Temporal Enhanced Training of Multi-view 3D Object Detector via Historical Object Prediction

Zhuofan Zong[1*] Dongzhi Jiang[1,2*] Guanglu Song[1] Zeyue Xue[3]
Jingyong Su[4] Hongsheng Li[2,5,6†] Yu Liu[1†]
[1]SenseTime Research [2]CUHK MMLAB [3]HKU [4]HITSZ
[5]CPII [6]Shanghai AI Laboratory
{zongzhuofan,jdzcarr7,liuyuisanai}@gmail.com hsli@ee.cuhk.edu.hk

Abstract

In this paper, we propose a new paradigm, named Historical Object Prediction (HoP) for multi-view 3D detection to leverage temporal information more effectively. The HoP approach is straightforward: given the current timestamp t, we generate a pseudo Bird's-Eye View (BEV) feature of timestamp t-k from its adjacent frames and utilize this feature to predict the object set at timestamp t-k. Our approach is motivated by the observation that enforcing the detector to capture both the spatial location and temporal motion of objects occurring at historical timestamps can lead to more accurate BEV feature learning. First, we elaborately design short-term and long-term temporal decoders, which can generate the pseudo BEV feature for timestamp t-k without the involvement of its corresponding camera images. Second, an additional object decoder is flexibly attached to predict the object targets using the generated pseudo BEV feature. Note that we only perform HoP during training, thus the proposed method does not introduce extra overheads during inference. As a plug-and-play approach, HoP can be easily incorporated into state-of-the-art BEV detection frameworks, including BEVFormer and BEVDet series. Furthermore, the auxiliary HoP approach is complementary to prevalent temporal modeling methods, leading to significant performance gains. Extensive experiments are conducted to evaluate the effectiveness of the proposed HoP on the nuScenes dataset. We choose the representative methods, including BEVFormer and BEVDet4D-Depth to evaluate our method. Surprisingly, HoP achieves 68.5% NDS and 62.4% mAP with ViT-L on nuScenes test, outperforming all the 3D object detectors on the leaderboard. Codes are available at `https://github.com/Sense-X/HoP`.

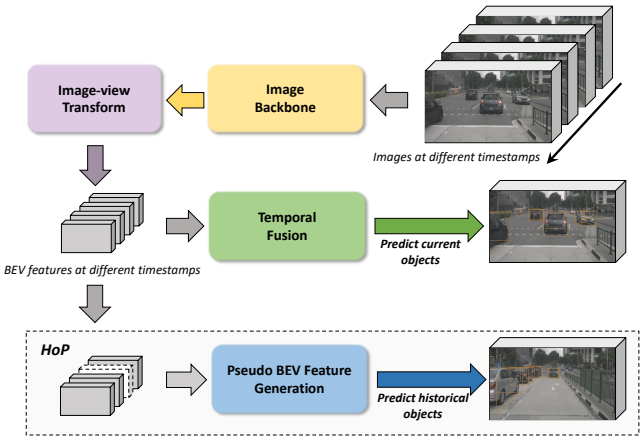

Figure 1: **Incorporating HoP into 3D object detector with other temporal fusion methods [8, 7, 32, 12, 24].** Our HoP is plug-and-play and complementary to them.

1. Introduction

Camera-only 3D detection from multi-view images is a challenging task for autonomous driving and has received increasing attention recently. Introducing the Bird's-Eye View (BEV) representation [25, 14, 8, 30, 18, 2] with the temporal features aggregation has become the superior design manner for camera-only 3D perception with multi-camera images. Based on this, a series of detectors [7, 15, 12, 24, 31, 19, 17] delve into the elaborated temporal information fusion methods and achieve significant breakthroughs. They consider the regions in 3D space and aggregate the image features corresponding to these hypothesis locations from multiple timestamps. These outstanding advances reveal the great potential of temporal information in camera-only 3D detection.

Beyond these temporal fusion mechanisms, in this paper, we propose a new paradigm for leveraging temporal information to enhance the temporal modeling ability of

*Equal contribution. Work done during internship at SenseTime Research. †Corresponding authors.

the 3D object detector. It's a temporal-based auxiliary task only adopted during training stage, namely Historical Object Prediction (HoP). Our approach is motivated by the observation that enforcing the detector to capture both the spatial location and temporal motion of objects occurring at historical timestamps can lead to more accurate BEV feature learning.

Specifically, given the current timestamp t, we generate a pseudo BEV feature of timestamp t-k from its adjacent frames and utilize this feature to predict the object set at t-k. First, we elaborately design short-term and long-term temporal decoders, which can generate the pseudo BEV feature for timestamp t-k without the involvement of its corresponding camera images. Thanks to the marginal temporal difference between two adjacent frames, the short-term temporal decoder process two adjacent frames of timestamp t-k to provide spatial semantics of objects. The long-term decoder captures long-term motion of the whole frame sequence and contributes to better object motion estimation, which is complementary to spatial localization of the short-term branch. Subsequently, an additional object decoder is flexibly attached to predict the object targets using the generated pseudo BEV feature. Note that we only perform HoP during training, thus the proposed method does not introduce extra overheads during inference. As a plug-and-play approach, HoP can be flexibly incorporated into the state-of-the-art BEV detection frameworks, including BEVFormer [15] and BEVDet series [8, 7, 14]. Furthermore, the auxiliary HoP approach is complementary to prevalent temporal modeling methods, leading to significant gains.

