LSSInst: Improving Geometric Modeling in LSS-Based BEV Perception with Instance Representation

Weijie Ma[1] ‡ Jingwei Jiang[1] ‡ Yang Yang[1] Zehui Chen[2] Hao Chen[1] *
[1]Zhejiang University [2]University of Science and Technology of China

Abstract

With the attention gained by camera-only 3D object detection in autonomous driving, methods based on Bird-Eye-View (BEV) representation especially derived from the forward view transformation paradigm, i.e., lift-splat-shoot (LSS), have recently seen significant progress. The BEV representation formulated by the frustum based on depth distribution prediction is ideal for learning the road structure and scene layout from multi-view images. However, to retain computational efficiency, the compressed BEV representation such as in resolution and axis is inevitably weak in retaining the individual geometric details, undermining the methodological generality and applicability. With this in mind, to compensate for the missing details and utilize multi-view geometry constraints, we propose LSSInst, a two-stage object detector incorporating BEV and instance representations in tandem. The proposed detector exploits fine-grained pixel-level features that can be flexibly integrated into existing LSS-based BEV networks. Having said that, due to the inherent gap between two representation spaces, we design the instance adapter for the BEV-to-instance semantic coherence rather than pass the proposal naively. Extensive experiments demonstrated that our proposed framework is of excellent generalization ability and performance, which boosts the performances of modern LSS-based BEV perception methods without bells and whistles and outperforms current LSS-based state-of-the-art works on the large-scale nuScenes benchmark.

1. Introduction

As a crucial component in 3D perception, 3D object detection can be applied in various fields, such as autonomous driving and robotics. Although LiDAR-based 3D detection methods [21, 43, 48, 55] are verified as having remarkable performance, research in camera-based methods [16, 17, 27, 35, 45, 46] has received increasing attention in recent years. The reasons can be attributed not only to the lower deployment cost but also to the advantages offered by long-range distance and the identification of visual road elements [27, 34]. However, unlike LiDAR sensors that provide direct and accurate depth information, detecting objects solely based on camera sensor images poses a significant challenge. Thus, how to utilize multi-view images to build up effective representations has become a key issue. Recently, significant progress has been achieved in methods that utilize the bird's-eye view (BEV) whose view transformation can be mainly categorized as forward type [11, 24, 25, 34] based on lift-splat-shoot (LSS) [35] and backward type [27, 49] based on the learnable BEV query. Due to its purely implicit aggregation by uninterpretable but forcibly dense queries, the backward type shows lower performance and expansibility [13], enabling the LSS-based forward type to become mainstream in modern BEV paradigms for camera-only 3D detection at present. Based on the LSS hypothesis and the fact that most objects in the scene are close to the ground, LSS-based BEV provides a perspective with minimal parallax ambiguity and information loss in observing the objects as a whole. Illustrated by Fig. 1 (a), these methods lift images into frustums based on depth distribution prediction and splat them into a

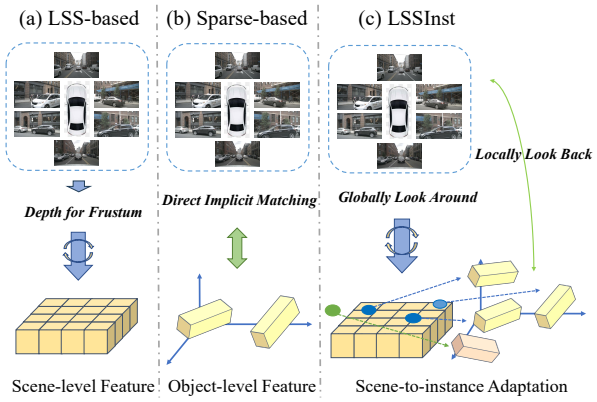

Figure 1. The conceptual comparison of LSSInst with previous camera-based fashions.

* is the corresponding author and ‡ denotes the equal contribution.

Table 1. The per-category AP comparison between two typical fashions with equivalent detection ability (ΔmAP less than **0.5%**) methods on the nuScenes `test` set.

	Group 1		Group 2	
Methods	**BEVDet [17]**	**Spatial-DETR [10]**	**CFT-BEV3D [19]**	**DETR3D [46]**
mAP	42.4	42.5	41.7	41.2
car	**0.643**	0.610	**0.628**	0.603
truck	**0.350**	0.332	**0.348**	0.333
bus	**0.358**	0.330	**0.347**	0.290
pedestrian	0.411	**0.462**	0.416	**0.455**
bicycle	0.296	**0.327**	0.299	**0.308**
traffic_cone	0.601	**0.629**	0.596	**0.627**
barrier	**0.614**	0.582	**0.607**	0.565

BEV space, gathering information from multiple 2D views for a global representation of the scene. This representation is in the form of a planar view with compressed height (z-axis) and reduced resolution to ensure computational efficiency. The BEV feature benefits from its holistic representation and dense feature space, making it well-suited for capturing the scene's structure and data distribution. However, the geometrically-compressed nature of the BEV representation such as resolution and axis reduction, inherently limits its ability to provide precise 3D position descriptions of objects or fully exploit detailed features for object matching particularly in the 3D detection task which requires accurate prediction of 3D object bounding boxes. Meanwhile, as another typical fashion and shown in Fig. 1 (b), sparse-based methods [3, 31, 46] skip the BEV formulation and directly leverage object-level representations and exploit the 3D geometric prior to regress object bounding boxes from the 3D perspective. However, due to the initialization semantic dispersity [22], especially in more complex 3D perception, they fail to capture abundant object-aware features from the image at once in comparison with the BEV feature that fits adequate semantic information in the scene, resulting in lower overall performance than the contemporaneous BEV-based methods. Following this, there are some interesting and corroborative findings in the per-class AP comparison between the two aforementioned fashions as shown in Tab. 1. Notably, considering the practical variety such as data augmentation and training strategies, the difference between the overall mAP values of selected methods in the same group is strictly less than **0.5%** which ensures both detection abilities are equivalent. We can observe that there is the same AP tendency among the classes. Specifically, BEV representation seems more attentive to regular objects (car, bus, truck, barrier) with distinct movements or common positions in the scene, with relative insensitivity to the objects (pedestrian, bicycle, traffic cone) with uncertain trajectories or dispersed locations, which further proves its characteristics of fitting data distribution and leaning to the scene-level focus. Inspired by this, to brighten the complementary synergy of both fashions and make up for the missing details in the representation formulation of current LSS-based BEV perception as well as utilize multi-view geometry constraints, we are motivated to propose **LSSInst**, incorporating the sparse instance-level representations based on the scene-level representations to look back for more detailed feature with geometric matching. As illustrated in Fig. 1 (c), based on the global scene-level pre-feature, instance-level features are pushed to look back at the image locally, focusing on more fine-grained pixel features and allowing for flexible geometric matching, which ultimately generates a final perception result that combines globally-semantic and locally-geometric information.

