

One Training for Multiple Deployments: Polar-based Adaptive BEV Perception for Autonomous Driving

Huitong Yang¹, Xuyang Bai², Xinge Zhu^{3*}, and Yuexin Ma^{1*}

Abstract—Current on-board chips usually have different computing power, which means multiple training processes are needed for adapting the same learning-based algorithm to different chips, costing huge computing resources. The situation becomes even worse for 3D perception methods with large models. Previous vision-centric 3D perception approaches are trained with regular grid-represented feature maps of fixed resolutions, which is not applicable to adapt to other grid scales, limiting wider deployment. In this paper, we leverage the Polar representation when constructing the BEV feature map from images in order to achieve the goal of training once for multiple deployments. Specifically, the feature along rays in Polar space can be easily adaptively sampled and projected to the feature in Cartesian space with arbitrary resolutions. To further improve the adaptation capability, we make multi-scale contextual information interact with each other to enhance the feature representation. Experiments on a large-scale autonomous driving dataset show that our method outperforms others as for the good property of one training for multiple deployments.

I. INTRODUCTION

Accurate 3D perception is critical for autonomous vehicles to understand the complexities surrounding scenes and make safe and effective driving decisions. There are tremendous research works contributing to this task, among which the vision-centric methods become more and more popular due to the low computation cost. Although such learning-based approaches have achieved promising performance, they are usually trained with a predefined feature map resolution, which can only be integrated into limited on-board chips with corresponding computing resources and are not applicable for multiple deployments under arbitrary computational budgets. Retraining a model with different resolutions will cost extra huge computing resources. Moreover, perception is not the only task conducted on the chip, other tasks also occupy computing resources. Dynamic resource allocation can improve the service efficiency of the chip, but it demands that the perception algorithm possesses dynamic adjustment capability as well. Thus, a vision-centric method that can adaptively adjust the computation requirements and be deployed to chips with various computing performances becomes extremely significant.

* Corresponding author

¹Huitong Yang and Yuexin Ma is with School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China. mayuexin@shanghaitech.edu.cn

²Xuyang Bai is with Department of Computer Science and Engineering, Hong Kong University of Science and Technology. xbaiad@connect.ust.hk

³Xinge Zhu is with Department of Information Engineering, the Chinese University of Hong Kong. zx018@ie.cuhk.edu.hk

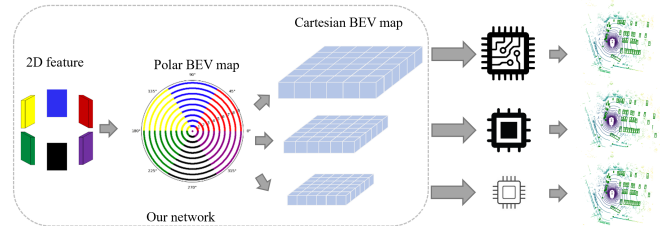


Fig. 1: Benefiting from the grid-insensitive sampling nature of the Polar-based feature map representation, our network can supply various Cartesian BEV grid resolutions for 3D perception according to the actual computing budget during deploying.

Current mainstream vision-centric 3D perception methods usually project the perspective-view (PV) image feature map to the bird's eye view (BEV) feature map and then attach detection or segmentation head for specific 3D perception tasks. However, such methods construct regular-grid represented BEV feature maps in the Cartesian coordinate system and assign fixed physical scale correspondences for the grid during training, making them hard to be adapted to other resolutions. Directly downsampling or upsampling the BEV feature map to satisfy various computing requirements will result in dramatic performance drop for 3D perception. Actually, compared with regular grid representation, Polar representation on BEV is more reasonable since each Polar ray can physically correspond to one column of the image pixels [34, 19] according to foreshortening effect of camera imaging. More importantly, the feature along each ray is grid-insensitive, and can be flexibly projected to multiple Cartesian feature maps with different resolutions, as shown in Figure. 1. That is to say, even though the network is trained under the condition that the Polar feature is transferred to 128×128 Cartesian feature map, the pre-trained Polar feature can easily adapt to new Cartesian feature maps with the resolution of 256×256 or 64×64 to fit diverse computing budget allocations. To our knowledge, we are the first to exploit this property of Polar-based BEV representation to achieve the goal of one training for multiple deployments.

In this paper, based on Polar BEV feature representation, we propose a novel vision-centric 3D detection method for autonomous driving, which can be flexibly deployed multiple times to adapt to diverse computing budgets with only one training procedure. The whole pipeline contains three main modules, as Figure. 2 shows, including the column-wise PV-to-BEV transformer module, the grid-insensitive Polar-to-Cartesian sampling module, and the multi-scale feature

interaction module. The first module projects the PV image feature to BEV Polar feature by multi-head attention and depth distribution estimation. For the second module, the feature along rays in the Polar coordinate system can be adaptively sampled and projected to the feature with different grid sizes in Cartesian space. Therefore, when the computing budget becomes limited, our network can naturally transfer the Polar feature to small Cartesian feature maps to reduce the occupancy of computing resources and keep high-quality performance at the same time. To further improve the adaptation capability, we design the multi-scale feature interaction module to enhance the feature representation through transformer mechanism. At the end, we attach one detection head for the 3D object localization. We evaluate our method on a large-scale autonomous-driving dataset, nuScenes [3], and achieve the state-of-the-art performance under the condition of one training for multiple deployments. Ablation study is also conducted to verify the effectiveness of important modules of our network.