Extensive experiments are conducted to evaluate the effectiveness of the proposed HoP on the nuScenes dataset [1]. We choose the representative methods, including BEVFormer and BEVDet4D-Depth [7, 14] to evaluate our method. To be specific, we obtain 55.8% NDS with ResNet-101-DCN and 60.3% NDS with VoVNet-99 when evaluating HoP on nuScenes *val*. Surprisingly, HoP achieves 68.5% NDS and 62.4% mAP with ViT-L [4] on nuScenes *test*, surpassing all the 3D object detectors on the leaderboard by a large margin.

In conclusion, our contributions can be summarized as follows:

- We propose a novel temporal enhanced training scheme, namely Historical Object Prediction (HoP), to encourage more accurate BEV feature learning. HoP can force the model to capture the spatial semantics and temporal motion of objects in the historical frame during training.
- We design a temporal decoder that consists of a short-term decoder and a long-term decoder to provide reliable spatial localization and accurate motion estimation of objects.
- We equip the competitive 3D object detector baselines with our approach and yield significant improvements on the nuScenes dataset. Surprisingly, HoP with ViT-L achieves 68.5% NDS and 62.4% mAP on nuScenes *test*, establishing the new state-of-the-art performance.

2. Related Works

2.1. Multi-view 3D Object Detection

Modern multi-view methods for 3D object detection could be mainly categorized into two branches, LSS-based [25] and query-based methods. We will introduce how these two methods differ in feature aggregation and leave the temporal modeling part in the next section.

BEVDet [8] is a representative method in LSS-based methods. Following the Lift-Splat-Shoot paradigm, the method first explicitly estimates depth for every image pixel, then lifts the 2D features to 3D voxels according to the depth, and finally splats 3D features to BEV features and conducts object detection on it. BEVDepth [14] further improves the view transformation module with explicit depth supervision.

Query-based methods typically employ learnable queries to aggregate 2D image features by attention [27, 37] mechanism. DETR3D [30] utilizes object queries for predicting 3D positions and projects them back to 2D coordinates to obtain the corresponding features. Graph-DETR3D [2] and Sparse4D [17] enhance it respectively with a learnable 3D graph and sparse 4D sampling. PETR [18] encodes 3D position into image features and therefore directly query with global 2D features. BEVFormer [15] leverages grid-shaped BEV queries to interact with 2D features by deformable attention; on its basis, BEVFormerv2 [32] introduces perspective supervision. Besides, PolarFormer and Ego3RT [22, 9] advocates modeling BEV queries in polar coordinates to fit the real-world scenario.

2.2. Temporal Modeling for Multi-view 3D Object Detection

The motion information in autonomous driving scenes has been increasingly explored to utilize the temporal cues for improving detection performance in recent 3D perception frameworks. BEVFormer [15] proposes the temporal self-attention mechanism to dynamically fuse the previous BEV features by deformable attention [37] in an RNN manner. BEVDet4D [7] introduces the temporal modeling to lift BEVDet [8] to spatial-temporal 4D space. The 3D position embedding (3D PE) in PETR [18] is extended by PETRv2 [19] with the temporal alignment. BEVStereo [12] and STS [31] both leverage temporal views for constructing multi-view stereo by an effective temporal stereo method. SOLOFusion [24] fully exploits the synergy of short-term and long-term temporal information that is highly comple-

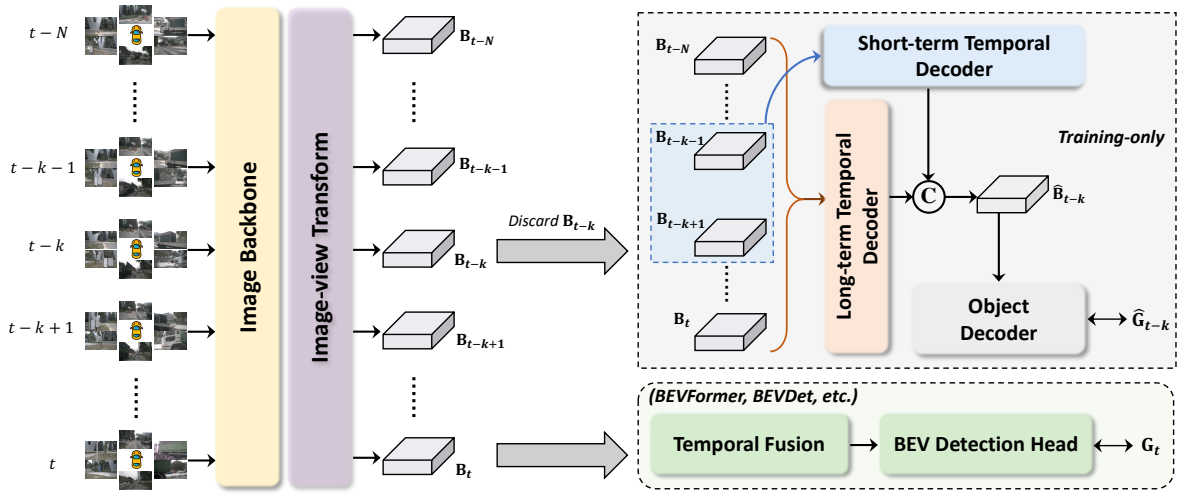

Figure 2: **Framework of our Historical Object Prediction (HoP).** The auxiliary branches are discarded during evaluation. The symbol © denotes feature concatenation and fusion.

mentary as well as achieving state-of-the-art performance. BEVFormerV2 [32] uses the bidirectional temporal encoder to utilize both history and future BEV features, improving the performance by a large margin. In contrast with aforementioned works, we provide a novel temporal enhanced training paradigm to improve the temporal modeling without additional inference costs.

