PowerBEV: A Powerful yet Lightweight Framework for Instance Prediction in Bird's-Eye View

Peizheng Li[*1,3] , **Shuxiao Ding**[*1,2] , **Xieyuanli Chen**[2] , **Niklas Hanselmann**[1,3] , **Marius Cordts**[1] and **Juergen Gall**[2,4]

[1]Mercedes-Benz AG, Stuttgart, Germany
[2]University of Bonn, Bonn, Germany
[3]University of Tübingen, Tübingen, Germany
[4]Lamarr Institute for Machine Learning and Artificial Intelligence, Germany
{peizheng.li, shuxiao.ding, niklas.hanselmann, marius.cordts}@mercedes-benz.com,
xieyuanli.chen@nudt.edu.cn, gall@iai.uni-bonn.de

Abstract

Accurately perceiving instances and predicting their future motion are key tasks for autonomous vehicles, enabling them to navigate safely in complex urban traffic. While bird's-eye view (BEV) representations are commonplace in perception for autonomous driving, their potential in a motion prediction setting is less explored. Existing approaches for BEV instance prediction from surround cameras rely on a multi-task auto-regressive setup coupled with complex post-processing to predict future instances in a spatio-temporally consistent manner. In this paper, we depart from this paradigm and propose an efficient novel end-to-end framework named POWERBEV, which differs in several design choices aimed at reducing the inherent redundancy in previous methods. First, rather than predicting the future in an auto-regressive fashion, POWERBEV uses a parallel, multi-scale module built from lightweight 2D convolutional networks. Second, we show that segmentation and centripetal backward flow are sufficient for prediction, simplifying previous multi-task objectives by eliminating redundant output modalities. Building on this output representation, we propose a simple, flow warping-based post-processing approach which produces more stable instance associations across time. Through this lightweight yet powerful design, POWERBEV outperforms state-of-the-art baselines on the NuScenes Dataset and poses an alternative paradigm for BEV instance prediction. We made our code publicly available[†].

1 Introduction

Accurately acquiring surrounding vehicle information is a key challenge for autonomous driving systems. In addition to the precise detection and localization of road users
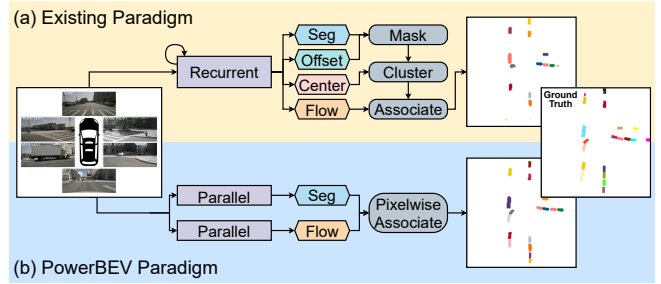

Figure 1: **PowerBEV vs. Existing Paradigm:** The existing prediction paradigm (a) outputs 4 predictions per frame using spatial RNNs. After masking out background grid cells, instance ID assignment is performed by grid clustering, followed by instance-level association. To eliminate the framework redundancy, we propose a more lightweight yet powerful parallel prediction paradigm, namely PowerBEV (b). It consists entirely of 2D CNNs supplemented by flow warping post-processing based on only 2 outputs.

at present, predicting their future motion is also of great importance, considering the high complexity and dynamics of the driving environment. A widely accepted paradigm is to decouple these tasks into separate modules. Under this paradigm, objects of interest are first detected and localized through sophisticated perception models and associated across multiple frames. Then, the past motion of these detected objects is used to forecast their potential future movement via parametric trajectory models [Luo *et al.*, 2018; Liang *et al.*, 2020]. However, by separating the perception and the motion model for forecasting, the whole system is prone to errors in the first stage.

In recent years, many works have demonstrated the potential of the bird's-eye view (BEV) representation for accurate vision-centric driving environment perception [Philion and Fidler, 2020; Huang *et al.*, 2021; Li *et al.*, 2022]. To solve the error accumulation problem, researchers seek to exploit end-to-end frameworks to determine object locations directly in the BEV and forecast global scene changes in the form of an occupancy grid map [Hu *et al.*, 2021; Akan and Güney, 2022; Zhang *et al.*, 2022]. Although employing the end-to-end paradigm, existing approaches forecast multiple, partially re-

dundant representations like segmentation map, instance centers, forward flow, and offsets pointing to instance centers as shown in Figs. 1 and 4. These redundant representations require not only various loss terms, but also complex post-processing to obtain instance predictions.

In this work, we simplify the multi-task setting used in previous works and propose an approach that requires only two output modalities: segmentation maps and flow. Specifically, we compute instance centers directly from the segmentation, allowing the omission of a redundant separate center map. This additionally eliminates the potential for inconsistencies between the estimated centers and predicted segmentation. Furthermore, contrary to the forward flow used in previous works, we compute a centripetal backward flow. This is a vector field pointing from each occupied pixel at present to its corresponding instance center in the previous frame. It combines the pixel- and instance-level association into a single pixel-wise instance assignment task. Thus, the offset head is no longer required. Morever, this design choice simplifies the association since it no longer needs multiple steps. Compared to auto-regressive models, we also find that 2D convolutional networks are sufficient for the proposed POWERBEV framework to obtain satisfactory instance predictions, which results in a lightweight yet powerful framework.

We evaluate our approach on the NuScenes dataset [Caesar et al., 2020], where our method outperforms existing frameworks and achieves state-of-the-art instance prediction performance. We further perform ablation studies to validate the design of our powerful but lightweight framework.

Our main contributions can be summarized as follows:

- We propose POWERBEV, a novel and elegant vision-based end-to-end framework that only consists of 2D-convolutional layers to perform perception and forecasting of multiple objects in BEVs.

