2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Learning Temporal Cues by Predicting Objects Move for Multi-camera 3D Object Detection

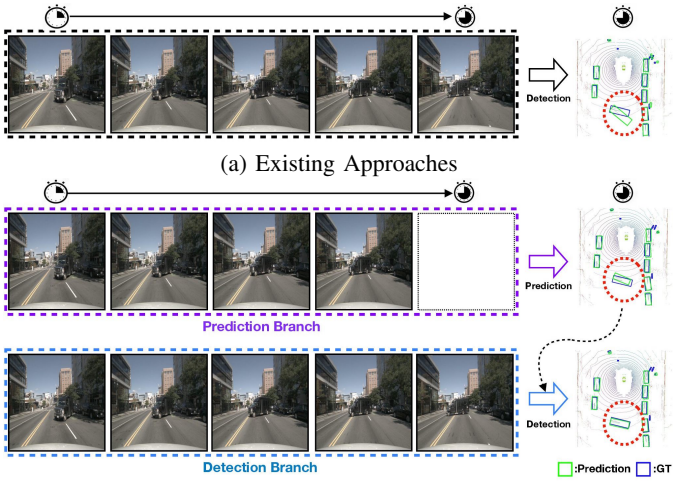
Seokha Moon, Hongbeen Park, Jaekoo Lee*, and Jinkyu Kim*

***Abstract*— In autonomous driving and robotics, there is a growing interest in utilizing short-term historical data to enhance multi-camera 3D object detection, leveraging the continuous and correlated nature of input video streams. Recent work has focused on spatially aligning BEV-based features over timesteps. However, this is often limited as its gain does not scale well with long-term past observations. To address this, we advocate for supervising a model to predict objects' poses given past observations, thus explicitly guiding to learn objects' temporal cues. To this end, we propose a model called DAP (Detection After Prediction), consisting of a two-branch network: (i) a branch responsible for forecasting the current objects' poses given past observations and (ii) another branch that detects objects based on the current and past observations. The features predicting the current objects from branch (i) is fused into branch (ii) to transfer predictive knowledge. We conduct extensive experiments with the large-scale nuScenes datasets, and we observe that utilizing such predictive information significantly improves the overall detection performance. Our model can be used plug-and-play, showing consistent performance gain.**

I. INTRODUCTION

Multi-camera 3D object detection is a crucial task for autonomous vehicles to safely navigate based on understanding their surrounding environment. Recent successes [1], [2] suggest that each image can be mapped into a frustum of features, rasterizing such frustums into a bird's eye view (BEV) grid. A task-specific object detection head is then applied to detect all objects over the BEV space. Recently, a large performance gain has been obtained by utilizing past BEV features [3], [4], [5] (i.e., in addition to the current BEV feature, they augment the current and the past BEV features together, thus utilizing temporal cues). However, its gain does not linearly increase with the number of past observations – its gain is often limited with only two consecutive frames, i.e., the current and previous frames. This may be due to (i) misalignments between BEV features and (ii) suboptimally trained networks that struggle to learn object motions' complicated distribution.

In this paper, we want to focus on the second issue by regularizing the model to learn better temporal cues. Specifically, we advocate for supervising a model to predict objects' current poses conditioned on past observations only – a model needs to predict an object's next poses (in the BEV space) based on past observations. We empirically observe that such a regularization and the use of predictive knowledge significantly improves the overall detection performance, potentially owing to better learned temporal cues. See Fig. 1 where we compare ours with existing approaches.



(a) Existing Approaches

(b) Our Proposed Approach

Fig. 1: Unlike prior multi-camera 3D object detection approaches, which utilize the current and past observations to detect objects, our proposed method regularizes the model by predicting objects' current poses from past observations, and such predictive knowledge is then augmented into object detector, enhancing overall object detection performance.