In summary, our contributions are as follows,

- We propose a novel Polar-based BEV perception method, which can be adapted to various computing budgets for multiple deployments once trained.
- We make use of the information interaction among multi-scale BEV features to enhance the feature representation for better adaptation.
- Our method achieves state-of-the-art generalization capability of inferencing on feature maps with unseen resolutions for 3D detection on large-scale autonomous driving dataset.

II. RELATED WORK

With the popularization of low-cost autonomous driving technology, vision-centric BEV perception algorithms have gradually attracted attention. Our work focuses on this promising technique and further proposes an adaptive deployment solution, which is urgent and significant for real applications. We introduce the related work from two aspects, including Cartesian-based BEV perception methods and polar-based ones.

A. Cartesian-based BEV Perception

Current BEV perception works [23] can mainly be divided into homograph-based, depth-based, MLP-based methods, and transformer-based four kinds of methods. Homograph-based methods [1, 46, 12, 4, 6, 20, 35, 29] exploit the Inverse Perspective Mapping (IPM) [24] to establish the geometric transformation between PV and BEV view so that many downstream vision tasks can be explored, such as 3D object detection, semantic map construction, and motion planning. In addition, depth-based methods [31, 9, 42, 28, 7, 2] estimate depth distribution explicitly or implicitly and use it as the bridge to generate BEV feature from multi-view image feature. LSS [27, 9] builds the camera frustum by pixel-level depth distribution and projects it to the world space. To obtain the more accurate depth estimation, BEVDepth [14] leverages the depth value of sparse Lidar points to provide

explicit depth supervision, which significantly improves the accuracy of 3D detection. Moreover, some methods [8, 40, 49, 25, 13, 30, 33] utilize MLP as a mapping function to transform views. For transformer-based methods [17, 18, 32, 26, 16], they usually design a set of BEV queries using positional encoding, then performs the view transformation through cross attention between BEV queries and the image features. DETR3D [38] first applies sparse 3D learn-able reference points as no-explicit BEV queries to link the 2D feature from multi-view cameras. BEVFormer [15] exploits the transformer to unify the spatial and temporal information to learn the dense BEV feature without depth estimation and enable multi-task learning. Although above algorithms exhibit competitive performance, their feature maps are all designed in Cartesian coordinate system, where regular grid division is defined with fixed scale correspondence to the physical world. If we use such networks trained under one scale to infer other scales of output, the performance will dramatically drop.

B. Polar-based BEV Perception

Polar coordinate system is another choice to represent the 3D space. It first appears in LiDAR-based 3D perception [48, 5] approaches. PolarNet [39] projects the point cloud to BEV representation in Polar coordinate, and controls the points with learnable grid cells, bringing significant effects to the long-tailed problem of spatial point cloud distribution. Cylinder3D [47] proposes dynamical 3D voxelization in the cylinder coordinates to adapt the varying-density property of outdoor Lidar points. Actually, polar representation is also applicable in the camera-based BEV perception task due to the foreshortening effect of camera imaging. Moreover, the Polar-based BEV representation [11] is more consistent to represent the whole scenes captured by the ego vehicle’s surrounding cameras. Following the ray tracing principle, Ego3RT [21] treats the dense polar BEV grids as human eyes and uses camera parameters to construct a 3D-to-2D reverse projection for multi-task learning. PolarBEV [19] establishes an uneven BEV grid representation, considering the different object scales according to the distance to the cameras by the angle-specific and radius-specific embedding. Our method also projects the image columns to polar rays with the transformer. However, we are the first to focus on the BEV grid insensitive sampling nature of the polar map and leverage the property to achieve multi-deployment adaptation according to the real-time computational budget and inference speed in actual applications.

III. METHOD

We propose a novel vision-centric Polar-based 3D object detector, which is trained once and can be deployed with various granularity of BEV feature maps based on the computation budget. As shown in Fig 2, our framework takes multi-view RGB images as input and produces 3D bounding boxes and their category as output. Following [9], the surround-view images are first encoded by a single shared backbone to obtain multi-scale 2D representations $\{F_i\}_{i=1}^{scale}$