- We demonstrate that over-supervision caused by redundant representations impairs the forecasting capability. In contrast, our method accomplishes both semantic and instance-level agent prediction by simply forecasting segmentation and centripetal backward flow.

- The proposed assignment based on centripetal backward flow is superior to the previous forward flow in combination with the traditional Hungarian Matching algorithm.

2 Related Work

2.1 BEV for Camera-based 3D Perception

While LiDAR-based perception approaches often map a 3D point cloud onto the BEV plane and perform BEV segmentation [Fei et al., 2021; Peng et al., 2021] or 3D bounding box regression [Yang et al., 2018; Lang et al., 2018; Yin et al., 2021], the transformation of a monocular camera image into a BEV representation remains an ill-posed problem. Although there are methods [Fei et al., 2020; Liu et al., 2022; Dong et al., 2022; Liang et al., 2022] that combine LiDAR and camera data to generate BEVs, they rely on accurate multi-sensor calibration and synchronization.

LSS (Lift Splat Shoot) [Philion and Fidler, 2020] can be regarded as the first work that lifts 2D features to 3D and projects the lifted features onto the BEV plane. It discretizes the depth and predicts a distribution over depth. The image features are then scaled and distributed across the depth dimension according to the distribution. BEVDet [Huang et al., 2021] adapts LSS to 3D object detection from BEV feature maps. Tesla AI Day 2021 [Tesla, 2021] first proposes to use a Transformer architecture [Vaswani et al., 2017] to fuse multi-view camera features into BEV feature maps, where the cross-attention between dense BEV queries and perspective image features acts as the view transformation. This approach is further improved by leveraging camera calibration and deformable attention [Zhu et al., 2020] in BEVFormer [Li et al., 2022] and BEVSegFormer [Peng et al., 2023] to reduce the quadratic complexity of Transformers. Furthermore, it has been shown that temporal modeling of BEV features achieves a significant performance improvement for 3D detection [Li et al., 2022; Huang and Huang, 2022; Jiang et al., 2022] at the cost of a high computation and memory consumption. Unlike detection or segmentation, the forecasting task naturally needs temporal modeling of the historical information. To tackle this, our approach extracts spatio-temporal information using a lightweight fully-convolutional network on top of LSS, which is both effective and efficient.

2.2 BEV-based Future Prediction

Early BEV-based prediction methods render the past trajectories into a BEV image and use CNNs to encode the rasterized input [Bansal et al., 2018; Hong et al., 2019; Chai et al., 2019], which assumes perfect detection and tracking of the objects. Another line of works conducts end-to-end trajectory forecasting directly from LiDAR point clouds [Casas et al., 2018; Luo et al., 2018; Casas et al., 2020; Liang et al., 2020]. Unlike instance-level trajectory prediction, MotionNet [Wu et al., 2020] and MP3 [Casas et al., 2021] tackle the forecasting task by a motion (flow) field for each occupancy grid.

In contrast to the above mentioned approaches that rely on LiDAR data, FIERY [Hu et al., 2021] first predicts a BEV instance segmentation solely from multi-view camera data. FIERY extracts multi-frame BEV features following LSS [Philion and Fidler, 2020], fuses them into a spatio-temporal state using a recurrent network, and then conducts a probabilistic instance prediction. StretchBEV [Akan and Güney, 2022] improves FIERY using a stochastic temporal model with stochastic residual updates. BEVerse [Zhang et al., 2022] proposes an iterative flow warping in latent space for prediction in a multi-task BEV perception framework. These approaches follow Panoptic-DeepLab [Cheng et al., 2020] that utilizes four different heads to compute a semantic segmentation map, instance centers, per-pixel centripetal offsets, and future flow. They rely on complex post-processing to generate the final instance prediction from these four representations. In this paper, we show that only two heads, namely semantic segmentation and centripetal backward flow, together with a simplified post-processing are sufficient for future instance prediction.
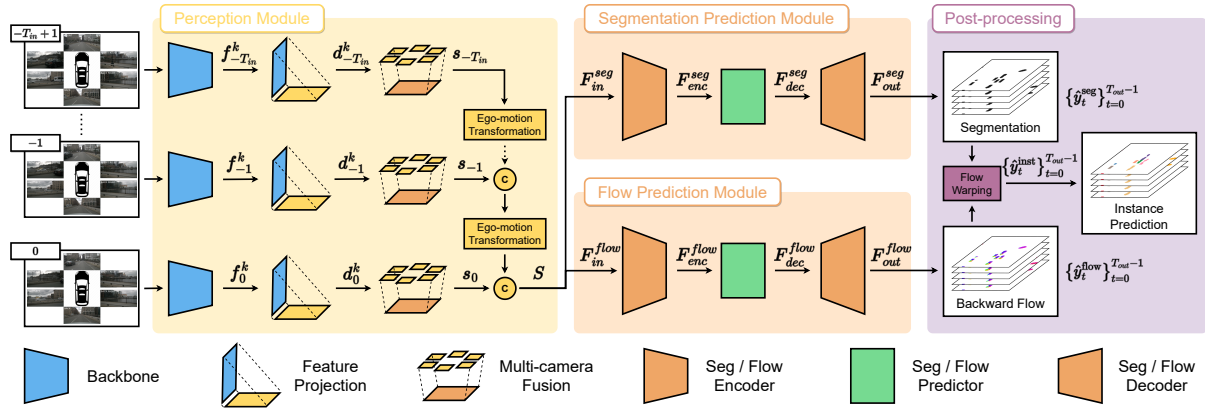
Figure 2: **Architecture of our Proposed End-to-End Framework:** In PowerBEV, the perspective features extracted by the perception module (yellow area) from surrounding camera images of each frame are projected into the BEV plane and then fused and stacked into the current global dynamic state. Subsequently, two independent prediction modules with the same structure (orange area) take the current state as input and predict the segmentation maps and centripetal backward flow for the future frames. Finally, future multi-frame instance predictions are generated by the flow warping post-processing (purple area).