As shown in Fig. 2 (a), our model consists of two main modules: (i) Temporal Context Extraction Module and (ii) Context-fused Detection Module. In (i), a network is trained to predict objects' current poses given past BEV-based observations. We further utilize (1) a spatiotemporal BEV encoder to model temporal information effectively and (2) a multi-resolution feature extractor to learn both local (i.e., fine-grained semantic cues per each object) and global (i.e., behavior encoding with large receptive field) representations. In (ii), 3D objects are detected given the current and past observations, where intermediate features from (i) are fused to transfer predictive information. Note that the module (ii) can be easily replaced by conventional BEV-based multi-camera 3D object detection models, such as BEVDet4D [3] and BEVDepth [6].

To evaluate our proposed approach, we conducted experiments with the publicly available large-scale nuScenes [7] dataset. Our model, which is applied to existing BEVDet4D and BEVDepth models, provides a significant improvement, showing comparable performance with the state-of-the-art approaches. Our ablation studies and qualitative analysis

S. Moon, H. Park, and J. Kim are with the Department of Computer Science and Engineering, Korea University, Seoul, Republic of Korea.

J. Lee is with the College of Computer Science, Kookmin University, Seoul, Republic of Korea

*Co-corresponding authors: J. Lee (jaekoo@kookmin.ac.kr) and J. Kim (jinkyukim@korea.ac.kr)

further confirm that regularizing a model with predictive information indeed improves the overall detection performance (especially for occluded agents and moving agents), encoding better temporal cues. Note that our code will be publicly available upon publication.

II. RELATED WORK

Camera-based Surround View 3D Detection. A fundamental approach, as proposed by LSS [8], involves the "Lift" and "Splat" methods, which project image features into Bird's-Eye-View (BEV) representations. Building upon this concept, BEVDet [1] effectively detects objects using BEV feature. BEVDet4D [3] takes a step further by incorporating information from previous timestamps, demonstrating the valuable contextual information that past data can provide for the present. BEVFormer [2] introduces a novel method that directly transforms image features into BEV representations using BEV query with BEV shape and deformable attention, drawing inspiration from DETR3D [9]. BEVDepth [6] adopts an approach by directly learning the depth distribution from LiDAR data to construct BEV features. In the context of temporal fusion, VideoBEV [5] effectively reduces computational costs while increasing accuracy by transforming the parallel time fusion method into a recursive fusion method. SOLOFusion [10] innovates with short-term and long-term fusion modules that leverage historical data for improved environmental understanding. PETRv2 [11] introduces a novel approach to achieve temporal alignment by aligning the 3D coordinates of historical and current frames, facilitating accurate tracking and detection.

Dense prediction for BEV. In the field of autonomous driving, future predictions are typically assessed from the BEV perspective, as the majority of objects and events occur on the ground plane.LSS [8] proposes a method of identifying the path of an autonomous vehicle by utilizing BEV features extracted from a camera image. TBP-Former [12] proposes a pose-synchronized BEV encoder and spatial-temporal pyramid transformer to accurately map visual features into synchronized BEV space and extract multi-scale features. FIERY [13] uses a conditional variational autoencoder to generate future instance predictions based on previous BEV features, although it models the entire scene in a single latent code. In contrast, HOPE [14] employs latent variables as Gaussian distributions in multi-scale Bird's-Eye-View (BEV) and utilizes aggregators to fuse high-level visual features. It adopts a deep multi-stage encoder-decoder architecture to predict dense occupancy and flow as future motion. UniAD [15] demonstrates that integrating various tasks such as detection, tracking, occupancy, and flow can enhance their individual performance. Furthermore, HoP [16] highlights the capability of temporal BEV features to generate BEV features at distinct time intervals.

III. METHOD

A. BEV-based Multi-camera 3D Object Detection

As shown in Fig. 2, our model starts from standard BEV-based multi-camera 3D object detection approaches, which consist of four main modules: (i) a visual encoder that extracts high-level representations given N multi-view images with a backbone network (e.g., ResNet-50), (ii) a view transformer that transforms image-view features onto the BEV space (with a data-driven dense depth predictor) conditioned on extrinsic matrix $E_n \in \mathbb{R}^{3\times4}$ and intrinsic matrix $I_n \in \mathbb{R}^{3\times3}$ for $n \in \{1,2,\ldots,N\}$, (iii) a BEV encoder that further learns pivotal cues in the BEV space (for better movable objects' scale, orientation, and velocity), and lastly (iv) a task-specific object detection head that detects (movable) objects in the BEV space. Following [1], our model is built upon Lift-Splat-Shoot [8] for a view transformer, Centerpoint [17] for a detection head, and BEVDet [1] for a visual encoder and a BEV encoder.