However, this collaboration also poses challenges, as the most straightforward solution of naively sharing the bounding box proposal is intuitively and experimentally failed [1]. As aforementioned, traditional sparse-based detection methods suffer from initialization semantic dispersity and inadequate semantic understanding of the scene, the above solution would sever the coherence with the dense representations. With this in mind, we propose the instance adapter module to establish semantic coherence between the scene and instances and an instance branch for detection. The instance adapter module generates multiple sparse queries and their corresponding 3D boxes through multi-level adaptive aggregation. The instance branch focuses on fine-grained sparse feature extraction and geometric matching using prepared inputs, such as box embeddings and spatiotemporal sampling and fusion. On the nuScenes dataset, our LSSInst method demonstrates strong generalization ability. Compared to other typical LSS-based methods, LSSInst achieves significant improvements in mAP. Specifically, it outperforms BEVDet by 5.0%, BEVDepth by 2.2%, BEVStereo by 2.6%, and surpasses the state-of-the-art LSS-based method SOLOFusion by 1.6%.

Our main contributions can be concluded as follows: i) We proposed LSSInst, a two-stage framework that improves the geometric details in LSS-based BEV perception with instance representations; ii) We proposed the instance adapter to maintain the BEV-to-instance semantic coherence and a newly-designed instance branch to look back and aggregate features spatiotemporally for improvement; iii) The proposed framework was verified with great generalization ability and surpassed the state-of-the-art LSS-based methods by extensive experimental results.

2. Related Work

2.1. LSS-based BEV Perception

As BEV has proved to be an effective representation for multi-view 3D detection, LSS-based methods that benefit from the explicit formulation process and superior performance become the recent mainstream paradigm. LSS [35] is proposed for an end-to-end view transform archi-

[1] See Sec. 4.5 and Tab. 6 for more details

tecture that lifts images into frustums by predicting depth distribution and splats them into a BEV representation. Then BEVDet [17] incorporates exclusive data augmentation techniques for the detection extension. BEVDepth [25] and BEVStereo [24] improved the depth accuracy by introducing an extra monocular depth network supervised by corresponding LiDAR depth, and multi-view stereo matching between adjacent frames, while BEVDistill [5] chose to the model-level distillation from LiDAR. OA-BEV [6] and SA-BEV [52] enhanced the utilization of depth, which integrated the 3D voxel network based on the additional proposal from the 2D detection network and proposed a depth and semantic fusion module respectively for a more enhanced feature. Besides, several works started to perceive the shortage of the current view transformation assumption. AeDet [11] introduced the positional compensation for existing coordinate projection while FB-BEV [28] integrated a novel forward-backward view transformation module that partially alleviates the projection issues. SOLOFusion [34] further unified long-term temporal information based on the short-term temporal optimization with Gaussian top-k sampling to boost performance.

Despite these methods making efforts to chase a flawless BEV representation from the LSS process, due to the avoidable depth error and the compressed nature of pooling operations, the yielded BEV representation is weak in retaining the individual geometric details, hence we differently focus on adapting the BEV representation into better geometrical modeling.

2.2. Instance-level Representation Integration in Camera 3D Detection

Integrating instance-level representation is ubiquitous in camera 3D detection to enhance perceptual ability. FQNet [30] is a three-stage framework for monocular detection that first locally searches for potential boxes and then follows a Fast-RCNN-like way [12] to aggregate the massive object candidate globally for location prediction. [23] apply a similar way in stereo 3D detection. They first borrow DSGN [4] to locally search for possible proposals and then establish the Vernier network to globally form the confidence map based on the stereo pair. Similarly in multi-view 3D detection, for the sparse-based methods [31, 46] which solely depend on the query decoding, there tends to be a weak correlation and slow convergence between foreground tokens and queries due to the perspective inconsistency. Focal-PETR [44] adopts extra 2D instance-level supervision to adaptively focus object queries on discriminative foreground regions. For BEV query-based methods, BEVFormer V2 [49] used the extra 3D perspective network such as [33, 46] to generate coarse instance features to serve as auxiliary proposals. Unlike they borrow instance-level features in a **bottom-up (i.e., local-to-global)** way, LSSInst uses a totally different **top-down (i.e., global-to-local)** way for improvement.