3. Method

3.1. Overall Architecture

As illustrated in Figure 2, HoP can be easily incorporated into many multi-view 3D object detectors, *e.g.*, BEVFormer and BEVDet4D-Depth. We transform the extrinsic parameters of previous frames to the coordinate system of the current ego to perform temporal alignment following BEVDet.

3.2. Historical Object Prediction

Pipeline. The architecture of our Historical Object Prediction (HoP) is composed of a temporal decoder $\mathcal{T}$ and an object decoder $\hat{\mathcal{D}}$. Given the BEV feature sequence $\{\mathbf{B}_{t-N}, \cdots, \mathbf{B}_t\}$ that consists of N historical BEV features and the current BEV feature, we denote the corresponding 3D ground truths as $\{\mathbf{G}_{t-N}, \cdots, \mathbf{G}_t\}$. If the BEV feature at timestamp t-k is discarded, we use the remaining BEV feature sequence $\{\mathbf{B}_{t-N}, \cdots, \mathbf{B}_t\}$ - $\{\mathbf{B}_{t-k}\}$ (denoted as $\mathbf{B}^{rem}$) to predict the 3D objects at timestamp t-k. To be specific, we adopt $\mathcal{T}$ and $\hat{\mathcal{D}}$ to transform the remaining BEV features to obtain the 3D predictions $\hat{\mathbf{P}}_{t-k}$:

$$\hat{\mathbf{P}}_{t-k} = \hat{\mathcal{D}}(\mathcal{T}(\{\mathbf{B}_{t-N}, \cdots, \mathbf{B}_t\} - \{\mathbf{B}_{t-k}\})). \quad (1)$$

The temporal decoder $\mathcal{T}$ that models both short-term and long-term information is developed to reconstruct the BEV feature at timestamp t-k. The object decoder $\hat{\mathcal{D}}$ is attached to generate the predictions $\hat{\mathbf{P}}_{t-k}$ using the pseudo BEV feature. Considering the ego-motion and temporal alignment, we transform 3D coordinates of $\mathbf{G}_{t-k}$ into the coordinate system of current frame t (denoted as $\hat{\mathbf{G}}_{t-k}$). The final training objective is to minimize the differences between $\hat{\mathbf{P}}_{t-k}$ and $\hat{\mathbf{G}}_{t-k}$.

The input to the temporal decoder is the remaining set of BEV features consisting of (i) short-term BEV features $\{\mathbf{B}_{t-k-1}, \mathbf{B}_{t-k+1}\}$, and (ii) long-term BEV features $\{\mathbf{B}_{t-N}, \cdots, \mathbf{B}_t\}$ - $\{\mathbf{B}_{t-k}\}$. We first add temporal positional embeddings to BEV features in this full set. Our temporal decoder reconstructs the feature $\hat{\mathbf{B}}_{t-k}$ by utilizing the spatial semantics and temporal cues of the BEV feature sequence. More specifically, we present a temporal decoder that consists of two separate branches with their special expertise to capture short-term and long-term information. The learned strong spatial-temporal representation can help the network better estimate the object location in frame at timestamp t-k.

Short-term temporal decoder. Thanks to the high temporal correlation between adjacent frames, the short-term temporal decoder only operates on the adjacent BEV feature set $\{\mathbf{B}_{t-k-1}, \mathbf{B}_{t-k+1}\}$ (denoted as $\mathbf{B}^{adj}$), building a detailed spatial representation in BEV space. We first define a grid-shaped learnable short-term BEV queries $\mathbf{Q}^{short}_{t-k} \in \mathbb{R}^{H \times W \times C}$, where H and W refer to the spatial shape of the BEV plane. The short-term BEV queries aggregate the spatial information and model the short-term motion from $\mathbf{B}_{t-k-1}$ and $\mathbf{B}_{t-k+1}$ through the short-term temporal attention, which can be formulated as:

$$\hat{\mathbf{B}}^{short}_{t-k} = \sum_{\mathbf{V} \in \mathbf{B}^{adj}} \text{DeformAttn}(\mathbf{Q}^{short}_{t-k,p}, p, \mathbf{V}), \quad (2)$$

where p is the spatial index in the BEV plane, $\mathrm{DeformAttn}(q, p, x)$ refers to the deformable attention [37] with query q, reference point p and input features x. We further feed $\hat{\mathbf{B}}_{t-k}^{short}$ into a feed-forward network [27] and obtain the output of this short-term branch. However, it is still challenging to precisely construct the temporal relations of the same objects between $\mathbf{B}_{t-k}$ and other BEV features with only two adjacent frames.