Learning Temporal Cues with Temporally Aligned BEV Features. Recent successes [3], [2], [5], [16] suggest that utilizing temporally augmented BEV features (i.e., fusing the most recent BEV features of previous frames with the current) would boost the overall detection performance by learning temporal cues. Such temporal inputs have been shown to be effective in dealing with objects' speed variations, occluded objects, or depth distribution prediction. We also follow this stream of work [3], we extract N recent (spatially aligned) BEV features $\mathbf{a}_n$ at the current time t for $n \in \{t, t-1, \ldots, t-N\}$ from the current and past observations. Similar to [3], we align multiple BEV features according to the current pose of an ego-vehicle, i.e., all previous BEV features are transformed (and interpolated) into the current ego-vehicle's coordinate system based on the measured ego-motion. The aligned BEV features are then concatenated and consumed by a BEV encoder, outputting a further high-level BEV feature $\mathbf{B}^o$ as follows:

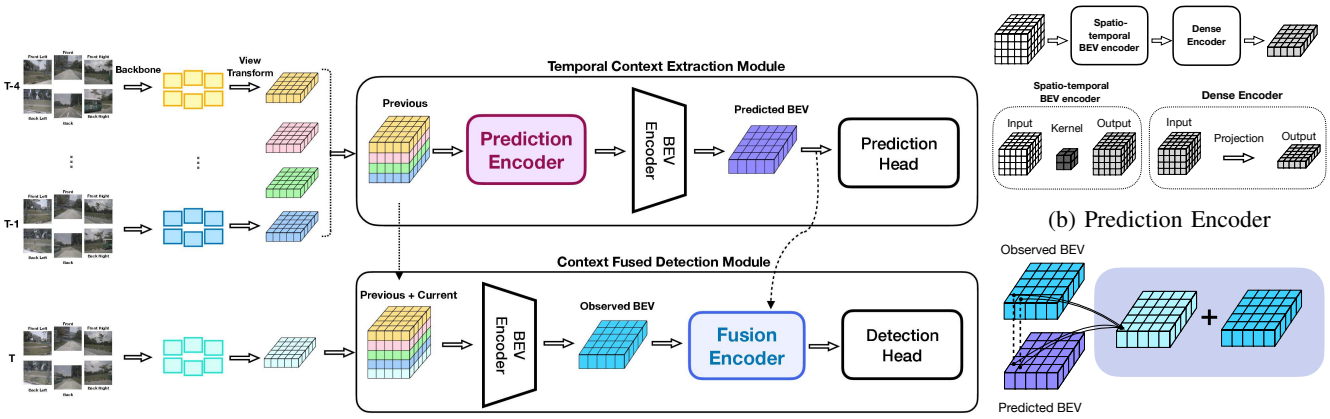
$$\mathbf{B}^o = \texttt{BEVEncoder}_o(\texttt{Concat}\{\mathbf{a}_t, \mathbf{a}_{t-1}, \ldots, \mathbf{a}_{t-N}\}) \quad (1)$$

where $\texttt{BEVEncoder}_o$ represents a BEV encoder. In conventional BEV-based multi-camera 3D object detection approaches, such temporally-augmented feature $\mathbf{B}^o$ is then consumed by an object detection head, detecting object attributes, classes, and bounding boxes.

B. Learning Temporal Cues by Predicting Objects' Current Poses from Past Observations

Utilizing such a concatenated BEV feature is, however, often limited in improving detection performance for moving objects, where a BEV encoder is often suboptimally trained to leverage the most recent few frames (i.e., the current and previous frames) instead of utilizing whole past observations. Thus, to learn object motions' complicated distributions (for further maximizing temporal information usage), we advocate for supervising a model to predict objects' current poses given past observations and utilizing such intermediate features to improve the overall object detection performance.