2.3. Two-stage 3D Object Detector

The two-stage design has been widely explored in the 3D detection domain and proved to be effective, whose multi-step workflow is favorable for more accurate prediction. For LiDAR 3D detection, drawing inspiration from 2D detection [2, 12, 37], a two-stage LiDAR detector typically generates Regions of Interest (RoIs) in the first stage and then refines these initial predictions in the second stage [36, 38, 50]. To address this issue of losing the ability to encode the geometric information of the proposals, point cloud pooling operations on the RoI [39] or virtual points with boundary offsets [26] were introduced. Instead of pooling from point features, Voxel R-CNN [9] designs a voxel-RoI pooling module to pool directly from voxel and BEV feature space according to the RoI-grid. To increase speed, CenterPoint [51] simplifies the pooling module by sampling five keypoints from BEV features using bilinear interpolation while RSN [41] uses a foreground segmentation as a first stage to sparsify the point clouds, thereby enhancing the efficiency of the second stage sparse convolution. For camera 3D detection, MonoDIS [40] extracts features from 2D bounding boxes for subsequent 3D bounding box regression and introduces a disentangling transformation to supervise the detection loss separately for 2D and 3D. SimMod [54] utilized a DETR3D head [46] to iteratively refine 2D-level object proposals output from a monocular network. BEVFormer v2 [49] extends BEVFormer [27] into the second stage by incorporating a first-stage 3D perspective detection network that directly supervises the image backbone network, leveraging both BEV and perspective information. Among current two-stage methods, they seek more refinement for jointly aggregating coarse samples because their first stage primarily relies on perspective views. Instead, we focus on the subsequent refinement of proposals with holistic semantics derived together from the scene-level layout.

3. Methodology

Utilizing instance-level representations on the basis of the scene-level BEV to excavate more detailed features and geometric information is of practical significance for generalized 3D perception. In this work, we propose LSSInst, which looks back for the more geometry-aware and fine-grained target feature extraction to bridge the adaptation between scene-level and instance-level 3D representations. The overview of our framework is shown in Fig. 2, and we organize the remaining part as follows. Firstly, Sec. 3.1 briefly introduces the BEV branch. Next, Sec. 3.2 introduces the instance adapter module, and the instance branch is given in Sec. 3.3.

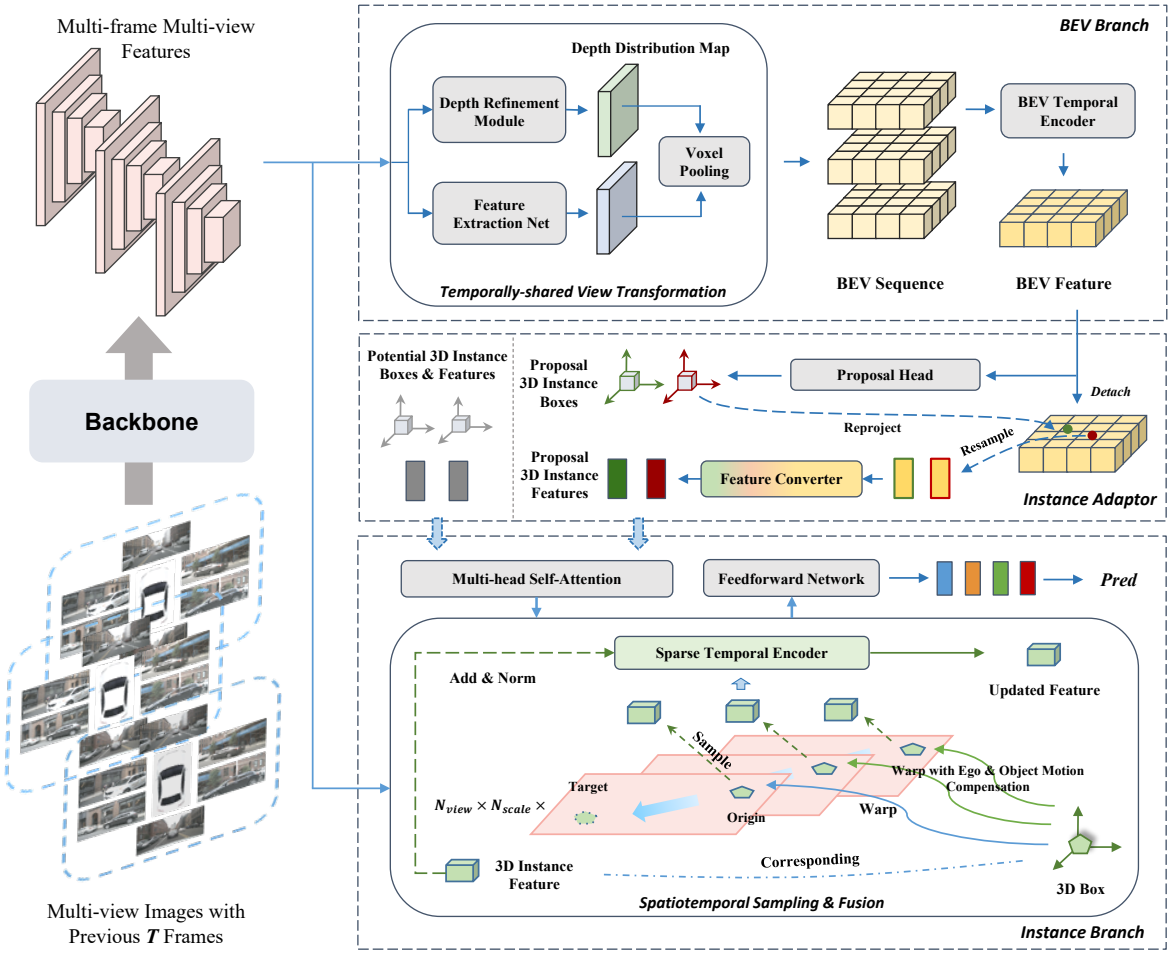

Figure 2. Overview of LSSInst. The multi-view images with previous T frames are fed into the backbone network for the image features. BEV branch looks around the image feature to generate the BEV feature by view transformation and temporal encoding. Instance adapter aggregates the sparse object-aware feature from the BEV feature and prepares the multiplicate 3D query combination. Instance branch looks back at the image feature and perfects the sparse feature by spatiotemporal sampling and fusion. Lastly, the model makes the final prediction based on the updated output.