Long-term temporal decoder. The long-term temporal processes the whole remaining BEV set to perceive the motion clues over long-term history, which increases the localization potential [24] and contributes to more accurate localization. Therefore, these two branches in the temporal decoder are complementary to each other. As for long-term motion, it is intuitive that we only focus on the *spatial motion* of the same objects in the bird's eye view, ignoring the height information of objects. For most 3D object detectors with 2D BEV features, *e.g.*, BEVFormer and BEVDet, the height information has been flattened to the feature embeddings. Accordingly, we first employ a channel reduction operation to the input set $\mathbf{B}^{rem}$ to prune the height information and achieve better training efficiency. Given the learnable long-term BEV queries $\mathbf{Q}_{t-k}^{long} \in \mathbb{R}^{H\times W\times C/r}$ and reduction layer with parameter $\mathbf{W}^r \in \mathbb{R}^{C\times C/r}$, we can capture the long-term dependencies as:

$$\hat{\mathbf{B}}_{t-k}^{long} = \sum_{\mathbf{V}\in\mathbf{B}^{rem}} \mathrm{DeformAttn}(\mathbf{Q}_{t-k,p}^{long}, p, \mathbf{V}\mathbf{W}^r), \quad (3)$$

where r denotes the reduction ratio and is set as 4 by default. After the feed-forward network, the long-term decoder output the BEV features. Finally, we concatenate the short-term and long-term BEV features and perform feature fusion by 3×3 convolution.

Object decoder. The reconstructed BEV feature $\hat{\mathbf{B}}_{t-k}$ is further processed by a lightweight object decoder. The decoder is used to generate 3D predictions $\hat{\mathbf{P}}_{t-k}$ upon the BEV feature and thus its implementation is flexible. Variants of BEV detection heads are an obvious choice and we will consider multiple instantiations in our experiments. After obtaining $\hat{\mathbf{P}}_{t-k}$, we should align the coordinates between learning targets $\mathbf{G}_{t-k}$ and the final predictions $\hat{\mathbf{P}}_{t-k}$. For clarity, we first denote the ego coordinate as $e(t)$ at frame t. In our implementation, we transform the extrinsic parameters of previous frames to the coordinate system of the current ego, thus BEV features and predictions at different timestamps share the same $e(t)$. To simplify the learning targets, 3D coordinates of $\mathbf{G}_{t-k}$ should be transformed into $e(t)$. Considering the ego motion, we convert $\mathbf{G}_{t-k}$ to $\hat{\mathbf{G}}_{t-k}$ by ego transformation matrix as:

$$\hat{\mathbf{G}}_{t-k} = \mathbf{T}_{e(t-k)}^{e(t)}\mathbf{G}_{t-k}, \quad (4)$$

where $\mathbf{T}_{e(t-k)}^{e(t)}$ is the transformation matrix from the source coordinate system $e(t-k)$ to the target coordinate system $e(t)$.

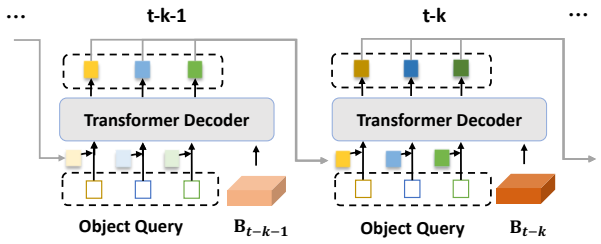

Figure 3: Illustration of the temporal multi-stage decoder. Object queries are enhanced by their counterparts derived from the last frame, forming a recurrent connection.

Discussion. The act of discarding a frame in HoP might share similarity with masked image modeling works, like MAE [5]. However, there are several points where HoP differs greatly from those works. First, HoP is not a self-supervised learning method as it requires 3D ground-truth boxes for training. HoP leverages short-term and long-term branches to extract diverse information from objects with different velocities to predict 3D targets in the discarded frame. In contrast, masked image modeling methods generally choose pixels or features as reconstruction targets, but predicting features fails to bring significant gains as HoP in Table 6c. Moreover, MAE is adopted for backbone pretraining. HoP is designed for 3D BEV detection to enhance temporal learning by predicting objects in the discarded frame. In fact, HoP is compatible with MAE since we can use a MAE pretrained ViT as the backbone.

3.3. Historical Temporal Query Fusion

Apart from the HoP method that exploits temporal information in BEV feature level, we also explore query-level temporal modeling in query-based methods, *e.g.*, BEVFormer, by fusing historical object queries into current queries. This fusion step provides current queries with an initialization of history perception, forming a refining process.

The BEV feature sequence $\{\mathbf{B}_{t-N}, \cdots, \mathbf{B}_t\}$ is inferred in the time order. For the timestamp t-k, we perform detection with the pre-defined object queries $\mathbf{O}$ as follows:

$$\bar{\mathbf{O}}_{t-k} = \mathcal{D}(\mathbf{B}_{t-k}, \mathbf{O}), \quad (5)$$

where $\mathcal{D}$ is the BEV decoder and $\bar{\mathbf{O}}_{t-k}$ is its output.

Each object query learns to specialize in certain areas in the BEV plane, so the output queries contain rich semantic information from the regions. When detecting on the current frame, we refine these output historical queries instead of starting from the pre-defined ones to make the learning process easier. For example, if an object query detects a still object in the last frame, its historical counterpart could help

Method	Backbone	Query-based	LiDAR	Epoch	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
BEVDepth [14]	R101-DCN‡		✓	90†	0.538	0.418	-	-	-	-	-
UVTR [13]	R101-DCN‡	✓		24	0.483	0.379	0.731	0.267	0.350	0.510	0.200
BEVFormer [15]	R101-DCN‡	✓		24	0.517	0.416	0.673	0.274	0.372	0.394	0.198
PETRv2 [19]	R101-DCN‡	✓		24	0.524	0.421	0.681	0.267	0.357	0.377	0.186
PolarFormer [9]	R101-DCN‡	✓		24	0.528	0.432	0.648	0.270	0.348	0.409	0.201
Sparse4D [17]	R101-DCN‡	✓		24	0.541	0.436	0.633	0.279	0.363	**0.317**	**0.177**
HoP-BEVFormer	R101-DCN‡	✓		24	**0.558**	**0.454**	**0.565**	**0.265**	**0.327**	0.337	0.194

Table 1: Comparison of recent works on the nuScenes val set. † indicates methods with CBGS which will elongate 1 epoch into 4.5 epochs. ‡ notes the backbone is initialized from FCOS3D [28] backbone. LiDAR: trained with LiDAR supervision.

it quickly locate the object without searching on its area in the BEV plane again. On the other hand, historical information from a moving object also contributes to localizing and detecting its velocity.