3 Approach

In this section, we outline our proposed end-to-end framework. An overview of the approach is illustrated in Figure 2. It consists of three main parts: a perception module, a prediction module and a post-processing stage. The perception module follows LSS [Philion and Fidler, 2020] and takes M multi-view camera images for T_{in} timestamps as input and lifts them into T_{in} BEV feature maps (see Section 3.1). The prediction module then fuses the spatio-temporal information contained in the extracted BEV features (see Section 3.2) and predicts a sequence of segmentation maps and centripetal backward flows for T_{out} future frames in parallel (see Section 3.3). Finally, future instance predictions are recovered from the predicted segmentation and flow through a warping-based post-processing (see Section 3.4). In the following we describe each of the involved components in detail.

3.1 LSS-based Perception Module

To obtain visual features for prediction, we follow previous works [Hu *et al.*, 2021; Akan and Güney, 2022] and build on LSS [Philion and Fidler, 2020] to extract BEV feature grids from surround camera images. More specifically, for each camera image $k \in \{1, \ldots, 6\}$ at time t, we apply a shared EfficientNet [Tan and Le, 2019] backbone to extract perspective features $f_t^k \in \mathbb{R}^{(C_p+D_p) \times H_p \times W_p}$, where we designate the first C_p channels of f_t^k to represent a context feature $f_{t,C}^k \in \mathbb{R}^{C_p \times H_p \times W_p}$ and the following D_p channels to represent a categorical depth distribution $f_{t,D}^k \in \mathbb{R}^{C_p \times H_p \times W_p}$. A 3D feature tensor $d_t^k \in \mathbb{R}^{C_p \times D_p \times H_p \times W_p}$ is constructed by means of the outer product

$$d_t^k = f_{t,C}^k \otimes f_{t,D}^k, \tag{1}$$

which represents a lifting of the context feature $f_{t,C}^k$ into different depths D_p according to the estimated depth distribution confidence $f_{t,D}^k$. Afterwards, the per-camera feature distribution maps d_t^k at each timestamp are transformed to the ego-vehicle-centered coordinate system, leveraging known intrin-
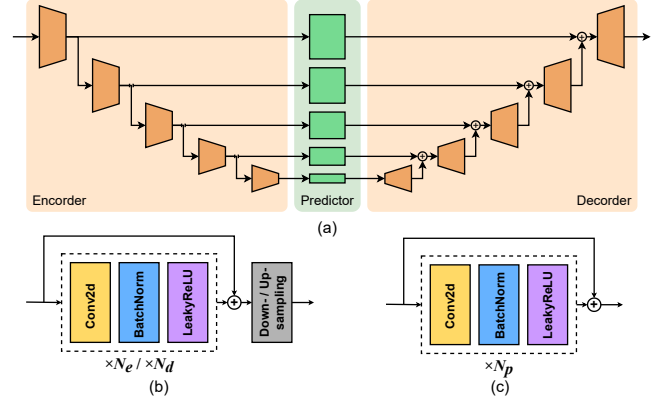

Figure 3: **Architecture of the Multi-scale Prediction Model:** (a) Overview; (b) Encoder/Decoder block with the down-/up-sampling layer; (c) Predictor block. N_e, N_d, N_p denotes the number of encoder, decoder and predictor blocks respectively.

sics and extrinsics of the corresponding cameras. They are then weighted along the height dimension to obtain the global BEV state $s_t \in \mathbb{R}^{C_{in} \times H \times W}$ at timestamp t, where C_{in} is the number of state channels and (H, W) the grid size of the BEV state maps. Finally, all BEV states $\{s_t\}_{t=-T_{in}+1}^{0}$ are unified to the current frame and stacked as in FIERY [Hu *et al.*, 2021], thus representing current global dynamics $S \in \mathbb{R}^{C_{in} \times T_{in} \times H \times W}$ independent of ego-vehicle positions.

3.2 Multi-scale Prediction Module

Having obtained a compact representation S of the past context, we use a multi-scale U-Net-like encoder-decoder architecture that takes the observed BEV feature maps as input and predicts future segmentation maps and centripetal backward flow fields, as shown in Figure 3. To achieve the spatio-temporal feature processing using only 2D convolutions, we collapse the time and feature dimensions into one single dimension, resulting in an input tensor $F_{in} \in \mathbb{R}^{(C_{in} \times T_{in}) \times H \times W}$. The encoder first downsamples F_{in} spatially step by step, pro-
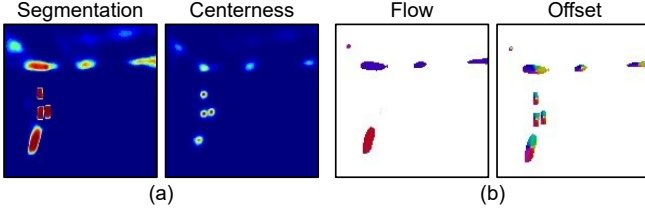
Figure 4: **Task Similarities:** (a) segmentation probability and centerness are both Gaussian distributions; (b) flow and offset are both regression tasks within occupied regions.

ducing multi-scale BEV features $F_{\mathrm{enc}} \in \mathbb{R}^{(C_i \times T_{\mathrm{in}}) \times \frac{H}{2^i} \times \frac{W}{2^i}}$, where $i \in \{1, \ldots, 5\}$. In an intermediate predictor stage, the features are mapped from $C_i \times T_{\mathrm{in}}$ to $C_i \times T_{\mathrm{out}}$ to get $F_{\mathrm{dec}} \in \mathbb{R}^{(C_i \times T_{\mathrm{out}}) \times \frac{H}{2^i} \times \frac{W}{2^i}}$. Finally, the decoder, which mirrors the encoder, reconstructs the future BEV features $F_{\mathrm{out}} \in \mathbb{R}^{(C_{\mathrm{out}} \times T_{\mathrm{out}}) \times H \times W}$ at the original scale.