3.1. BEV Branch: Looking around for scene-level representation

The multi-view sequential images with the previous T frames are first input into the 2D image backbone network for feature extraction. Then the BEV branch receives the extracted image feature $F_{img} \in \mathbb{R}^{N_v \times N_s \times C \times H \times W}$ and functions as an around looker, translating F_{img} from 2D camera views to BEV for preliminary scene-level representation $F_\beta \in \mathbb{R}^{C \times H_\beta \times W_\beta}$, where N_v, N_s denote the camera-view and scale number respectively. This branch can be briefly divided into temporally-shared view transformation for BEV generation and BEV sequence fusion. The 2D-to-BEV view transformation is naturally based on LSS paradigm, which can be mainly concluded as the depth refinement module (DRM), feature extraction net, and voxel pooling. For the best version of the framework, we adopted the Gaussian-spaced top-k stereo [34] for a better depth distribution map before voxel pooling. After the shared view transformation, a sequence of BEV representations will be aligned into current time t and fed to the BEV temporal encoder to form the final current BEV. Here the encoder is designed as a very lightweight residual network for dimension reduction only.

3.2. Instance Adapter: Scene-to-instance adaptation

For the sake of preserving a coherent and solid semantic consistency between BEV and instance representations, we propose the instance adapter module to eliminate the gaps in the position description and space discrepancy. Since the

BEV feature F_β is a scene-level representation surrounding the ego car, there is redundancy and inflexibility in modeling instance-level features. To that end, the proposed adapter module first performs a reprojection of the proposal box coordinates $P_o \in \mathbb{R}^{N_\beta \times 3}$ obtained through the BEV proposal head, returning to the BEV-recognized position $P_\beta \in \mathbb{R}^{N_\beta \times 2}$ to resample the object-related features. Here N_β denotes the number of BEV proposals. Given the BEV point-cloud range R_β with the corresponding voxel size S_v and up-sampling factor σ, we can formulate the 2D reprojected coordinate as follow:

$$[P_\beta, z_\beta] = (P_o - R_\beta)/(\sigma S_v) \tag{1}$$

here z_β denotes the z-axis homogeneous term and is actually a constant of 1. Moreover, due to the overfitting bias in BEV, the focused area may deviate from the actual object position. Inspired by deformable attention [56], the adapter module incorporates the learnable offsets based on the original focused feature $F_\beta^{'} \in \mathbb{R}^{N_\beta \times C}$ for misalignment compensation by exploring more semantically-aware regions. Suppose $i \in \{1, 2, ..., N_\beta\}$ denotes an arbitrary element index of $F_\beta^{'}$ and its aggregated instance-wise feature F_α^i can be formulated by

$$F_\alpha^i = \mathbf{W}_\alpha \sum_{k=1}^{K} A_k^i \cdot \mathbf{W}_\alpha^{'} \Phi\left(F_\beta, P_\beta^i + \Delta P_{\beta k}^i\right) \tag{2}$$

where $\mathbf{W}_\alpha, \mathbf{W}_\alpha^{'} \in \mathbb{R}^{C \times C}$ are the weight matrix of linear projections and k indexes the resampled keys with the total resampled key number K ($K << H_\beta W_\beta$). $\Delta P_{\beta k}^i \in \mathbb{R}^2$ is the learnable offset and A_k^i is the normalized attention weight of the kth sampling point with $\sum_{k=1}^{K} A_k^i = 1$, which are both linearly projected over $F_\beta^{'}$. Φ denotes the bilinear interpolation function.

Having said that aforementioned, there still remains the inherent space discrepancy between BEV encoding space and the 3D sparse space suitable for looking back at the image feature. Therefore, we first introduce an extremely shallow convolutional feature converter f_{cnv} to reparameterize the aggregated features for the inter-space narration. Meanwhile, even with the extensive aggregation and enhancement based on BEV attentive features, a portion of irregular or separated objects cannot be detected due to the BEV overfitting of regular objects and relatively rough perception granularity. Therefore, we introduce extra learnable queries $F_\gamma \in \mathbb{R}^{N_\gamma \times C}$ and reference boxes independent of BEV proposals, named potential 3D instances and boxes, aimed at capturing the potential BEV-insensitive objects and learning a BEV-agnostic 3D spatial prior. Thus, we can get the multiplicate sparse feature $F_\chi \in \mathbb{R}^{N \times C}$ and here we let $N = N_\beta + N_\gamma$ for simplicity. The whole F_χ formulation can be derived by

$$F_\chi = \left\{ f_{cnv}\left(\{F_\alpha^i\}_{i=1}^{N_\beta}\right), F_\gamma \right\} \tag{3}$$

3.3. Instance Branch: Looking back for instance-level representation

Given the sequential image features $\{F_{img}^t\}_{t=0}^{T_\chi}$ ($T_\chi \leq T$) from the image backbone network and the sparse instance features F_χ with corresponding 3D boxes $P_{box} \in \mathbb{R}^{N \times C_{box}}$ from the instance adapter, the instance branch will spatially and temporally look back the image feature based on the referenced box coordinates and iteratively extract the abundant but more fine-grained representations to update pre-features. This branch can be roughly regarded as a multi-layer Transformer-decoder-like [42] module for 3D detection, which is briefly divided into two parts: box-level offset and embedding, as well as spatiotemporal sampling and fusion.