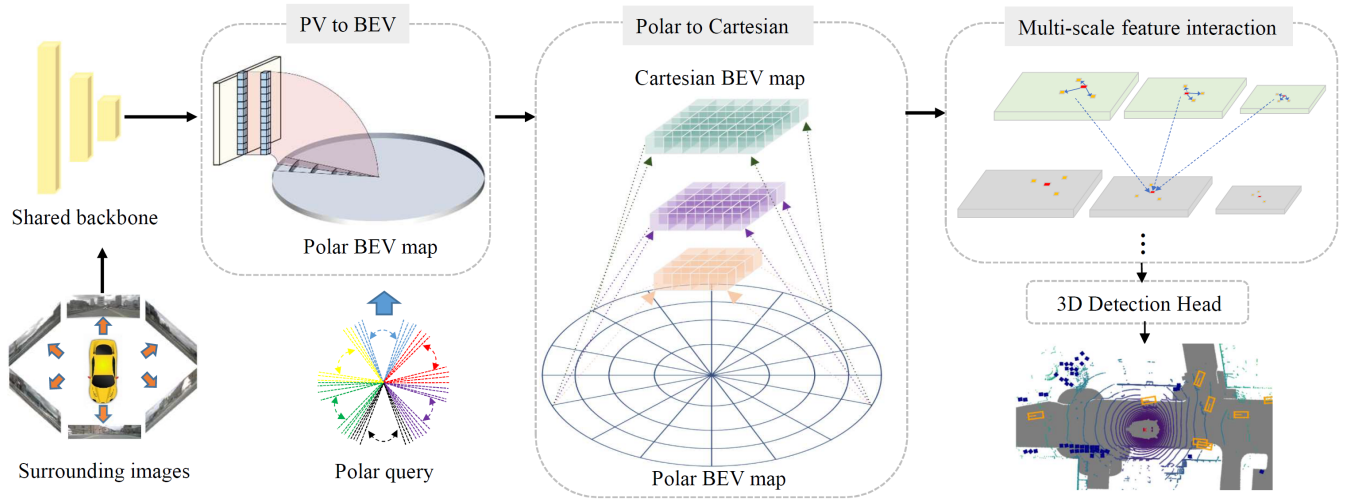


Fig. 2: Overall pipeline of our method. We exploit standard 2D backbone to extract multi-view features. Then, a Column-wise PV-to-BEV Transformer module projects the columns from image feature map to corresponding Polar rays using a cross-attention mechanism. Next, we propose a grid-insensitive Polar-to-Cartesian sampling strategy to transform the Polar BEV map to various granularities of the Cartesian BEV grid, which can be adjusted during deployment according to the computation budgets in the actual autonomous scenes. We additionally propose a multi-scale BEV feature interaction encoder to correlate contextual information across multi-scale Cartesian BEV map. Finally, a detection head is attached to predict objects’ 3D bounding boxes and categories.

with size of $\frac{H}{2^{i+1}} \times \frac{W}{2^{i+1}} \times C$ and then unified into a single resolution 2D feature map using FPNLSS. To conduct 3D perception tasks from images, the mainstream methods usually start with projecting the PV feature map onto BEV space, which can be directly deployed by many downstream real-world applications such as behavior prediction, motion planning, etc. However, since they usually adopt a regular grid-based Cartesian BEV representation with a predefined grid size during training, it is difficult to adapt to different scales during inference, limiting their applications in self-driving scenarios that are having different computational budgets from the training process. To mitigate this issue and design a detector that can be flexibly deployed under different computational budgets, we adopt the Polar coordinate representation, which represents the physical world more naturally according to the image-forming principle. More importantly, the feature along each ray is grid-insensitive, and can be utilized to project into multiple Cartesian BEV feature maps with different resolutions. In this way, our method can avoid multiple times of training for deployments under different computational budgets, saving lots of time and resources, which is essential to applications on limited on-board chips. To achieve this goal, we propose three novel modules, including the column-wise PV-to-BEV transformer module, the grid-insensitive Polar-to-Cartesian sampling module, and the multi-scale feature interaction module, which will be illustrated in the following.

A. Column-wise PV-to-BEV Transformer Module

One key step in vision-centric perception algorithms is to perform view transformation from PV to BEV, as BEV is a natural representation of the world and is fusion-friendly. The mainstream methods can be divided into two categories:

geometry-based transformation and network-based transformation [23]. The former fully utilizes the physical principles of the camera to transfer the view in an interpretable manner, but might suffer from inaccurate depth prediction. The latter performs the view transformation in a data-driven way without explicitly leveraging the camera projection model. In our work, we build a network-based view transformation module by extending the single-image view transformer [34] to multiview scenario. As shown in Fig. 2, we first assume the 1-1 relationship between vertical scanlines in the image and the rays on the BEV plane starting from the camera center. Then the view transformation can be formulated as a set of 1D sequence-to-sequence translation problems and modeled by a transformer module. Such column-wise transformer module avoids the dense attention between 2D image feature maps and BEV queries, and instead only relies on 1D sequence-to-sequence translation, leading to a memory-friendly and data-efficient architecture.

Specifically, we first design a set of positional embedding (called Polar query) initialized on the polarized BEV plane. Next, based on the mapping relationship between each image column and Polar ray established by the camera geometry, we translate the image features from the image column to its associated ray by performing cross attention between image features on the column and the Polar queries on the ray. This process can be viewed as a learnable assignment of semantic PV features to their positional slots along a Polar ray in the BEV plane. Different from the single image case in [34], multiple surround images usually overlap with each other, where the multiview observations of the overlap region is critical for many detection targets, such as the large objects being truncated in a single image.

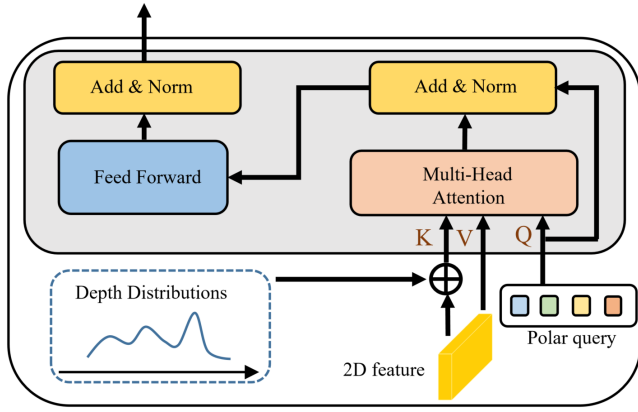


Fig. 3: Detailed network of the column-wise PV-to-BEV transformer module.