Each branch is supervised to predict future segmentation maps or centripetal backward flow fields, respectively. Considering the differences in tasks and supervision, we use the same architecture for each branch but without weight-sharing. Compared to previous work building on spatial LSTMs or spatial GRUs, our architecture leverages only 2D convolutions and largely alleviates the limitations of spatial RNNs in solving long-range temporal dependencies.

3.3 Multi-task Settings

Existing approaches [Hu *et al.*, 2021; Akan and Güney, 2022; Zhang *et al.*, 2022] follow a bottom-up pipeline [Cheng *et al.*, 2020] that generates instance segmentation for each frame and then associates instances across frames using Hungarian Matching (HM) [Kuhn, 1955] based on the forward flow. Consequently, four different heads are required: semantic segmentation, centerness, future forward flow and per-pixel centripetal offsets in BEV (Figure 1.a). This leads to model redundancy and instability due to multi-task training.

By comparison, we first find that both flow and centripetal offsets are regression tasks within the instance mask (Figure 4.b) and the flow can be understood as the motion offset. In addition, both quantities are combined with the centerness in two stages: (1) centripetal offset groups pixels to the predicted instance center in each frame to assign pixels to instance IDs; (2) flow is used to match the centers in two consecutive frames for instance ID association. Based on the above analysis, it is intuitive to solve both tasks using a unified representation.

To this end, we propose the *backward centripetal flow field*, which is the displacement vector from each foreground pixel at time t to the object center of the associated instance identity at time $t-1$. This unifies the pixel-to-pixel backward flow vector and the centripetal offset vector into a single representation. Using our proposed flow, each occupied pixel can be directly associated to an instance ID in the previous frame. This eliminates the need for an additional clustering step that assigns pixels to instances, simplifying the two-stage post-processing used in previous works into a single-stage association task (Figure 1.b). This instance association mechanism is further discussed in detail in Section 3.4.
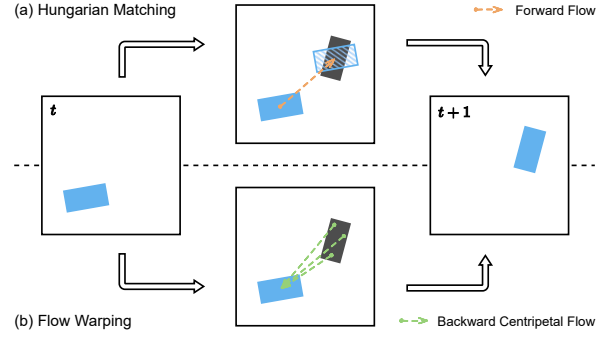

Figure 5: **Instance Matching Illustration:** The top branch is the Hungarian Matching algorithm (a) with forward flow as used in FIERY. The bottom branch is our backward flow warping operation (b) with centripetal backward flow.

Furthermore, we find that the predictions of the semantic segmentation map and the centerness reveals a very high similarity since the centerness essentially corresponds to the center positions of the semantic instances (Figure 4.a). Thus, we propose to directly infer the object centers by extracting the local maxima in the predicted segmentation map using the method by [Zhou *et al.*, 2019]. This eliminates the need to seperately predict centerness.

To summarize, our network produces only two outputs: the semantic segmentation $\left\{\hat{y}_t^{\mathrm{seg}}\right\}_{t=0}^{T_{\mathrm{out}}-1}$ and the backward centripetal flow $\left\{\hat{y}_t^{\mathrm{flow}}\right\}_{t=0}^{T_{\mathrm{out}}-1}$. We use top-$k$ cross-entropy [Berrada *et al.*, 2018] with $k = 25\%$ as segmentation loss and a smooth ℓ_1 distance as flow loss. The overall loss function is given by:

$$\mathcal{L} = \frac{1}{T_{\mathrm{out}}} \left\{ \sum_{t=0}^{T_{\mathrm{out}}-1} \gamma^t \left(\lambda_1 \mathcal{L}_{\mathrm{ce}}(\hat{y}_t^{\mathrm{seg}}, y_t^{\mathrm{seg}}) + \lambda_2 \mathcal{L}_{\ell_1}(\hat{y}_t^{\mathrm{flow}}, y_t^{\mathrm{flow}}) \right) \right\}, \tag{2}$$

with a future discount parameter $\gamma = 0.95$ and balance factors λ_1 and λ_2 that are dynamically updated using uncertainty weighting [Kendall *et al.*, 2018].

3.4 Instance Association

For the instance predictions, we need to associate the future instances $\left\{\hat{y}_t^{\mathrm{inst}}\right\}_{t=0}^{T_{\mathrm{out}}-1}$ over time. Existing methods project instance centers to the next frame using the forward flow, and then match the nearest agent centers using Hungarian Matching [Kuhn, 1955] as shown in Figure 5.a. This method performs an *instance-level association*, where an instance identity is represented by its center. Therefore, only the flow vector located on the object center is used for motion prediction. This has two disadvantages: Firstly, object rotation is not considered and, secondly, a single displacement vector is more prone to errors than multiple displacement vectors covering the entire instance. In practice, this can lead to overlapping projected instances, resulting in incorrect ID assignments. This is particularly evident for close objects over a long prediction horizon.