Box-level Offset and Embedding Different from the previous DETR-like 3D approaches such as DETR3D [46], Polarformer [20], VEDet [3] which iteratively refine only with 3D coordinate offset regression, the instance branch adopts the box-level offset regression based on P_{box}. With this convenience, we can encode all the geometric-aware information of the entire box to substitute the transitional positional encoding, thereby expanding and enriching the space of feature expression rather than the superficial positional level. With it combined with the sparse instance features, there will be more geometric priors and implicit compensation in subsequent attention interactions. Precisely, we first categorize P_{box} based on the element semantic of box dimension into four divisions, which are position $P_{pos} \in \mathbb{R}^{N \times 3}$ (*i.e.*, x, y, z), scale $P_{sca} \in \mathbb{R}^{N \times 3}$ (*i.e.*, w, l, h), velocity $P_{vel} \in \mathbb{R}^{N \times 2}$ (*i.e.*, v_x, v_y), and orientation $P_{ori} \in \mathbb{R}^{N \times 2}$ (*i.e.*, $sin(\theta_{yaw}), cos(\theta_{yaw})$) respectively. Then we introduce five separated linear projections $\{\mathbf{E}_{l3}^j\}_{j=1}^2 \in \mathbb{R}^{3 \times C}$, $\{\mathbf{E}_{l2}^j\}_{j=1}^2 \in \mathbb{R}^{2 \times C}$ and $\mathbf{E}_g \in \mathbb{R}^{C \times C}$ for comprehensive encoding, of which the former four embed every category locally and the last one embeds them globally. The final box embedding $G_\chi \in \mathbb{R}^{N \times C}$ can be formulated by

$$G_\chi = \mathbf{E}_g \left[\sum_{j=1}^{2} \mathbf{E}_{l3}^j \left(P_{d3}^j\right) + \sum_{j=1}^{2} \mathbf{E}_{l2}^j \left(P_{d2}^j\right) \right] \tag{4}$$

where P_{d3}^j, P_{d2}^j denote the three and two-dimensional categorized element respectively.

Spatiotemporal Sampling and Fusion The sparse feature F_χ with the box embedding G_χ will be updated by the

spatial and temporal sampling after being fed into the multi-headed self-attention block [42]. Given the corresponding 3D coordinate $P_\chi \in \mathbb{R}^{N\times 3}$ from P_{box}, we first warp it from the 3D system to the 2D correspondence $p_\chi \in \mathbb{R}^{N\times 2}$ at the current time by the intrinsic and extrinsic matrix.

On the spatial hand, intending to access the target region, we sample the original feature to intermediately regress the existing offset from p_χ to the target. To expand the search breadth, we extend the sampling points analogous to Eq. 2 and enlarge the proportion of residual addition with a weight η. On the other temporal hand, with time going, there exists ego-car motion and object motion in the autonomous driving scenario, which requires compensation before sampling. In light of the short term in this sparse temporal stereo, *i.e.*, T_χ is a small positive integer, we approximate the object motion as a uniform rectilinear motion. Thus, we first compensate p_χ with current 2D velocity $p_{vel} \in \mathbb{R}^{N\times 2}$, *i.e.*, the 2D correspondence of P_{vel} and then warp it into every coordinate system as $\{p_\chi^t\}_{t=1}^{T_\chi}$ of per historical time by the medium transition in the global world coordinate system. Then the per-frame sampled feature $F_{\delta t}, t \in \{0, 1, ..., T_\chi\}$ is formulated by

$$F_{\delta t} = \mathbf{W}_\chi \sum_{k=1}^{K} A_{kt} \cdot \mathbf{W}_\chi^{'} \Phi[F_{img}, (\mathbf{M}_t(p_\chi - \tau t \cdot p_{vel}) + \Delta p_{\chi kt})] \tag{5}$$

where τ is the time interval between every two adjacent frames, and $\mathbf{M}_t$ is the ego-motion transform matrix from current time to previous t time. Then the multi-frame features are fed into the sparse temporal encoder f_{enc}, a naive three-layer MLP, for temporal iterative fusion. Based on our approximation, the projection error will increase with t larger. Thus we let λ denote a constant in the range [0, 1], which is introduced for long-term suppression. With iterative fusion $F_{\delta(t-1)} \leftarrow f_{enc}(\{F_{\delta(t-1)}, \lambda F_{\delta t}\})$, we can get the final sparse sampled feature F_δ from $\{F_{\delta t}\}_{t=0}^{T_\chi}$. The whole box-level offset δ_χ can be calculated as follows:

$$F_\chi \leftarrow F_\chi + \eta F_\delta \qquad \delta_\chi = f_{reg}\{F_\chi + G_\chi\} \tag{6}$$

here f_{reg} is the box offset regression function of every layer. Notably, we omit the calculation in the scale and view level of F_{img} for simplicity.

4. Experiments and Analysis

4.1. Experimental Settings

Dataset We conducted extensive experiments on the nuScenes 3D detection benchmark [1], a large-scale dataset in the autonomous driving scene. This benchmark consists of 1,000 autonomous driving scenes, with each scene spanning approximately 20 seconds. The dataset is divided into 850 scenes for training (`train`) or validation (`val`) purposes and 150 scenes for testing (`test`). Each frame in the dataset contains six cameras capturing surrounding views, along with a LiDAR-generated point cloud. The dataset provides annotations for up to 1.4 million 3D bounding boxes across 10 different classes.

Implementation Details We implemented our network framework utilizing the open-source MMDetection3D [7] in PyTorch. The learning rate, optimizer, and data augmentation methods used were the same as those in BEVDepth. By default, We set the image size to 256 $\times$ 704 and utilized ResNet50 [14], pretrained on ImageNet [8], as the image backbone. The size of the BEV feature in all our experiments was set to 128 $\times$ 128. Here we set $k = 6$, $T_\chi = 3$, $\eta = 3$. The feature dimension C is 256 and the box dimension C_{box} is 10. The perception ranges for the X and Y axis was [-51.2m, 51.2m], and the resolution of each BEV grid was 0.8m. The time interval τ is 0.5s, and long-term suppression λ is 0.6.

4.2. Benchmark Results

We compared our approach with LSS-based and two-stage state-of-the-art methods on the nuScenes `val` and `test` sets. The main results are presented in Tab. 2 and Tab. 3 respectively. On the `val` set, we evaluated the performance of LSSInst against other models with the same setting and without the CBGS strategy and future frame usage. The results clearly showcased the superiority of LSSInst, as it outperformed the current LSS-based SOTA, SOLOFusion by a margin of 1.6% in mAP and 1.7% in NDS, and the current two-stage SOTA, BEVFormer v2 by a margin of 3.4% in mAP and 1.6% in NDS. On the `test` set, our LSSInst achieves an mAP of 54.6% and an NDS of 62.9% without any additional augmentation, outperforming all LSS-based methods. Such improvements demonstrate the effectiveness of our LSSInst for improving LSS-based BEV perception with instance representations.