Thus to enhance the information on overlap regions, we count the number of columns projected to the same ray and adaptively assign higher attention weights, so that our column-wise cross-attention module can better handle the information from overlapping regions.

To further facilitate the geometric reasoning ability of transformer module, we inject the depth information into the view transformation process, as shown in Fig 3. We first predict the depth distribution of each image pixel following [14] and then use an MLP network to embed the depth distribution into positional embedding, which will be added with the 2D features to provide the depth guidance for PV-to-BEV transformation. Although our column-wise PV-to-BEV transformation module does not necessarily need per-pixel depth for view transformation, we still empirically find the depth information to be important for geometric reasoning of transformers.

B. Grid-insensitive Polar-to-Cartesian Sampling Module

To enable the deploy-time adjustment of perception granularity based on the actual computation budget, we propose a grid-insensitive sampling module so that our network can generalize to different BEV feature resolutions. Specifically, we first generate a Cartesian meshgrid $M \in R^{H \times W \times 2} = (w_i, h_i)$ within the multi-view perception range $[-51.2m, 51.2m]$ with corresponding resolutions, where $H \times W$ is the spatial resolution of BEV feature map. The meshgrid will be leveraged during Polar-to-Cartesian feature transformation by sampling the Polar map to get the Cartesian BEV feature map. The conversion between the Polar coordinate system and Cartesian coordinate system is calculated as follow,

$$\phi^r = \arctan\left(\frac{w_i}{h_i}\right), \sigma^r = \sqrt{(w_i^2 + h_i^2)}, \quad (1)$$

where (ϕ^r, σ^r) is the coordinate of Polar map. To cover the perception range of $[-51.2m, 51.2m]$, we set the range of Polar coordinate system to $\sigma^r \in [0m, 72m]$, $\phi^r \in [0, 2\pi]$ and normalize the Polar-to-Cartesian transformation. Finally, we stack the normalized Polar coordinate to get the Polar-to-Cartesian BEV meshgrid:

$$\begin{aligned} \phi^r &= (\phi^r - P_{\min}^r) / (P_{\max}^r - P_{\min}^r), \\ \sigma^r &= (\sigma^r - \sigma_{\min}^r) / (\sigma_{\max}^r - \sigma_{\min}^r), \\ P &= \text{Stack}([\phi^r, \sigma^r]) \in R^{H \times W \times 2}. \end{aligned} \quad (2)$$

Then our grid-insensitive sampler can transform the Polar BEV map to Cartesian BEV grid with different resolutions through the bi-linear interpolation operation. The multi-resolution BEV feature map derived from the grid-insensitive sampler enables the potential of adapting our single network to various granularities of feature maps during inference to facilitate the deployment of different computational budgets.

C. Multi-scale Feature Interaction Module

To capture multi-scale contextual information and improve the representation power of multiple BEV feature maps with different resolutions, we propose a Multi-scale BEV Interaction Encoder (MBIE) to extract different granularity of contextual information from sampled Cartesian BEV maps. Following [11], we build the MBIE module by stacking multiple deformable attention layers to fuse the information from different scales. Formally, the deformable attention operator is defined as:

$$\begin{aligned} \text{MSBEVDeformAttn}(b_q, \hat{r}_q, \{b^s\}_{s=1}^S) = \\ \sum_h^H W_h \left[\sum_{s=1}^S \sum_{r=1}^R A_{hsqr} W_h' b^s (\sigma_s(\hat{r}_q) + \Delta f_{hsqr}) \right], \end{aligned} \quad (3)$$

where q is a query element in the BEV query feature $b_q, \hat{r}_q \in [0, 1]^2$ represents the normalized coordinates of reference points, b^s is the multi-scale BEV features, $\sigma_s(\ast)$ is the rescaling operator that converting r_q to corresponding scale and Δf_{hsqr} is the offset that helps the query feature interacting with the neighbour contextual BEV information from all levels. MBIE performs information exchange among neighboring pixels and across multiple Cartesian BEV feature maps, which enhances the multi-scale context for each feature map, and serves as a crucial step for the generalization to different BEV feature resolutions.

Followed by the MBIE module, a deformable convolution based FPN module [10] is leveraged to fuse the BEV feature map with different resolutions into the target resolution. Since we can manually control the down-sampling or up-sampling factors in FPN, it is convenient to adjust the resolution of output feature during inference. More details about the inference strategy will be illustrated in Sec. IV-C.

Given the dense BEV feature map generated from the previous stages, we adopt the first stage of CenterPoint as our detection head due to its simplicity and popularity. Following [41], we adopt the heatmap loss for classification supervision and the L1 regression loss for bounding box supervision.

IV. EXPERIMENT

In this section, we first introduce the dataset, evaluation metrics, and implementation details. Then, we give detailed

TABLE I: In this table, we show the result on different output resolutions with fixed training resolution, which also meets our motivation, *i.e.*, one training for multiple deployments. - means the value is not available for the method.