Leveraging our backward centripetal flow, we further propose a *warping-based pixel-level association* to tackle the

above mentioned problems. An illustration of our association method is shown in Figure 5.b. For each foreground grid cell, this operation directly propagates the instance ID from the pixel at the flow vector destination in the previous frame to the current frame. Using this method, the instance ID of each pixel is assigned seperately, yielding a pixel-level association. Compared to the instance-level association, our method is tolerant to more severe flow prediction errors, because neighboring grid cells around the true center are inclined to share the same identity and errors tend to occur at individual peripheral pixels. In addition, by using backward flow warping, multiple future positions can be associated with one pixel in the previous frame [Mahjourian *et al.*, 2022]. This is beneficial for the multi-modal future prediction.

As described, the backward association needs instance IDs at the previous frame. A special case is the instance segmentation generation of the first frame ($t = 0$), where no instance information at its previous frame ($t = -1$) is available. Thus, only for the timestamp $t = 0$, we assign instance IDs by grouping pixels to past instance centers. This is similar to [Cheng *et al.*, 2020] but without the usage of an additional centerness head since the centers are extracted from the semantic segmentation as discussed in Section 3.3.

4 Experimental Evaluation

4.1 Experimental Setup

Dataset We evaluate our method and compare it with state-of-the-art frameworks on the NuScenes Dataset [Caesar *et al.*, 2020], a widely used public dataset for perception and prediction in autonomous driving. It contains 1 000 driving scenes collected from Boston and Singapore, split into training, validation, and test sets with 750, 150, and 150 scenes, respectively. Each scene consists of 20 seconds of traffic data and is labeled with semantic annotations at 2 Hz frequency.

Implementation Details We follow the setup from existing studies [Hu *et al.*, 2021; Akan and Güney, 2022; Zhang *et al.*, 2022] that use the information of 3 frames corresponding to the past 1 s (including the present frame) to predict the semantic segmentation, flow, and instance motion of 4 frames corresponding to the future 2 s. Input images are scaled and cropped to the size of 480×224, while the BEV map corresponds to a grid size of 200×200. To evaluate the model performance at different perceptual scopes, two spatial resolutions are adopted: (1) $100\,\text{m} \times 100\,\text{m}$ area with $0.5\,\text{m}$ resolution (long) and (2) $30\,\text{m} \times 30\,\text{m}$ area with $0.15\,\text{m}$ resolution (short). Using Adam optimizer with a learning rate of 3×10^{-4}, the end-to-end framework is trained for 20 epochs on four Tesla V100 GPUs with 16 GB memory and with a batch size of 8. Our implementation is based on the code of FIERY [Hu *et al.*, 2021].

Metrics We follow the evaluation procedure of FIERY [Hu *et al.*, 2021]. To evaluate the segmentation accuracy, we use the Intersection-over-Union (IoU) as evaluation metric for the segmentation quality, i.e.:

$$\text{IoU}(\hat{y}_t^{\text{seg}}, y_t^{\text{seg}}) = \frac{1}{T_{\text{out}}} \sum_{t=0}^{T_{\text{out}}-1} \frac{\sum_{h,w} \hat{y}_t^{\text{seg}} \cdot y_t^{\text{seg}}}{\sum_{h,w} \hat{y}_t^{\text{seg}} + y_t^{\text{seg}} - \hat{y}_t^{\text{seg}} \cdot y_t^{\text{seg}}}, \tag{3}$$

where $\hat{y}_t^{\text{seg}}$ and y_t^{seg} are the predicted and ground truth semantic segmentations at timestamp t, respectively.

We use video panoptic quality (VPQ) as a more adequate metric for the instance prediction task. It consists of two parts: (1) recognition quality (RQ), i.e., the ID consistency of detected instances across the entire time horizon; (2) segmentation quality (SQ), i.e., the accuracy of the instance segmentation itself. VPQ is calculated as

$$\text{VPQ}(\hat{y}_t^{\text{inst}}, y_t^{\text{inst}}) = \sum_{t=0}^{T_{\text{out}}-1} \frac{\sum_{(p_t,q_t) \in TP_t} IoU(p_t, q_t)}{|TP_t| + \frac{1}{2}|FP_t| + \frac{1}{2}|FN_t|}, \tag{4}$$

where TP_t, FP_t and FN_t correspond to true positives, false positives and false negatives at time point t, respectively.

Baseline Methods We compare PowerBEV with the three state-of-the-art methods FIERY [Hu *et al.*, 2021], Stretch-BEV [Akan and Güney, 2022], and BEVerse [Zhang *et al.*, 2022]. FIERY and StretchBEV have the same experimental setup as our work, except for a larger batch size of 12 on four Tesla V100 GPUs with 32GB memory each. BEVerse upgrades the backbone to the more advanced SwinTransformer [Liu *et al.*, 2021], significantly increasing the image input size to 704×256 and the batch size to 32 using 32 NVIDIA GeForceRTX 3090 GPUs to train the end-to-end model. To demonstrate the effectiveness of our framework, we intentionally do not use large models or large image size like BEVerse, but limit ourselves to the FIERY setting in terms of FLOPs and GPU memory usage for a fair comparison.

Label Generation Optimization In preliminary experiments, we found that the data pre-processing and label generation of the baseline method FIERY introduces systematic errors. The original data pre-processing consists of three steps: (1) Surrounding vehicle positions are transformed from the global coordinate system (GCS) to the ego coordinate system (ECS) at the corresponding timestamp based on the ego-motion. At this timestamp, the instance parameters are then rendered into BEV maps to generate segmentation, instance, centerness and offset ground truth. (2) The generated instance map is warped by ego-motion to obtain the warped instance map of the next frame. The flow map is calculated as the geometric center displacement of the future warped instance from the current instance where all pixels of this instance share the same flow. (3) During pre-processing, all BEV maps are transformed back into GCS. As all these steps are performed in discrete BEV space, the two inverse coordinate transformations in (1) and (3) introduce considerable numerical errors into the ground truth maps. Additionally, these errors further spread to the flow generation: Even for a stationary vehicle, its BEV segmentation will slightly jitter over time and may also be assigned to non-zero flow values. Overall, these errors in the ground truth generation significantly affect the prediction performance.