4.3. Generalization Ability and Geometric-Wise Boost

To demonstrate the generalization ability of our LSSInst method, we selected prominent LSS-based methods as the BEV branch of LSSInst. The results are presented in Tab. 4. The table reveals that our LSSInst achieves notable improvements in mAP and NDS compared to standalone BEV detectors at a minor cost. In spite of the impressive detection enhancement with 2-5% mAP and NDS, the corresponding costs increase by an acceptable margin. In particular, among all the methods, there is a significant improvement in mATE [2], mASE, and mAOE, indicating

[2] Actually in the mATE column of Tab. 4, 0.620 mATE in 0.422 mAP also beats 0.609 mATE in 0.402 mAP, please see Supplementary Material.

Table 2. Comparison results of LSS-based and two-stage detectors on 3D detection on the nuScenes `val` set. † denotes the performance without future frames for a fair comparison.

Method	Backbone	Image Size	mAP↑	NDS↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
BEVDet [17]	ResNet50	256 × 704	0.283	0.350	0.773	0.288	0.698	0.864	0.291
BEVDet4D [16]	ResNet50	256 × 704	0.322	0.457	0.703	0.278	0.495	0.354	0.206
BEVDepth [25]	ResNet50	256 × 704	0.330	0.436	0.702	0.280	0.535	0.553	0.227
SimMOD [54]	ResNet50	800 × 1333	0.331	0.427	0.721	0.267	0.401	0.810	0.184
BEVStereo [24]	ResNet50	256 × 704	0.346	0.452	0.659	0.277	0.550	0.498	0.228
AeDet [11]	ResNet50	256 × 704	0.358	0.473	0.655	0.273	0.493	0.427	0.216
SA-BEV [52]	ResNet50	256 × 704	0.370	0.488	0.660	0.269	0.470	0.353	0.218
FB-BEV [28]	ResNet50	256 × 704	0.378	0.498	0.620	0.273	0.444	0.374	0.200
BEVFormer v2 [49] †	ResNet50	640 × 1600	0.388	0.498	0.679	0.276	0.417	0.403	0.189
SOLOFusion [34]	ResNet50	256 × 704	0.406	0.497	0.609	0.284	0.650	0.315	0.204
LSSInst	ResNet50	256 × 704	**0.422**	**0.514**	0.620	0.277	0.516	0.360	0.202

Table 3. Comparison results of LSS-based detectors on 3D detection on the nuScenes `test` set. TTA denotes test time augmentation strategy.

Method	Backbone	Image Size	TTA	mAP↑	NDS↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
BEVDet [17]	V2-99	900 × 1600	✔	0.424	0.488	0.524	0.242	0.373	0.950	0.148
BEVerse [53]	Swin-B	900 × 1600	✔	0.393	0.531	0.541	0.247	0.394	0.345	0.129
BEVDet4D [16]	Swin-B	900 × 1600	✔	0.451	0.569	0.511	0.241	0.386	0.301	0.121
OA-BEV [6]	V2-99	900 × 1600	✔	0.494	0.575	0.571	0.256	0.377	0.385	0.132
BEVDistill [5]	ConvNeXt-B	640 × 1600	✘	0.496	0.594	0.475	0.249	0.378	0.313	0.125
BEVDepth [25]	ConvNeXt-B	640 × 1600	✘	0.520	0.609	0.445	0.243	0.352	0.347	0.127
BEVStereo [24]	V2-99	640 × 1600	✘	0.525	0.610	0.431	0.246	0.358	0.357	0.138
AeDet [11]	ConvNeXt-B	640 × 1600	✔	0.531	0.620	0.439	0.247	0.344	0.292	0.130
TiG-BEV [18]	ConvNeXt-B	640 × 1600	✔	0.532	0.619	0.450	0.244	0.343	0.306	0.132
SA-BEV [52]	V2-99	640 × 1600	✘	0.533	0.624	0.430	0.241	0.338	0.282	0.139
FB-BEV [28]	V2-99	640 × 1600	✘	0.537	0.624	0.439	0.250	0.358	0.270	0.128
SOLOFusion [34]	ConvNeXt-B	640 × 1600	✘	0.540	0.619	0.453	0.257	0.376	0.276	0.148
LSSInst	ConvNeXt-B	640 × 1600	✘	**0.546**	**0.629**	0.464	0.251	0.341	0.265	0.120

that LSSInst can exploit fine-grained pixel-level features and better enhance perceptual capability with aspects of translation, scale, and orientation, which are all relevant in geometric-wise perception.

4.4. Noise Resistance for Practical Robustness

Although we have verified the high performance of LSSInst on nuScenes [1], even the large-scale autonomous driving dataset inevitably contain disturbances in the extrinsics obtained when sensors collect data in huge quantities. In actual autonomous driving scenarios, the detector is required to be resistant to the disturbance noise caused by small measurement errors. Therefore, here we add a set of random rotation noise with increasing proportions to the extrinsics, exploring the robustness of LSSInst under inaccurate extrinsics. Here, the baseline is BEVDepth4D [25] with 4 frames. As shown in Tab. 5, we demonstrate that LSSInst maintains good robustness, exhibiting higher performance and smaller overall attenuation.

