, National Natural Science Foundation of China (Nos. 62525103, 62571294, 62441614), Beijing Natural Science Foundation (L252009) and CCF-DiDi GAIA Collaborative Research Funds.

References

[1] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11618–11628, 2020. 2, 6

[2] 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, 2020. 3

[3] Yiqun Chen, Qiang Chen, Qinghao Hu, and Jian Cheng. Date: Dual assignment for end-to-end fully convolutional object detection. *arXiv preprint arXiv:2211.13859*, 2022. 3

[4] 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*, pages 770–778, 2016. 3, 6

[5] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022. 2, 3, 6, 7

[6] Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 1, 2, 3, 6, 7

[7] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19702–19712, 2023. 3

[8] 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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1042–1050, 2023. 3

[9] Youngwan Lee, Joong-won Hwang, Sangrok Lee, Yuseok Bae, and Jongyoul Park. An energy and gpu-computation efficient backbone network for real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 3

[10] 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*, pages 13619–13627, 2022. 3

[11] Yinhao Li, Han Bao, Zheng Ge, Jinrong Yang, Jianjian Sun, and Zeming Li. Bevstereo: Enhancing depth estimation in multi-view 3d object detection with temporal stereo. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1486–1494, 2023. 3, 6, 7

[12] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1477–1485, 2023. 1, 2, 3, 7

[13] Yangguang Li, Bin Huang, Zeren Chen, Yufeng Cui, Feng Liang, Mingzhu Shen, Fenggang Liu, Enze Xie, Lu Sheng, Wanli Ouyang, et al. Fast-bev: A fast and strong bird's-eye view perception baseline. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):8665–8679, 2024. 2, 3, 6, 7

[14] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European Conference on Computer Vision*, pages 1–18, 2022. 1, 2, 3

[15] Zhenxin Li, Shiyi Lan, Jose M Alvarez, and Zuxuan Wu. Bevnext: Reviving dense bev frameworks for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20113–20123, 2024. 3

[16] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2980–2988, 2017. 6

[17] Feng Liu, Tengteng Huang, Qianjing Zhang, Haotian Yao, Chi Zhang, Fang Wan, Qixiang Ye, and Yanzhao Zhou. Ray denoising: Depth-aware hard negative sampling for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 200–217, 2024. 3

[18] Hou-I Liu, Christine Wu, Jen-Hao Cheng, Wenhao Chai, Shian-Yun Wang, Gaowen Liu, Hugo Latapie, Jhih-Ciang Wu, Jenq-Neng Hwang, Hong-Han Shuai, et al. Monotakd: Teaching assistant knowledge distillation for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22266–22275, 2025. 1

[19] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. PETR: position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548, 2022. 2, 3

[20] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3239–3249, 2023. 3

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[22] 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. 3

[23] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learn-

ing and geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7074–7082, 2017. 1

[24] Rishubh Parihar, Srinjay Sarkar, Sarthak Vora, Jogendra Nath Kundu, and R Venkatesh Babu. Monoplace3d: Learning 3d-aware object placement for 3d monocular detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6531–6541, 2025. 1

[25] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, pages 194–210, 2020. 2, 3

[26] 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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6520–6530, 2025. 1

[27] Zhangyang Qi, Jiaqi Wang, Xiaoyang Wu, and Hengshuang Zhao. OCBEV: object-centric BEV transformer for multi-view 3d object detection. In *International Conference on 3D Vision*, pages 1188–1197, 2024. 3

[28] Peize Sun, Yi Jiang, Enze Xie, Wenqi Shao, Zehuan Yuan, Changhu Wang, and Ping Luo. What makes for end-to-end object detection? In *International Conference on Machine Learning*, pages 9934–9944, 2021. 3, 5

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 2

[30] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, et al. Yolov10: Real-time end-to-end object detection. *Advances in Neural Information Processing Systems*, 37:107984–108011, 2024. 3

[31] Jianfeng Wang, Lin Song, Zeming Li, Hongbin Sun, Jian Sun, and Nanning Zheng. End-to-end object detection with fully convolutional network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15849–15858, 2021. 3, 5, 8

[32] Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3621–3631, 2023. 3

[33] Yue Wang, Vitor Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. DETR3D: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2021. 2, 3

[34] 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. 3

[35] Enze Xie, Zhiding Yu, Daquan Zhou, Jonah Philion, Anima Anandkumar, Sanja Fidler, Ping Luo, and Jose M Alvarez. M^2bev: multi-camera joint 3d detection and segmentation with unified birds-eye view representation. *arXiv preprint arXiv:2204.05088*, 2022. 3

[36] Ziteng Xue, Mingzhe Guo, Heng Fan, Shihui Zhang, and Zhipeng Zhang. Corrbev: Multi-view 3d object detection by correlation learning with multi-modal prototypes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27413–27423, 2025. 3

[37] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, Jie Zhou, and Jifeng Dai. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17830–17839, 2023. 3

[38] Fan Yang, Hui Chen, Yuwei He, Sicheng Zhao, Chenghao Zhang, Kai Ni, and Guiguang Ding. Geometry-guided domain generalization for monocular 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6467–6476, 2024. 1

[39] Yung-Hsu Yang, Luigi Piccinelli, Mattia Segu, Siyuan Li, Rui Huang, Yuqian Fu, Marc Pollefeys, Hermann Blum, and Zuria Bauer. 3d-mood: Lifting 2d to 3d for monocular open-set object detection. *arXiv preprint arXiv:2507.23567*, 2025. 1

[40] Huizai Yao, Sicheng Zhao, Shuo Lu, Hui Chen, Yangyang Li, Guoping Liu, Tengfei Xing, Chenggang Yan, Jianhua Tao, and Guiguang Ding. Source-free object detection with detection transformer. *IEEE Transactions on Image Processing*, 2025. 3

[41] Xin Ye, Burhaneddin Yaman, Sheng Cheng, Feng Tao, Abhirup Mallik, and Liu Ren. Bevdiffuser: Plug-and-play diffusion model for BEV denoising with ground-truth guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1495–1504, 2025. 3

[42] Jinqing Zhang, Yanan Zhang, Yunlong Qi, Zehua Fu, Qingjie Liu, and Yunhong Wang. Geobev: Learning geometric bev representation for multi-view 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9960–9968, 2025. 2, 3, 6, 7

[43] 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

[44] 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

[45] Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16965–16974, 2024. 3

[46] Hongyu Zhou, Zheng Ge, Zeming Li, and Xiangyu Zhang. Matrixvt: Efficient multi-camera to bev transformation for 3d perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8548–8557, 2023. 3

[47] Qiang Zhou and Chaohui Yu. Object detection made simpler

by eliminating heuristic nms. *IEEE Transactions on Multimedia*, pages 9254–9262, 2023. 3

[48] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 2, 3, 6

[49] Benjin Zhu, Zhengkai Jiang, Xiangxin Zhou, Zeming Li, and Gang Yu. Class-balanced grouping and sampling for point cloud 3d object detection. *arXiv preprint arXiv:1908.09492*, 2019. 6

[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. 3

[51] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6748–6758, 2023. 3