To solve the above problems, we modify the process in both training and validation as follows: We propose to determine the perceptual area based on ego-vehicle position and render BEV maps directly in GCS to avoid errors caused by these two inverse transformations. In addition, the flow ground truth is calculated using the instance maps of two adjacent frames without warping. We also filter the flow ground

Table 1: Instance prediction benchmark results on NuScenes dataset. † uses a larger image size of 704×256, others use 480×224. Models with ‡ use our optimized label generation.

Method	IoU		VPQ	
	Short	Long	Short	Long
FIERY	59.4	36.7	50.2	29.9
FIERY‡ (repr.)	58.3	38.2	48.2	30.9
StretchBEV	55.5	37.1	46.0	29.0
BEVerse†	60.3	38.7	52.2	33.3
PowerBEV‡	**62.5**	**39.3**	**55.5**	**33.8**

Table 2: Comparison of different multi-task training setups with different heads used. Our method [D] requires fewer heads than the typical baseline approach [A] and yields best results.

Model	Segmentation	Flow	Center	Offset	IoU		VPQ	
					Short	Long	Short	Long
[A]	✓	✓	✓	✓	61.7	39.2	52.8	33.5
[B]	✓	✓	✓	-	62.4	39.2	55.3	33.1
[C]	✓	✓	-	✓	**62.5**	39.2	51.8	33.6
[D]	✓	✓	-	-	**62.5**	**39.3**	**55.5**	**33.8**

truth by zeroing all values below a threshold to eliminate noise and artifacts of stationary vehicles.

4.2 Comparison to Baselines

We first compare the performance of our method to baseline frameworks in Table 1. We also reproduced FIERY [Hu *et al.*, 2021] with our proposed label generation method, c.f. Section 4.1, yielding improvements in the long-range domain, which is essential for the safety of autonomous vehicles.

In comparison with the baseline methods, our approach achieves significant improvements in terms of both evaluation metrics IoU and VPQ for both perceptual range settings. For the long-range setting, PowerBEV outperforms the reproduced FIERY by 1.1% IoU and 2.9% VPQ. Furthermore, PowerBEV performs better than BEVerse in all metrics, despite using a lower input image resolution and less parameters. Compared to other baselines that introduce a stochastic process in their model, PowerBEV is a deterministic approach which is able to accomplish accurate forecasting. This also shows the ability of the backward flow in capturing multi-modal futures [Mahjourian *et al.*, 2022].

Figure 6 visualizes qualitative results of our method. We show the comparison with FIERY in three typical driving scenarios: an urban scene with dense dynamic traffic, a parking lot with many static vehicles and a rainy scene. Our approach provides more precise and reliable trajectory predictions for the most common dense traffic scenes, which becomes particularly evident in the first example with the vehicles turning into the side street on the left side of the ego-vehicle. While FIERY only makes a few vague guesses about vehicle locations and has difficulties handling their dynamics, our approach, on the other hand, provides sharp object boundaries that match the real vehicle shapes better as well as their possible future trajectories. Furthermore, as evident from the comparison in the second example, our framework can detect

Table 3: Comparison of different post-processing methods.

Model	Post-processing	Time Horizon	IoU		VPQ	
			Short	Long	Short	Long
[E]	Instance-level HM	2 s	60.0	38.3	46.9	31.9
[F]	Pixel-level Warping	2 s	**61.7**	**39.2**	**52.8**	**33.5**
[G]	Instance-level HM	8 s	48.8	33.4	39.6	27.2
[H]	Pixel-level Warping	8 s	**57.7**	**34.8**	**48.2**	**29.4**

Table 4: Ablation study regarding generalization capabilities of our design. We applied our multi-task setup and post-processing to the FIERY RNN-based backbone, yielding significant improvements.

Method	IoU		VPQ	
	Short	Long	Short	Long
FIERY (reproduced)	58.3	38.2	48.2	30.9
FIERY + Our Contributions	**61.6**	**38.9**	**56.4**	**32.4**

vehicles located at long distances, where FIERY fails. In addition, our method detects the trucks occluded by walls in the rainy scene, which are difficult to spot even for human eyes.

4.3 Ablation Studies

We conduct several ablation studies of PowerBEV to analyze the effectiveness of different components in our framework.

Multi-task Learning: We first investigate the effect of our reduced number of tasks and analyze whether typical problems in multi-task learning affect the performance, e.g. task balancing and training instability. To this end, we keep the segmentation and backward centripetal flow heads as well as our post-processing unchanged, but attach additional heads to our prediction module (Section 3.2). These heads are trained according to the objectives from the original FIERY baseline, i.e. centerness and centripetal offset. The additional heads are trained jointly with segmentation and flow heads but not used in the post-processing. All the experiments use uncertainty weighting [Kendall *et al.*, 2018] to balance different tasks.