4.5. Multiplicate Queries Ablations

To further investigate the impact of the multiplicate queries, as shown in Tab. 6, we explored two scenarios: using only proposal queries (referred to as Q_β) or learnable potential queries (referred to as Q_γ), and incorporating both queries. Here we follow the classical setting of query maximum number by default 900 in [46]. We can observe that on the one hand, relying solely on the potential queries cannot play a major role, and even utilizing all 900 queries yielded mediocre performance, which shows the slow convergence because of the initialization semantic dispersity without the scene-level information basis from BEV as aforementioned. On the other hand, though the proposal queries from BEV alone can achieve overall good results, adding more queries does not achieve a better improvement, which proves its overfitting characteristics for the scene and the fact of the neglectful detection of missing objects in the scene. However, when incorporating two kinds of queries, the performance is further improved and reaches a new level. It can be

Table 4. Generalization and Geometric-wise Results of LSSInst compared with LSS-type Baselines. (‡ please refer to Footnote 2).

Method	mAP↑	NDS↑	mATE↓	mASE↓	mAOE↓	Param (M)	Training Cost (min/epoch)	Inference Cost (sec/frame)
BEVDet [17]	0.260	0.319	0.830	0.292	0.758	55.7	14	0.04
LSSInst with BEVDet	**0.310**	**0.367**	**0.771**	**0.285**	**0.658**	64.0	18	0.05
BEVDepth4D [25]	0.343	0.458	0.691	0.281	0.610	83.5	11	0.10
LSSInst with BEVDepth4D	**0.365**	**0.477**	**0.671**	**0.275**	**0.492**	91.8	13	0.11
BEVStereo [24]	0.348	0.463	0.675	0.278	0.577	92.0	7	0.19
LSSInst with BEVStereo	**0.372**	**0.481**	**0.658**	**0.275**	**0.492**	102.3	10	0.21
SOLOFusion [34]	0.406	0.497	0.609	0.284	0.650	64.4	23	0.07
LSSInst with SOLOFusion	**0.422**	**0.514**	0.620‡	**0.277**	**0.516**	72.8	26	0.08

Table 5. The noise resistance results for robustness.

Method	**Noise**	**mAP%↑**	**Attenu.%↓**	**NDS%↑**	**Attenu.%↓**
Baseline	0	35.74	-	46.84	-
LSSInst		38.28	-	49.43	-
Baseline	0.5%	35.38	1.01	46.44	0.85
LSSInst		**38.01**	0.71	**49.19**	0.49
Baseline	1%	34.13	4.5	45.31	3.3
LSSInst		**36.58**	4.4	**47.85**	3.2
Baseline	2%	30.32	15.2	41.77	10.8
LSSInst		**32.23**	15.8	**44.28**	10.4
Baseline	3%	26.22	26.7	37.84	19.2
LSSInst		**27.92**	27.1	**40.47**	18.1

Table 6. Query Composition

Composition of Queries	mAP↑	NDS↑
450 Q_γ	0.157	0.226
900 Q_γ	0.263	0.297
450 Q_β	0.331	0.447
900 Q_β	0.330	0.446
450 Q_β + 450 Q_γ	**0.362**	**0.474**

Table 7. Segmentation mIoU

Methods	with GT	with baseline
Baseline	44.56	-
LSSInst	**46.63**	**66.21** (>50)

concluded that these two types of queries play their unique roles, and their inseparable and complementary synergy enables the model to have a comprehensive understanding from the global scene level to the local instance level.

4.6. BEV-to-Instance Semantic Coherence

To confirm the BEV-to-instance semantic coherence, we conduct the relevant experiments in two aspects. Assuming there is only one foreground class, we calculate the mIoU metric of semantic segmentation compared with the ground truth and baseline as shown in Tab. 7. According to the results with the ground truth, LSSInst is observed as having better semantic maintenance than the LSS baseline, which shows the improvement of perceptual capability to the extra BEV-insensitive objects in the scene. As for mIoU with the LSS baseline, the value 66.21% is over 50% which also indicates the promising BEV-to-instance semantic coherence. See Supplementary Material for qualitative details.

5. Conclusion

Existing LSS-based methods make efforts to build up a desirable BEV representation, but they ignore its inherent shortage of geometric loss in the formulation, suppressing its generality in 3D perception. In this paper, we propose LSSInst, a two-stage detector that improves the geometric modeling of the BEV perception with instance representation. To address the challenge of the gap between two representation spaces, we propose the instance adapter to keep the BEV-to-instance semantic coherence. Then a newly-designed instance branch is introduced to look back for fine-grained geometric matching and feature aggregation. Extensive experimental results demonstrated that our framework is of great generalization ability in modern LSS-based BEV perceptions and excellent performance, surpassing the current state-of-the-art works. We hope that our work will inspire further exploration of generalized 3D perception in more complex and fine-grained outdoor-scene tasks.

Acknowledgement

This work is partially supported by the National Key R&D Program of China (No. 2022ZD0160101) and the National Natural Science Foundation of China (No. 62206244).

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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 6, 7, 3

[2] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *CVPR*, 2018. 3

[3] Dian Chen, Jie Li, Vitor Guizilini, Rares Andrei Ambrus, and Adrien Gaidon. Viewpoint equivariance for multi-view 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9213–9222, 2023. 2, 5

[4] Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Dsgn: Deep stereo geometry network for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12536–12545, 2020. 3

[5] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Bevdistill: Cross-modal bev distillation for multi-view 3d object detection. *International Conference on Learning Representations (ICLR)*, 2022. 3, 7

[6] Xiaomeng Chu, Jiajun Deng, Yuan Zhao, Jianmin Ji, Yu Zhang, Houqiang Li, and Yanyong Zhang. Oa-bev: Bringing object awareness to bird's-eye-view representation for multi-camera 3d object detection. *arXiv preprint arXiv:2301.05711*, 2023. 3, 7

[7] MMDetection3D Contributors. Mmdetection3d: Openmmlab next-generation platform for general 3d object detection. *https://github.com/open-mmlab/mmdetection3d*, 2020. 6, 2

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009. 6

[9] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wengang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. In *AAAI*, 2021. 3