Method	BEV Grid Size	Output Resolution	Training Head	NDS \uparrow	mAP \uparrow	mATE \downarrow	mASE \downarrow	mAOE \downarrow	mAVE \downarrow	mAAE \downarrow
BEVDet	0.4 Meter	128 \times 128	128 \times 128	0.392	0.312	0.691	0.272	0.523	0.909	0.247
	0.256 Meter	200 \times 200		0.324	0.244	0.888	0.308	0.606	0.958	0.211
	0.2 Meter	256 \times 256		0.275	0.201	1.004	0.328	0.702	1.010	0.221
BEVerse	0.4 Meter	128 \times 128	128 \times 128	-	0.321	0.682	0.278	0.464	0.327	0.189
	0.256 Meter	200 \times 200		-	0.275	0.804	0.318	0.543	0.517	0.195
	0.2 Meter	256 \times 256		-	0.230	0.898	0.337	0.615	0.662	0.194
Ours	0.4 Meter	128 \times 128	128 \times 128	0.398	0.321	0.669	0.275	0.494	0.956	0.231
	0.256 Meter	200 \times 200		0.378	0.305	0.710	0.280	0.519	1.024	0.241
	0.2 Meter	256 \times 256		0.346	0.278	0.789	0.300	0.568	0.991	0.252

TABLE II: Results of more challenging setting. It can be found that these output resolutions are totally unseen and flexible.

Method	BEV Grid Size	Output Resolution	Training Head	NDS \uparrow	mAP \uparrow	mATE \downarrow	mASE \downarrow	mAOE \downarrow	mAVE \downarrow	mAAE \downarrow
BEVDet	0.8 Meter	64 \times 64	128 \times 128	0.181	0.092	1.078	0.505	0.781	0.877	0.486
	0.53 Meter	96 \times 96		0.336	0.231	0.860	0.301	0.575	0.842	0.217
	0.4 Meter	128 \times 128		0.392	0.312	0.691	0.272	0.523	0.909	0.247
BEVerse	0.8 Meter	64 \times 64	128 \times 128	-	0.107	1.054	0.365	0.641	0.510	0.251
	0.53 Meter	96 \times 96		-	0.254	0.778	0.296	0.509	0.385	0.187
	0.4 Meter	128 \times 128		-	0.321	0.682	0.278	0.464	0.327	0.189
Ours	0.8 Meter	64 \times 64	128 \times 128	0.290	0.143	1.057	0.288	0.589	0.713	0.217
	0.53 Meter	96 \times 96		0.370	0.280	0.749	0.279	0.504	0.951	0.223
	0.4 Meter	128 \times 128		0.398	0.321	0.669	0.275	0.494	0.956	0.231

comparison of generalization capability with different baseline methods. Furthermore, we evaluate our method with various methods on nuScenes dataset to show its effectiveness and extensive ablation studies are conducted to validate each proposed component.

A. Datasets & Metrics

We evaluate our approach on a large-scale dataset, nuScenes [3], which contains 1000 scenes with durations of 20 seconds, and each frame contains 6 surrounding cameras with the resolution of 1600 \times 900. For 3D object detection, we mainly evaluate the performance by the following metrics: mean Average Precision (mAP), Average Translation Error (ATE), Average Scale Error (ASE), Average Orientation Error (AOE), Average Velocity Error (AVE), Average Attribute Error (AAE), and NuScenes Detection Score (NDS) which is computed as $(NDS = \frac{1}{10}[5 * mAP + \sum_{mTP \in TP} (1 - \min(1, mTP))])$.

TABLE III: The ablations on our proposed modules. CPBT represents the Column-wise PV-to-BEV Transformer Module. MBIE is the Multi-scale Feature Interaction Module.

CPBT	MBIE	NDS \uparrow	mAP \uparrow
		0.384	0.294
✓		0.391	0.307
✓	✓	0.398	0.321

B. Implementation Details

For a fair comparison, We employ the pretrained Swin-Transformer as our 2D backbone network, following existing works [14, 15, 42]. Our perception range is $[-51.2m, 51.2m]$ along the X and Y axis. For the view symmetry of multi-view cameras, the radius and azimuth are set to (144, 360), which represents the polar perception range. The input multi-view image is rescaled to (256, 704) for prototype verification. The network is trained with CBGS [45] on 4 RTX 3090 GPUs with total batch size of 8. During the inference, we apply