Spatiotemporal BEV Encoder. Similar to the process to obtain a BEV feature $\mathbf{B}^o$, we use concatenated (spatially aligned) BEV features of the past observations, i.e., $\{a_{t-1}, a_{t-2}, \ldots, a_{t-N}\}$. Note that BEV features are aligned



(a) Overall architecture (b) Prediction Encoder (c) Fusion Encoder

Fig. 2: Our proposed multi-view 3D object detection architecture. Built upon a conventional BEV(Bird's Eye View)-based multi-view object detection model, our model consists of two main modules: (i) Temporal Context Extraction Module, which predicts objects' current poses conditioned on past BEV-based observations. (ii) Context Fused Detection Module, which detects 3D objects in the scene based on the current and past BEV-based observations. Intermediate BEV feature from the Temporal Context Extraction Module is fused into the Context Fused Detection Module through the Fusion Encoder for the final verdict.

with the ego vehicle's position at each respective time step to be aligned with the current time step. To further capture both temporal and spatial information, we concatenate these aligned features and input them into our Spatiotemporal BEV encoder, denoted as $\mathbb{T}$, which utilizes a 3D ConvNet [18] to model temporal information effectively. We empirically observe that utilizing such 3D ConvNets preserves the temporal information better than 2D ConvNets (consistent with existing reports [19]), which often lose their input's temporal signal, under-performing in the objects' future pose prediction task.

Learning Multi-resolution Features. Further, we use a multi-resolution feature learning strategy to learn both local and global representations, i.e., encoding temporal cues for fast-moving vehicles may need larger receptive fields to encode their behaviors (i.e., global), while a model needs to retain fine-grained semantic cues per each object (i.e., local). Thus, we further utilize an additional encoder that maintains high-resolution (or local) representations, coupling the low-resolution (or global) representation in parallel. Specifically, we HRNet [20]-based encoder that takes outputs from our spatiotemporal BEV encoder as an input, producing multi-resolution feature $\mathbf{B}^f$, which (i) will be consumed to predict objects' current poses and (ii) will be augmented into the main branch of object detection. Formally, the multi-resolution feature $\mathbf{B}^f$ is defined as follows:

$$\mathbf{B}^f = \texttt{BEVEncoder}_p(\mathbb{D}(\mathbb{T}(\texttt{Concat}\{\mathbf{a}_{t-1}, \mathbf{a}_{t-2}, \ldots, \mathbf{a}_{t-N}\}))) \tag{2}$$

The term BEVEncoder_p represents the BEVEncoder used in the Temporal Context Extraction Module, while $\mathbb{D}$ denotes the Dense Encoder used in the Prediction Encoder.

C. Fusion Module

The Predicted BEV feature explicitly predicts the positions of agents within the current BEV frame. This provides insights into where attention should be focused to detect agents at this time point. Therefore, in the Fusion Module, as shown in Fig. 2c, our objective is to effectively integrate the Observed BEV features $\mathbf{B}^o$, generated using images from all frames, with the Predicted BEV features $\mathbf{B}^f$. This integration aims to enhance our understanding of the current scene.

$$FDFA(\mathbf{P}_i, (\mathbf{B}^o, \mathbf{B}^f)) = \sum_{h=1}^{H} W_h \left[\sum_{k=1}^{K} \left(\mathbf{A}_{hik} \bar{W}_h \mathbf{B}^o_{(\mathbf{P}_i + \triangle \mathbf{P}_{hik})} + \mathbf{A'}_{hik} \bar{W}_h \mathbf{B}^f_{(\mathbf{P}_i + \triangle \mathbf{P}_{hik})} \right) \right] \tag{3}$$

$$\hat{\mathbf{B}}_{\mathbf{P}_i} = \mathbf{B}^o_{\mathbf{P}_i} + FDFA(\mathbf{P}_i, (\mathbf{B}^o, \mathbf{B}^f)) \tag{4}$$