[10] Simon Doll, Richard Schulz, Lukas Schneider, Viviane Benzin, Enzweiler Markus, and Hendrik P.A. Lensch. Spatialdetr: Robust scalable transformer-based 3d object detection from multi-view camera images with global cross-sensor attention. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[11] Chengjian Feng, Zequn Jie, Yujie Zhong, Xiangxiang Chu, and Lin Ma. Aedet: Azimuth-invariant multi-view 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21580–21588, 2023. 1, 3, 7, 2

[12] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 3

[13] Chunrui Han, Jianjian Sun, Zheng Ge, Jinrong Yang, Runpei Dong, Hongyu Zhou, Weixin Mao, Yuang Peng, and Xiangyu Zhang. Exploring recurrent long-term temporal fusion for multi-view 3d perception, 2023. 1

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 6

[15] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units. *ArXiv*, abs/1606.08415, 2016. 1

[16] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022. 1, 7, 2

[17] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 1, 2, 3, 7, 8

[18] Peixiang Huang, Li Liu, Renrui Zhang, Song Zhang, Xinli Xu, Baichao Wang, and Guoyi Liu. Tig-bev: Multi-view bev 3d object detection via target inner-geometry learning. *arXiv preprint arXiv:2212.13979*, 2022. 7

[19] Hongxiang Jiang, Wenming Meng, Hongmei Zhu, Qian Zhang, and Jihao Yin. Multi-camera calibration free bev representation for 3d object detection. *arXiv preprint arXiv:2210.17252*, 2022. 2

[20] Yanqin Jiang, Li Zhang, Zhenwei Miao, Xiatian Zhu, Jin Gao, Weiming Hu, and Yu-Gang Jiang. Polarformer: Multi-camera 3d object detection with polar transformers. In *AAAI*, 2023. 5

[21] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019. 1

[22] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M. Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. *CVPR*, 2022. 2

[23] Shichao Li, Zechun Liu, Zhiqiang Shen, and Kwang-Ting Cheng. Stereo neural vernier caliper. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1376–1385, 2022. 3

[24] Yinhao Li, Han Bao, Zheng Ge, Jinrong Yang, Jianjian Sun, and Zeming Li. Bevstereo: Enhancing depth estimation in multi-view 3d object detection with dynamic temporal stereo. In *AAAI*, 2023. 1, 3, 7, 8, 2

[25] 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 *AAAI*, 2023. 1, 3, 7, 8, 2

[26] Zhichao Li, Feng Wang, and Naiyan Wang. Lidar r-cnn: An efficient and universal 3d object detector. In *CVPR*, 2021. 3

[27] 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 *Eur. Conf. Comput. Vis.*, 2022. 1, 3

[28] Zhiqi Li, Zhiding Yu, Wenhai Wang, Anima Anandkumar, Tong Lu, and José Manuel Álvarez. Fb-bev: Bev representation from forward-backward view transformations. *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 3, 7

[29] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2016. 2

[30] Lijie Liu, Jiwen Lu, Chunjing Xu, Qi Tian, and Jie Zhou. Deep fitting degree scoring network for monocular 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1057–1066, 2019. 3

[31] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. PETR: position embedding transformation for multi-view 3d object detection. In *Eur. Conf. Comput. Vis.*, 2022. 2, 3

[32] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[33] Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3142–3152, 2021. 3

[34] Jinhyung Park, Chenfeng Xu, Shijia Yang, Kurt Keutzer, Kris Kitani, Masayoshi Tomizuka, and Wei Zhan. Time will tell: New outlooks and A baseline for temporal multi-view 3d object detection. In *Int. Conf. Learn. Represent.*, 2023. 1, 3, 4, 7, 8, 2

[35] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Eur. Conf. Comput. Vis.*, 2020. 1, 2

[36] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017. 3

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *TPMAI*, 2016. 3

[38] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, 2019. 3

[39] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *TPAMI*, 2020. 3

[40] Andrea Simonelli, Samuel Rota Bulò, Lorenzo Porzi, Manuel López-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection: From single to multi-class recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(3):1219–1231, 2022. 3

[41] Pei Sun, Weiyue Wang, Yuning Chai, Gamaleldin Elsayed, Alex Bewley, Xiao Zhang, Cristian Sminchisescu, and Dragomir Anguelov. Rsn: Range sparse net for efficient, accurate lidar 3d object detection. In *CVPR*, 2021. 3

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 5, 6

[43] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4604–4612, 2020. 1

[44] Shihao Wang, Xiaohui Jiang, and Ying Li. Focal-petr: Embracing foreground for efficient multi-camera 3d object detection. *IEEE Transactions on Intelligent Vehicles*, pages 1–10, 2023. 3

[45] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. FCOS3D: fully convolutional one-stage monocular 3d object detection. In *Int. Conf. Comput. Vis. Worksh.*, 2021. 1

[46] 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 (CoRL)*, 2021. 1, 2, 3, 5, 7

[47] Zengran Wang, Chen Min, Zheng Ge, Yinhao Li, Zeming Li, Hongyu Yang, and Di Huang. STS: surround-view temporal stereo for multi-view 3d detection. In *AAAI*, 2023. 2

[48] Yan Yan, Yuxing Mao, and Bo Li. SECOND: sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 1, 2

[49] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023. 1, 3, 7

[50] Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Std: Sparse-to-dense 3d object detector for point cloud. In *ICCV*, 2019. 3

[51] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking. *CVPR*, 2021. 3, 2

[52] Jinqing Zhang, Yanan Zhang, Qingjie Liu, and Yunhong Wang. Sa-bev: Generating semantic-aware bird's-eye-view feature for multi-view 3d object detection. *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 3, 7, 2

[53] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *arXiv preprint arXiv:2205.09743*, 2022. 7

[54] Yunpeng Zhang, Wenzhao Zheng, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. A simple baseline for multi-camera 3d object detection. In *AAAI*, 2023. 3, 7

[55] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018. 1

[56] 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. 5