As shown in Figure 3, to infer on the BEV feature $\mathbf{B}_{t-k}$, we first collect historical object queries $\mathbf{O}_{t-k}^{his}$ as follows:

$$\mathbf{O}_{t-k}^{his} = \mathrm{MLP}(\bar{\mathbf{O}}_{t-k-1}), \tag{6}$$

where MLP is a multi-layer perceptron with two linear layers and an expansion ratio of 4, which is used to fit the history query into the current timestamp. Note that we only use historical queries from the last timestamp because we hypothesize that queries from more distant history possess little information overlap with currently visible objects.

Then we merge them with the pre-defined object queries $\mathbf{O}$ as Equation 7,

$$\bar{\mathbf{O}}_{t-k} = \mathcal{D}(\mathbf{B}_{t-k}, \mathbf{O} + \mathbf{O}_{t-k}^{his}). \tag{7}$$

Considering the first shown-up objects in the current frame, the merge process takes both historical queries and pre-defined queries into account.

4. Experiments

4.1. Dataset and Metrics

We validate our method on the popular nuScenes[1] dataset. The nuScenes dataset contains 1000 driving scenes in total, which are split into 700, 150, and 150 respectively for training, validation, and testing. Each scene lasts for 20 seconds and annotations of 3D boxes are supplied every 0.5s in the keyframe. There are 6 cameras surrounding the vehicle, providing images covering the whole 360-degree FOV. As for evaluation, we adopt the standard metrics for nuScenes detection task. The final nuScenes detection score (NDS) is a weighted sum of mean Average Precision (mAP) and five True Positive metrics including Average Translation Error (ATE), Average Scale Error (ASE), Average Orientation Error (AOE), Average Velocity Error (AVE), and Average Attribute Error (AAE).

4.2. Implementation Details

We mainly implement our method on the optimized version of BEVFormer [15] (denoted as BEVFormer-opt), where we generate the current BEV feature with resolution 200×200 by 6 BEV spatial encoder layers and fuse the current and 4 previous BEV features by a BEV temporal encoder. All models are trained with a mini-batch of 8, an initial learning rate of 0.0002, and a weight decay of 0.01 for 24 epochs without CBGS [36]. Unless otherwise specified, we use ResNet-50 [6] pre-trained on COCO [16] as the image backbone, and the image size is processed to 1408×512. Both image and BEV augmentation are adopted following BEVDet [8]. Besides, we also adopt the BEVDet4D-Depth [7] as baseline models and follow the original settings.

4.3. Main Results

To compare with previous state-of-the-art 3D object detection methods, we adopt ResNet101-DCN [6, 3] initialized from FCOS3D [28] and VoVNet-99 [10, 11] initialized from DD3D [23] checkpoint. We also enlarge the input resolution to 1600×640 and use 1500 object queries. The historical temporal query fusion is employed in Table 1 and 2.

We report the results of our HoP with the ResNet101-DCN backbone on nuScenes *val* in Table 1. Surprisingly, our method outperforms BEVDepth with CBGS strategy and depth supervision from LIDAR over +2.0% NDS and +3.6% mAP under comparable complexity levels. Besides, HoP achieves the best performance compared with other competitive query-based object detectors, *e.g.*, PETRv2 [19] and PolarFormer [9]. We also compared HoP with other state-of-the-art algorithms on nuScenes *test*, and still obtain the best performance of 61.2% NDS and 52.8% mAP.

To show the scalability of our model to stronger backbones, we apply HoP to BEVDet4D-Depth with ViT-L [4] as the backbone. The ViT-L we employ is initialized with the Co-DETR [38] pre-trained on the 2D detection bench-

Method	Backbone	Query-based	LiDAR	Epoch	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
BEVDepth* [14]	V2-99		✓	90†	0.600	0.503	**0.445**	0.245	0.378	0.320	0.126
DETR3D [30]	V2-99	✓		24	0.479	0.412	0.641	0.255	0.394	0.845	0.133
UVTR [13]	V2-99	✓		24	0.551	0.472	0.577	0.253	0.391	0.508	0.123
BEVFormer [15]	V2-99	✓		24	0.569	0.481	0.582	0.256	0.375	0.378	0.126
PETRv2 [19]	V2-99	✓		24	0.582	0.490	0.561	0.243	0.361	0.343	0.120
Sparse4D [17]	V2-99	✓		24	0.595	0.511	0.533	0.263	0.369	**0.317**	0.124
HoP-BEVFormer	V2-99	✓		24	0.603	0.517	0.501	0.245	0.346	0.362	**0.105**
HoP-BEVFormer†	V2-99	✓		24	**0.612**	**0.528**	0.491	**0.242**	**0.332**	0.343	0.109

Table 2: Comparison of recent works on the nuScenes test set. $*$ indicates test-time augmentation. † means we do not detach 1 historical frame. VoVNet-99 (V2-99) [10] was pre-trained on the depth estimation task with extra data [23].