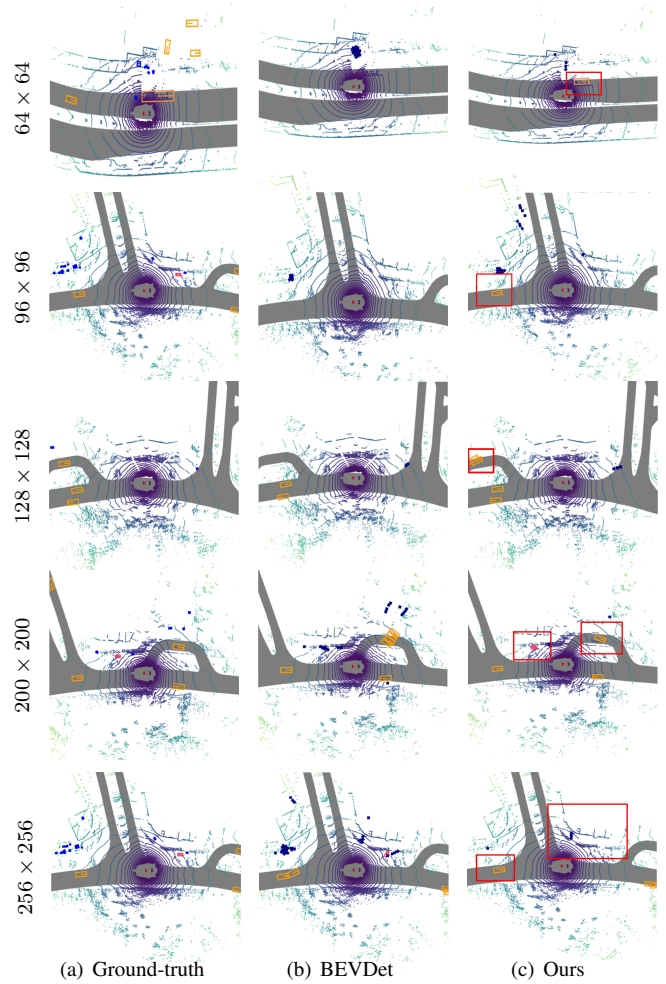


Fig. 4: Visualization of the experimental results in Table. I II. We highlight the area with more accurate predictions for our method with blue rectangles.

TABLE IV: Quantitative evaluation of recent vision-centric 3D detection works on the nuScenes val set. Among all methods, DETR3D, PETR, BEVDet and our method are trained with CBGS [45].

Method	Backbone	Size	NDS \uparrow	mAP \uparrow	mATE \downarrow	mASE \downarrow	mAOE \downarrow	mAVE \downarrow	mAAE \downarrow
CenterNet [44]	DLA	–	0.328	0.306	0.716	0.264	0.609	1.426	0.658
FCOS3D [36]	ResNet-50	1600 \times 900	0.368	0.288	0.777	0.266	0.544	1.228	0.170
DETR3D [38]	ResNet-50	1600 \times 900	0.373	0.302	0.811	0.282	0.493	0.979	0.212
PGD [37]	ResNet-50	1600 \times 900	0.394	0.320	0.735	0.266	0.492	1.114	0.170
Ego3RT [22]	ResNet-50	1600 \times 900	0.380	0.332	0.706	0.281	0.663	0.964	0.249
PETR [17]	ResNet-50	1056 \times 384	0.381	0.313	0.768	0.278	0.564	0.923	0.225
BEVDet [9]	ResNet-50	704 \times 256	0.379	0.298	0.725	0.279	0.589	0.860	0.245
BEVDet [9]	ResNet-101	704 \times 256	0.381	0.302	0.722	0.269	0.543	0.900	0.269
BEVDet [9]	Swin-tiny	704 \times 256	0.392	0.312	0.691	0.272	0.523	0.909	0.247
PersDet [43]	ResNet-50	704 \times 256	0.389	0.319	0.676	0.284	0.589	0.924	0.229
CaDDN [28]	ResNet-50	704 \times 256	0.370	0.294	0.702	0.283	0.579	0.988	0.222
Ours	Swin-tiny	704 \times 256	0.398	0.321	0.669	0.275	0.494	0.956	0.231

our model to different BEV grid resolutions with the fixed perception range, where we sample the polar feature maps to rectangle Cartesian BEV feature maps with various grid resolutions for adapting to different computational budgets. For the inference speed, our method is 9 FPS for 64×64 BEV resolution and 6.5 FPS for 256×256 BEV resolution.

C. One Training for Multiple Deployments

In this section, we show the results of our method in the one-training-multiple-deployments setting. Specifically, the proposed method is trained with only one head (128×128 feature map) while it would conduct the inference on multiple feature maps with different size, like 200×200 and 256×256 , which also denotes the different computational budgets. Due to the resolution-invariant property of Polar feature space, our method can adapt to different BEV grid sizes by various degrees of sampling. While for these existing methods, including BEVDet [9] and BEVerse [42], they cannot be employed to other resolution directly. To make a comparison, a modification of these methods is applied where we perform the interpolation over the BEV feature map and keep the detection head untouched to adapt them to different output sizes. For some other related BEV perception works [38, 17, 18], they do not build the explicit BEV feature map and lack the ability for diverse deployments with various computing budgets, so we do not compare with them here.

As shown in Table I, all methods are trained with feature map size of 128×128 and different feature maps are used for inference (200×200 and 256×256). Note that different sizes of output features indicate the different resolution of BEV grid, also indicate the different computational budgets. It can be found that our method achieves the consistent performance gain compared to these baseline methods in almost all metrics. Specifically, when we perform the inference on different output feature maps (200×200 and 256×256), our method shows decent generalization capability with marginal performance drop, like -2% ($128 \times 128 \rightarrow 200 \times 200$) and -5% ($128 \times 128 \rightarrow 256 \times 256$) in terms of NDS, while BEVDet [9] gives a large degradation, -7% and -12%. Note that BEVerse performs better in mAVE, mAAE and NDS because the multi-frame images are leveraged as input. In our work, we focus more on the generalization ability among different output resolutions, which is clearly demonstrated from the marginal mAP drops when inferencing with an output different resolution.