We vary the number and type of the additional training objectives, as shown in Table 2. Our approach with only two heads (Model [D]) performs better than all the other variants. Adding the center (Model [B]) or offset (Model [C]) head negatively impacts various metrics. For example, we observe a large decrease of VPQ for the short-range setting when training with the offset head. The reason is that the backward centripetal flow is equal to the centripetal offset for static objects but different for moving objects, thus both training objectives lead to confusions during network training. Compared to the Model [A] that uses all the four heads as in existing works, Model [D] achieves improvements of 2.7% for the short and 0.3% for the long-range setting in VPQ. Another observation is that different loss terms converge at different speeds, making them difficult to support each other during training. These findings support that eliminating redundant tasks is one of the sources for the performance improvements of our approach. Although uncertainty weighting avoids tuning of loss weights, our approach directly reduces the amount of hyper-parameters, which simplifies the balancing of different train-
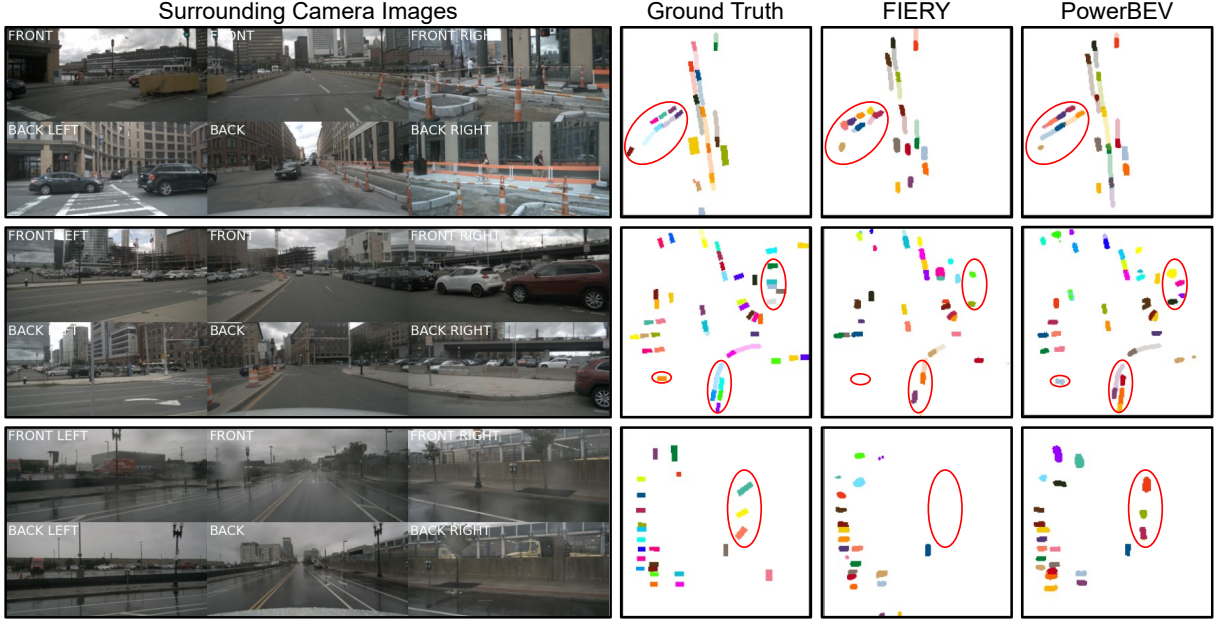
Figure 6: **Visualization of Instance Predictions:** We compare our approach with ground truth and the baseline FIERY. Each vehicle instance is assigned to a unique color and the predicted trajectory is represented by the same color with slight transparency.

ing objectives.

Post-processing: We further show the effectiveness of our proposed warping-based association for post-processing. To this end, we want to compare our work with the traditionally used Hungarian Matching (HM), c.f. Figure 1.a. As our network does not offer all required outputs for such a setup, we use the training setting of Model [A] in Table 2 as a baseline. As this model offers all four heads, we can directly compare both post-processing methods on the same network. We also evaluate the prediction results over a longer prediction horizon (8s) to show the ability in maintaining ID consistency of different methods.

As evident from the the upper part of Table 3, our method (Model [F]) outperforms the HM-based instance-level association (Model [E]) in both IoU and VPQ. For the longer time horizon, we observe an even more significant improvement of our method, especially for the short-range setting. Figure 7 provides a more detailed illustration of the metrics at each frame. It can be observed that our flow warping remains stable over the long time horizon while the HM-based approach shows a clear performance decrease over time. We attribute this to the fact that pixel-level association fully utilizes the network predictions of all pixels inside the instance boundary to boost the robustness. In addition, our warping-based association is a many-to-one matching from the present to the past, which further increases temporal stability.

4.4 Generalization

Next, we show that our design is not limited to CNN-based prediction models. Our multi-task setting and flow warping operation can be ported to other model structures. To verify this, we applied both to FIERY, keeping the remaining model structure and parameters unchanged. The results
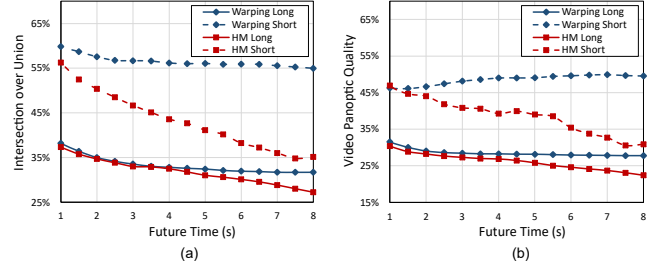

Figure 7: Comparison of different post-processing methods for the 8s time horizon: (a) IoU and (b) VPQ.

in Table 4 confirm that our approach also generalizes well to RNN-based models. Hence, we believe that PowerBEV is a promising new paradigm for instance prediction and can serve as basis for future work.

5 Conclusion

In this work, we presented a novel framework POWERBEV for future instance prediction in BEV. Our approach only forecasts semantic segmentation and centripetal backward flow using 2D-CNNs in a parallel scheme. It furthermore adopts a novel post-processing, which better handles multi-modal future motions, achieving state-of-the-art instance prediction performance in the NuScenes benchmark. We provided thorough ablation studies that analyze our method and show its effectiveness. The experiments confirm that POWERBEV is more lightweight than previous approaches albeit yielding an improved performance. Hence, we believe that this method could become a new design paradigm for instance prediction in BEV.

References

[Akan and Güney, 2022] Adil Kaan Akan and Fatma Güney. Stretchbev: Stretching future instance prediction spatially and temporally. In *European Conference on Computer Vision*, 2022.