Method	Backbone	TTA	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
SOLOFusion [24]	ConvNeXt-B [21]		0.619	0.540	0.453	0.257	0.376	0.276	0.148
BEVFormerV2 [32]	InternImage-XL [29]		0.648	0.580	0.448	0.262	**0.342**	0.238	0.128
BEVDet-Gamma [7]	Swin-B [20]	✓	0.664	0.586	0.375	**0.243**	0.377	0.174	**0.123**
HoP-BEVDet4D-Depth	ViT-L [4]		**0.685**	**0.624**	**0.367**	0.249	0.354	**0.171**	0.131

Table 3: Comparisons on the nuScenes camera-only 3D detection leaderboard. TTA denotes test-time augmentation.

mark Objects365 [26]. We use 8 frames into the future to provide additional information from the future. To reduce the training time, we first train the ViT-L using FCOS3D on the nuScenes *trainval* for 36 epochs. Then we only train HoP-BEVDet4D-Depth for 5 epochs with CBGS. Without test-time augmentation, HoP pushes the current best result to a new record of 68.5% NDS and 62.4% mAP on nuScenes *test*.

4.4. Ablation Studies

Unless stated otherwise, we conduct the ablation experiments on BEVFormer-opt with a ResNet-50 backbone. We follow the original settings when adopting BEVDet4D as the baseline.

Effectiveness of each component. We perform a component-wise ablation to thoroughly analyze the effect of each component in Table 4. Surprisingly, incorporating the historical temporal prediction into the optimized BEVFormer brings significant performance gains (+1.8% NDS, +1.4% mAP) over the baseline. The overall improvements hold when we apply the training strategy to BEVDet4D-Depth, achieving +1.6% NDS and +1.4% mAP improvements, respectively. Alternatively, the temporal information provided by historical queries also contributes to the performance improvements for BEVFormer, where a +1.2% NDS gain is observed. Without detaching the gradients of historical BEV features in the last two layers of BEV encoder, the model can be further improved to 53.9% NDS. These results prove the effectiveness and generalization of our HoP.

Optimized BEVFormer baseline. Table 5 illustrates the critical designs of our optimized BEVFormer baseline. We can observe setting the dropout rate within the transformer as 0 significantly improves the performance and look forward twice [34] is an essential factor to improve mAP. Besides, we replace the heavy backbone with the ResNet50 and decrease the input resolution to 1408×512 to improve the training efficiency of BEVFormer. At the same time, we also introduce step learning rate decay and data augmentation, which includes BEV augmentation, random flipping, cropping, and rotation, to achieve better performance without inference costs increase. Decoupling the BEV temporal encoder from the spatial encoder and using 4 previous frames, is crucial in the later promotion process. Combining these aforementioned modifications on BEVFormer results in our optimized baseline BEVFormer-opt. Finally, we achieve comparable performance with the original BEVFormer-R101-DCN while running much faster and requiring fewer parameters.

Temporal decoder design. Table 6a illustrates the key designs of our temporal decoder for prediction targets prediction. The performance is always improved when using the structure of one individual decoder alone, reaching 52.9% NDS and 52.7% NDS with the short-term and the long-term decoder, respectively. We achieve the best performance with 53.1% NDS and 42.8% mAP when both short-term and long-term decoders are adopted, demonstrating the complementarity between these two branches.

Object decoder design. The object decoder detects the 3D

Method	HoP	HQ Fusion	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
BEVFormer-opt			0.513	0.414	0.655	0.274	0.447	0.368	0.194
BEVFormer-opt		✓	0.525	0.425	0.660	0.274	0.395	0.357	0.188
BEVFormer-opt	✓		0.531	0.428	0.638	0.269	0.352	0.369	0.185
BEVFormer-opt†	✓		0.539	0.435	0.629	0.268	**0.342**	0.360	**0.184**
BEVFormer-opt†	✓	✓	**0.544**	**0.439**	**0.607**	**0.265**	0.354	**0.340**	0.193
BEVDet4D-Depth			0.493	0.385	0.632	0.283	0.581	0.289	0.212
BEVDet4D-Depth	✓		**0.509**	**0.399**	**0.608**	**0.272**	**0.541**	**0.281**	**0.205**

Table 4: Component-wise ablations. † indicates we do not detach historical BEV features of last two layers in the BEV encoder. "HQ Fusion" refers to historical temporal query fusion.

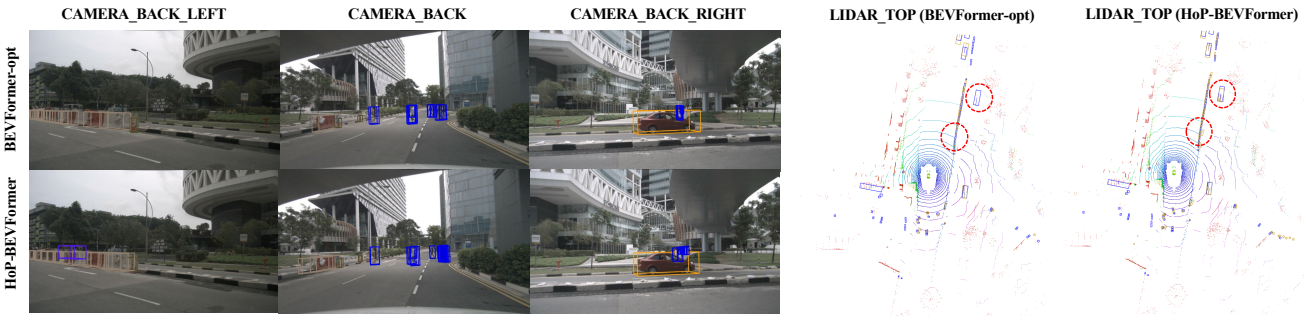

Figure 4: Visualization result of BEVFormer-opt and HoP-BEVFormer on nuScenes val set. The predicted boxes are marked in yellow, while the ground truth boxes are marked in blue in the LIDAR_TOP figure.