We also provide visualization results in Figure. 4. From the highlighted areas, it is obvious to see that our method has more accurate detection ability, especially when adapting to other feature map scales. It further demonstrates the superiority of our method.

More challenging deployments In Table II, we conduct the inference on a more challenging and flexible setting, where 64×64 and 96×96 output feature maps are conducted and the training head is also 128×128 . It is noted that these output feature maps are flexible and unseen during training. From the results, we can find that our method achieves the best generalization performance with the least performance drop compared to baseline methods, indicating its good generalization capability in the challenging setting.

D. Ablation Study and Qualitative Evaluation

To verify the effectiveness of important modules of our network, we conduct ablation study and the results are shown in Table. III. We can see that the polar-based BEV feature representation is superior to pure Cartesian feature representation and the multi-scale feature interaction process further improve the performance. We also show the quantitative evaluation results for vision-centric BEV perception in Table. IV. Compared with other methods, our method is decent and competitive.

V. CONCLUSION

In this paper, we propose a novel vision-centric BEV perception approach for autonomous driving, which can be flexibly deployed to multiple platforms with various computing budgets. In particular, we leverage the Polar-represented BEV feature map to solve this problem based on its good property that the feature along rays can be easily mapped to Cartesian grid-represented feature with arbitrary resolutions. We also design an effective interaction module to further enhance the feature representation by multi-scale feature interactions. Extensive experiments show that our method has superior generalization capability for inferring novel feature map scales with only one training, which is significant for the real deployment in diverse scenarios.

VI. ACKNOWLEDGEMENTS

This work was supported by NSFC (No.62206173), Shanghai Sailing Program (No.22YF1428700), and Shanghai Frontiers Science Center of Human-centered Artificial Intelligence (ShangHAI).

REFERENCES

- [1] Syed Ammar Abbas and Andrew Zisserman. “A Geometric Approach to Obtain a Bird’s Eye View From an Image”. In: *ICCV*. IEEE, 2019, pp. 4095–4104.
- [2] Adil Kaan Akan and Fatma Güney. “StretchBEV: Stretching Future Instance Prediction Spatially and Temporally”. In: *ArXiv* (2022).
- [3] Holger Caesar et al. “nusenes: A multimodal dataset for autonomous driving”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11621–11631.
- [4] Yigit Baran Can et al. “Understanding bird’s-eye view of road semantics using an onboard camera”. In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 3302–3309.
- [5] Peishan Cong et al. “STCCrowd: A Multimodal Dataset for Pedestrian Perception in Crowded Scenes”. In: *CVPR*. June 2022, pp. 19608–19617.
- [6] Noa Garnett et al. “3d-lanenet: end-to-end 3d multiple lane detection”. In: *ICCV*. 2019, pp. 2921–2930.
- [7] Nikhil Gosala and Abhinav Valada. “Bird’s-eye-view panoptic segmentation using monocular frontal view images”. In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 1968–1975.
- [8] Noureldin Hendy et al. “FISHING Net: Future Inference of Semantic Heatmaps In Grids”. In: *ArXiv abs/2006.09917* (2020).
- [9] Junjie Huang et al. “BEVDet: High-performance Multi-camera 3D Object Detection in Bird-Eye-View”. In: *CoRR abs/2112.11790* (2021).
- [10] Shihua Huang et al. “FaPN: Feature-aligned Pyramid Network for Dense Image Prediction”. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (2021), pp. 844–853.
- [11] Yanqin Jiang et al. “PolarFormer: Multi-camera 3D Object Detection with Polar Transformers”. In: *arXiv preprint arXiv:2206.15398* (2022).
- [12] Youngseok Kim and Dongsuk Kum. “Deep Learning based Vehicle Position and Orientation Estimation via Inverse Perspective Mapping Image”. In: *2019 IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2019, pp. 317–323.
- [13] Qi Li et al. “Hdmapnet: An online hd map construction and evaluation framework”. In: *ArXiv* (2021).
- [14] Yin hao Li et al. “BEVDepth: Acquisition of Reliable Depth for Multi-view 3D Object Detection”. In: *CoRR* (2022).
- [15] Zhiqi Li et al. “BEVFormer: Learning Bird’s-Eye-View Representation from Multi-Camera Images via Spatiotemporal Transformers”. In: *ArXiv* (2022).
- [16] Zhiqi Li et al. “Panoptic SegFormer: Delving deeper into panoptic segmentation with transformers”. In: *CVPR*. 2022, pp. 1280–1289.
- [17] Yingfei Liu et al. “Petr: Position embedding transformation for multi-view 3d object detection”. In: *ArXiv* (2022).
- [18] Yingfei Liu et al. “PETRv2: A Unified Framework for 3D Perception from Multi-Camera Images”. In: *ArXiv* (2022).
- [19] Zhi Liu et al. *Vision-based Uneven BEV Representation Learning with Polar Rasterization and Surface Estimation*. 2022.
- [20] Abdelhak Loukkal et al. “Driving among Flatmobiles: Bird-Eye-View occupancy grids from a monocular camera for holistic trajectory planning”. In: *WACV*. 2021, pp. 51–60.
- [21] Jiachen Lu et al. “Learning Ego 3D Representation as Ray Tracing”. In: *arXiv preprint arXiv:2206.04042* (2022).
- [22] Jiachen Lu et al. “Learning Ego 3D Representation as Ray Tracing”. In: *ArXiv* (2022).
- [23] Yuexin Ma et al. “Vision-Centric BEV Perception: A Survey”. In: *arXiv preprint arXiv:2208.02797* (2022).
- [24] Hanspeter A Mallot et al. “Inverse perspective mapping simplifies optical flow computation and obstacle detection”. In: *Biological cybernetics* (1991), pp. 177–185.
- [25] Bowen Pan et al. “Cross-view semantic segmentation for sensing surroundings”. In: *IEEE Robotics and Automation Letters* 5.3 (2020), pp. 4867–4873.
- [26] Lang Peng et al. “BEVSegFormer: Bird’s Eye View Semantic Segmentation From Arbitrary Camera Rigs”. In: *ArXiv* (2022).
- [27] Jonah Philion and Sanja Fidler. “Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D”. In: *ECCV*. Ed. by Andrea Vedaldi et al. Springer, 2020, pp. 194–210.
- [28] Cody Reading et al. “Categorical depth distribution network for monocular 3d object detection”. In: *CVPR*. 2021, pp. 8555–8564.
- [29] Lennart Reiher, Bastian Lampe, and Lutz Eckstein. “A sim2real deep learning approach for the transformation of images from multiple vehicle-mounted cameras to a semantically segmented image in bird’s eye view”. In: *ITSC*. IEEE. 2020, pp. 1–7.
- [30] Thomas Roddick and Roberto Cipolla. “Predicting semantic map representations from images using pyramid occupancy networks”. In: *CVPR*. 2020, pp. 11138–11147.
- [31] Thomas Roddick, Alex Kendall, and Roberto Cipolla. “Orthographic feature transform for monocular 3d object detection”. In: *ArXiv* (2018).
- [32] Won Kyoung Roh et al. “ORA3D: Overlap Region Aware Multi-view 3D Object Detection”. In: *ArXiv abs/2207.00865* (2022).
- [33] Avishkar Saha et al. “Enabling spatio-temporal aggregation in Birds-Eye-View Vehicle Estimation”. In: *ICRA* (2021), pp. 5133–5139.
- [34] Avishkar Saha et al. “Translating Images into Maps”. In: *ICRA*. IEEE. 2022.
- [35] Sunando Sengupta et al. “Automatic dense visual semantic mapping from street-level imagery”. In: *IROS*. IEEE. 2012, pp. 857–862.