[Bansal *et al.*, 2018] Mayank Bansal, Alex Krizhevsky, and Abhijit S. Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *ArXiv*, abs/1812.03079, 2018.

[Berrada *et al.*, 2018] Leonard Berrada, Andrew Zisserman, and M. Pawan Kumar. Smooth loss functions for deep top-k classification. *ArXiv*, abs/1802.07595, 2018.

[Caesar *et al.*, 2020] 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/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[Casas *et al.*, 2018] Sergio Casas, Wenjie Luo, and Raquel Urtasun. Intentnet: Learning to predict intention from raw sensor data. In *Conference on Robot Learning*, pages 947–956. PMLR, 2018.

[Casas *et al.*, 2020] Sergio Casas, Cole Gulino, Renjie Liao, and Raquel Urtasun. Spagnn: Spatially-aware graph neural networks for relational behavior forecasting from sensor data. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 9491–9497. IEEE, 2020.

[Casas *et al.*, 2021] Sergio Casas, Abbas Sadat, and Raquel Urtasun. Mp3: A unified model to map, perceive, predict and plan. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14403–14412, 2021.

[Chai *et al.*, 2019] Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. In *Conference on Robot Learning*, 2019.

[Cheng *et al.*, 2020] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 12475–12485, 2020.

[Dong *et al.*, 2022] Hao Dong, Xianjing Zhang, Xuan Jiang, Jinchao Zhang, Jintao Xu, Rui Ai, Weihao Gu, Huimin Lu, Juho Kannala, and Xieyuanli Chen. Superfusion: Multi-level lidar-camera fusion for long-range hd map generation and prediction. *ArXiv*, abs/2211.15656, 2022.

[Fei *et al.*, 2020] Juncong Fei, Wenbo Chen, Philipp Heidenreich, Sascha Wirges, and Christoph Stiller. Semanticvoxels: Sequential fusion for 3d pedestrian detection using lidar point cloud and semantic segmentation. *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 185–190, 2020.

[Fei *et al.*, 2021] Juncong Fei, Kunyu Peng, Philipp Heidenreich, Frank Bieder, and Christoph Stiller. Pillarsegnet: Pillar-based semantic grid map estimation using sparse lidar data. *IEEE Intelligent Vehicles Symposium (IV)*, pages 838–844, 2021.

[Hong *et al.*, 2019] Joey Hong, Benjamin Sapp, and James Philbin. Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8454–8462, 2019.

[Hu *et al.*, 2021] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: Future instance prediction in bird's-eye view from surround monocular cameras. In *IEEE/CVF International Conference on Computer Vision*, pages 15273–15282, 2021.

[Huang and Huang, 2022] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *ArXiv*, abs/2203.17054, 2022.

[Huang *et al.*, 2021] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *ArXiv*, abs/2112.11790, 2021.

[Jiang *et al.*, 2022] Yan Jiang, Li Zhang, Zhenwei Miao, Xiatian Zhu, Jin Gao, Weiming Hu, and Yulin Jiang. Polarformer: Multi-camera 3d object detection with polar transformers. *ArXiv*, abs/2206.15398, 2022.

[Kendall *et al.*, 2018] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

[Kuhn, 1955] Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics (NRL)*, 52, 1955.

[Lang *et al.*, 2018] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12689–12697, 2018.

[Li *et al.*, 2022] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *ArXiv*, abs/2203.17270, 2022.

[Liang *et al.*, 2020] Ming Liang, Bin Yang, Wenyuan Zeng, Yun Chen, Rui Hu, Sergio Casas, and Raquel Urtasun. Pnpnet: End-to-end perception and prediction with tracking in the loop. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11553–11562, 2020.

[Liang *et al.*, 2022] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. *ArXiv*, abs/2205.13790, 2022.

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[Liu *et al.*, 2022] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. *ArXiv*, abs/2205.13542, 2022.

[Luo *et al.*, 2018] Wenjie Luo, Bin Yang, and Raquel Urtasun. Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In *IEEE conference on Computer Vision and Pattern Recognition*, pages 3569–3577, 2018.

[Mahjourian *et al.*, 2022] Reza Mahjourian, Jinkyu Kim, Yuning Chai, Mingxing Tan, Ben Sapp, and Dragomir Anguelov. Occupancy flow fields for motion forecasting in autonomous driving. *IEEE Robotics and Automation Letters*, 7(2):5639–5646, 2022.

[Peng *et al.*, 2021] Kunyu Peng, Juncong Fei, Kailun Yang, Alina Roitberg, Jiaming Zhang, Frank Bieder, Philipp Heidenreich, Christoph Stiller, and Rainer Stiefelhagen. Mass: Multi-attentional semantic segmentation of lidar data for dense top-view understanding. *IEEE Transactions on Intelligent Transportation Systems*, 23:15824–15840, 2021.

[Peng *et al.*, 2023] Lang Peng, Zhirong Chen, Zhangjie Fu, Pengpeng Liang, and Erkang Cheng. Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5935–5943, 2023.

[Philion and Fidler, 2020] 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. Springer, 2020.

[Tan and Le, 2019] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[Tesla, 2021] Tesla. Tesla ai day 2021, Aug. 2021.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wu *et al.*, 2020] Pengxiang Wu, Siheng Chen, and Dimitris N Metaxas. Motionnet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 11385–11395, 2020.

[Yang *et al.*, 2018] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *IEEE conference on Computer Vision and Pattern Recognition*, pages 7652–7660, 2018.

[Yin *et al.*, 2021] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021.

[Zhang *et al.*, 2022] Yunpeng Zhang, Zheng Hua 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*, abs/2205.09743, 2022.

[Zhou *et al.*, 2019] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *ArXiv*, abs/1904.07850, 2019.

[Zhu *et al.*, 2020] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *ArXiv*, abs/2010.04159, 2020.