Method	NDS	mAP	#params (M)	Training time (sec/iter)
BEVFormer-R101-DCN	0.517	0.416	78.6	2.7
+ Dropout 0	0.529	0.424	78.6	2.7
+ Look forward twice	0.531	0.430	78.6	2.7
+ Smaller backbone R50	0.487	0.388	57.7	2.0
+ Step LR decay	0.493	0.391	57.7	2.0
+ Data augmentation	0.499	0.404	57.7	2.0
+ Decoupled BEV encoder	0.521	0.420	55.5	2.2
+ Smaller resolution	0.513	0.414	55.5	1.6
BEVFormer-opt-R50	**0.513**	**0.414**	**55.5**	**1.6**

Table 5: Step-by-step optimizing the BEVFormer baseline.

targets with the reconstructed features, thus it can be flexibly designed, as studied in Table 6b. We find these detection heads can bring consistent gains. For example, ATSS [35] is extended to 3D object detection and brings 1.6% NDS and 1.2% mAP gains. We choose CenterPoint [33], which achieves the best performance as the default object decoder.

Prediction target. We also study the design that only predicts pseudo BEV features at timestamp t-k supervised by $\mathbf{B}_{t-k}$ generated by the BEV encoder in a self-distillation manner. As presented in Table 6c, the feature supervision signals only slightly improve the baseline while the 3D objects supervision boosts the performance by a large margin. In a summary, explicit supervision from 3D objects in the BEV space is crucial to HoP.

Choice of prediction index k. The selection of prediction index k is a critical design of our method and is ablated in Table 6d. We first study whether the model can benefit from predicting the 3D objects in the future frame, *e.g.*, frame t+1 ($k = -1$). Interestingly, this future 3D object prediction also brings non-trivial gains but performs inferior to history prediction. The relatively weaker gains can be explained by the difference between a history prediction task and a future prediction task: it is much more challenging to predict 3D objects in the future frame at timestamp t+1 with only history frames from timestamp t-N to t. Predicting objects in frame t-k with both history frames from timestamp t-N to t-k-1 and future frames from timestamp t-k+1 to t is much easier. A challenging auxiliary task can disturb the learning progress and slightly decrease the performance gains. When predicting the objects of history frame t-k, we do not detach the feature maps of frame t-k+1 to allow the gradients back propagate through the both short-term and long-term temporal decoder. Therefore, we observe the method imposes a significant increase in GPU memory for $k > 1$ since the features at timestamp t and t-k+1 are not detached. Accordingly, we choose k as 1 since it achieves the best trade-offs between accuracy and training costs.

Training time. We present the training speed (seconds per training iteration) under different settings in Table 6e. This training speed is evaluated with 8 NVIDIA A100 GPUs. Compared with the baseline, HoP introduces an additional temporal decoder and object decoder during training, leading to an inevitable increase in training time. Thanks to the lightweight design of our temporal decoder and object decoder, we only observe a 20% increase (from 1.6 seconds to 1.9 seconds) in training time for BEVFormer baseline.

Case	NDS	mAP
None	0.513	0.414
Short-term	0.528	0.426
Long-term	0.526	0.425
Both	**0.531**	**0.428**

(a) **Temporal decoder design**. Both short-term and long-term can improve accuracy.

Case	NDS	mAP
None	0.513	0.414
Transformer	0.521	0.421
ATSS	0.529	0.426
CenterPoint	**0.531**	**0.428**

(b) **Object decoder design**. The decoder can be flexibly implemented.

Case	NDS	mAP
None	0.513	0.414
BEV feature	0.518	0.418
3D objects	**0.531**	**0.428**

(c) **Prediction target**. The explicit supervisions of 3D targets are critical.

k	NDS	mAP	Memory
None	0.513	0.414	1×
-1	0.523	0.420	1.6×
1	0.531	0.428	1.3×
2	0.534	0.430	**2.3**×
3	**0.536**	**0.432**	**2.3**×

(d) **Prediction index**. Predicting the objects in frame t-1 achieves the best trade-offs.

Case	NDS	Speed
BEVFormer-opt	0.513	**1.6**
HoP-BEVFormer-opt	0.531	1.9
HoP-BEVFormer-opt†	**0.539**	2.1
BEVDet4D-Depth	0.493	4.5
HoP-BEVDet4D-Depth	0.509	4.7

(e) **Training speed**. HoP slightly increases the training time.

Connection form	NDS	mAP
None	0.513	0.414
Recurrent	**0.525**	**0.425**
Fully-Connected	0.521	0.422
Dense	0.523	0.421

(f) **Historical query collection form**. All three connection forms bring improvements.

Table 6: **Ablation experiments** with ResNet-50 on nuScenes *val*. Default settings are marked in gray. † indicates we do not detach historical BEV features of last two layers in the BEV encoder.