- [36] Tai Wang et al. “FCOS3D: Fully Convolutional One-Stage Monocular 3D Object Detection”. In: *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021*. IEEE, 2021, pp. 913–922.
- [37] Tai Wang et al. “Probabilistic and Geometric Depth: Detecting Objects in Perspective”. In: *Conference on Robot Learning, 8-11 November 2021, London, UK*. Ed. by Aleksandra Faust, David Hsu, and Gerhard Neumann. Vol. 164. Proceedings of Machine Learning Research. PMLR, 2021, pp. 1475–1485.
- [38] Yue Wang et al. “DETR3D: 3D Object Detection from Multi-view Images via 3D-to-2D Queries”. In: *Conference on Robot Learning, 8-11 November 2021, London, UK*. Ed. by Aleksandra Faust, David Hsu, and Gerhard Neumann. PMLR, 2021, pp. 180–191.
- [39] Xiongwei Wu, Doyen Sahoo, and Steven C. H. Hoi. “PolarNet: Learning to Optimize Polar Keypoints for Keypoint Based Object Detection”. In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.
- [40] Weixiang Yang et al. “Projecting your view attentively: Monocular road scene layout estimation via cross-view transformation”. In: *CVPR*. 2021, pp. 15536–15545.
- [41] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. “Center-based 3d object detection and tracking”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 11784–11793.
- [42] Yunpeng Zhang et al. “BEVerse: Unified Perception and Prediction in Birds-Eye-View for Vision-Centric Autonomous Driving”. In: *ArXiv* (2022).
- [43] Hongyu Zhou et al. “PersDet: Monocular 3D Detection in Perspective Bird’s-Eye-View”. In: *CoRR* abs/2208.09394 (2022).
- [44] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. “Objects as Points”. In: *CoRR* abs/1904.07850 (2019).
- [45] Benjin Zhu et al. “Class-balanced Grouping and Sampling for Point Cloud 3D Object Detection”. In: *CoRR* abs/1908.09492 (2019).
- [46] Minghan Zhu et al. “Monocular 3D Vehicle Detection Using Uncalibrated Traffic Cameras through Homography”. In: *IROS*. 2021.
- [47] Xinge Zhu et al. “Cylindrical and asymmetrical 3d convolution networks for lidar-based perception”. In: *TPAMI* (2021).
- [48] Xinge Zhu et al. “SSN: Shape Signature Networks for Multi-class Object Detection from Point Clouds”. In: *ECCV* (2020).
- [49] Jiayu Zou et al. “HFT: Lifting Perspective Representations via Hybrid Feature Transformation”. In: *ArXiv* (2022).