Figure 5: Visualization results of object decoder with three variants of the temporal decoder.

When the gradients of historical BEV features are not detached in the last two layers of BEV encoder, the training time is increased by 30%. As for BEVDet4D-Depth, the BEV feature resolution and dimension is smaller than BEVFormer. Therefore, the additional training costs brought by HoP are *negligible* (from 4.5 seconds to 4.7 seconds).

Historical query collection form. We explore different methods to obtain historical object queries $\mathbf{O}_{t-k}^{his}$ in Table 6f. The Recurrent method only utilizes historical queries from the last frame, which is described in Equation 6; the Fully-Connected method collects all the historical object queries $\{\bar{\mathbf{O}}_{t-N}, ..., \bar{\mathbf{O}}_{t-1}\}$ but only uses them for the current frame t; the Dense method extends the Fully-Connected method with using all historical queries on every timestamp. These three variants can bring consistent improvement, proving the insensitivity of the collection form; the recurrent form obtains the best result, where mAP increased by 1.2% and NDS increased by 1.3%. The results also prove our hypothesis that historical queries from the previous frame provide a more reliable initialization.

4.5. Visualization

Figure 4 shows the detection result of the baseline method BEVFormer-opt and our proposed HoP-BEVFormer. The predictions are marked in yellow, while the ground truth boxes are marked in blue in the LIDAR_TOP figure. We observe that HoP-BEVFormer successfully detects small objects or occluded objects thanks to the superior ability to aggregate temporal information.

We also visualize the predictions of the object decoder with three variants of temporal decoder $\mathcal{T}$: only long-term, only short-term, and both of them in Figure 5. The decoder with only a short-term branch can detect accurate attributes of objects, *e.g.*, height and length, but fails to localize them precisely. Besides, results in the second row show that the short-term branch contributes to detecting more foreground objects because of its detailed spatial semantics. On the contrary, the decoder with only long-term modeling performs precise localization due to the long-term motion while struggling to detect the size of the object, *e.g.*, height.

When both long-term and short-term branches are adopted, we achieve the best result by combining their advantages.

5. Conclusion

In this paper, we propose a new paradigm, named Historical Object Prediction (HoP) for multi-view 3D detection to leverage temporal information more effectively. The HoP approach is straightforward: given the current timestamp t, we generate a pseudo BEV feature of timestamp t-k from its adjacent frames and utilize this feature to predict the object set at timestamp t-k. First, we elaborately design short-term and long-term temporal decoders, which can generate the pseudo BEV feature for timestamp t-k without the involvement of its corresponding camera images. Second, an additional object decoder is flexibly attached to predict the object targets using the generated pseudo BEV feature. As a plug-and-play approach, HoP can be easily incorporated into state-of-the-art BEV detection frameworks, including BEVFormer and BEVDet series. Extensive experiments are conducted to evaluate the effectiveness of the proposed HoP on the nuScenes dataset. Surprisingly, HoP achieves 68.5% NDS and 62.4% mAP on nuScenes test, outperforming all the 3D object detectors on the leaderboard by a significant margin.

Acknowledgement

This project is funded in part by National Key RD Program of China Project 2022ZD0161100, by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Commission (ITC)'s InnoHK, by General Research Fund of Hong Kong RGC Project 14204021. Hongsheng Li is a PI of CPII under the InnoHK.

References

[1] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[2] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Graph-detr3d: rethinking overlapping regions for multi-view 3d object detection. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5999–6008, 2022.

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[5] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16000–16009, June 2022.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

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

[8] 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.

[9] 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. *arXiv preprint arXiv:2206.15398*, 2022.

[10] Youngwan Lee, Joong-won Hwang, Sangrok Lee, Yuseok Bae, and Jongyoul Park. An energy and gpu-computation efficient backbone network for real-time object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019.

[11] Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13906–13915, 2020.

[12] 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. *arXiv preprint arXiv:2209.10248*, 2022.

[13] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. *arXiv preprint arXiv:2206.00630*, 2022.

[14] 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. *arXiv preprint arXiv:2206.10092*, 2022.

[15] 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 *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IX*, pages 1–18. Springer, 2022.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[17] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d: Multi-view 3d object detec-

tion with sparse spatial-temporal fusion. *arXiv preprint arXiv:2211.10581*, 2022.

[18] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVII*, pages 531–548. Springer, 2022.

[19] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Qi Gao, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petrv2: A unified framework for 3d perception from multi-camera images. *arXiv preprint arXiv:2206.01256*, 2022.

[20] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[21] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[22] Jiachen Lu, Zheyuan Zhou, Xiatian Zhu, Hang Xu, and Li Zhang. Learning ego 3d representation as ray tracing. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, pages 129–144. Springer, 2022.

[23] 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.

[24] 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. *arXiv preprint arXiv:2210.02443*, 2022.

[25] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 194–210. Springer, 2020.

[26] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019.

[27] 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.

[28] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 913–922, 2021.

[29] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. *arXiv preprint arXiv:2211.05778*, 2022.

[30] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2022.

[31] 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. *arXiv preprint arXiv:2208.10145*, 2022.

[32] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. *arXiv preprint arXiv:2211.10439*, 2022.

[33] 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*, pages 11784–11793, 2021.

[34] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations*, 2022.

[35] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020.

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

[37] 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.

[38] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. *arXiv preprint arXiv:2211.12860*, 2022.