To fuse the two features, $\mathbf{B}^o$ and $\mathbf{B}^f$, each of which has a shape of $\{H, W, C\}$, we use a deformable attention [21] based module called Fusion DeFormable Attention($FDFA$). We extract K points for each grid cell from H heads. Given a reference point $\mathbf{P}_i$, we ensure that the offset $\triangle \mathbf{P}_{hik}$ in both the $\mathbf{B}^o$ and $\mathbf{B}^f$ aligns consistently, taking into account the described characteristics. Here, $\triangle \mathbf{P}_{hik}$ represents the offset of the k-th point in the h-th head with respect to the reference point $\mathbf{P}_i$. $\mathbf{B}^o(\mathbf{P}_i + \triangle \mathbf{P}_{hik})$ and $\mathbf{B}^f(\mathbf{P}_i + \triangle \mathbf{P}_{hik})$ are obtained by bilinear interpolation from their respective features $\mathbf{B}^o$ and $\mathbf{B}^f$ at the position $(\mathbf{P}_i + \triangle \mathbf{P}_{hik})$. $\mathbf{A}_{hik}$ and $\mathbf{A'}_{hik}$ represent the attention weights of the feature for the k-th point in the h-th head with respect to the reference point $\mathbf{P}_i$, considering both $\mathbf{B}^o$ and $\mathbf{B}^f$. Hence, $\sum_{k=1}^{K} (\mathbf{A}_{hik} + \mathbf{A'}_{hik}) = 1$. $\bar{W}_h$ and W_h denote linear layers. $\hat{\mathbf{B}}$ represents the Fusion BEV feature generated by the output of the Fusion Module.

D. Loss Function

The detection head of the Context Fused Detection Module follows the CenterPoint [17] head. It takes the fused BEV feature $\hat{\mathbf{B}}$ as input and predicts several attributes for each agent, including the center heatmap, box scale, velocity,

TABLE I: Multi-camera 3D object detection performance comparison in terms of NDS (nuScenes Detection Score) and mAP metrics. Our model is applied to existing two approaches, including BEVDet [1] and BEVDepth [6], and shows improved performance in all metrics. Note that we do not apply CBGS (Cross-balanced Grouping and Sampling) technique, and we use four past frames (i.e., $N = 4$). nuScenes [7] validation set is used.

Method	**NDS** ↑	mAP ↑	mATE ↓	mASE ↓	mAOE ↓	mAVE ↓
BEVDet4D [3]	0.454	0.330	0.715	0.290	0.592	0.315
+ Ours	**0.489**	**0.374**	**0.634**	**0.282**	**0.575**	**0.280**
	(3.5%↑)	(4.4%↑)	(8.1%↓)	(0.8%↓)	(1.7%↓)	(3.5%↓)
BEVDepth [6]	0.483	0.362	0.633	0.280	0.577	0.296
+ Ours	**0.507**	**0.389**	**0.575**	**0.276**	**0.550**	**0.271**
	(2.4%↑)	(2.7%↑)	(5.8%↓)	(0.4%↓)	(1.7%↓)	(2.5%↓)

orientation, and the translation from the center heatmap. By minimizing losses for each of these attributes, the model aims to output optimized bounding boxes and their corresponding classes for the detected objects. To achieve this, Gaussian Focal Loss [22] is employed for the center heatmap prediction, while L1 loss is used for the other attribute predictions. These losses are combined to define the overall loss for the head, denoted as $\mathscr{L}_{\text{det}}$. Plus, given the past BEV observations $\mathbf{B}^f$, in Temporal Context Extraction Module predicts objects' bounding boxes and their classes in the current time t. We minimize the similar loss function $\mathscr{L}_{\text{pred}}$ together with the main detection loss as follows:

$$\mathscr{L} = \lambda_{\text{pred}}\mathscr{L}_{\text{pred}} + \lambda_{\text{det}}\mathscr{L}_{\text{det}} \tag{5}$$

where we use hyperparameters λ_{pred} and λ_{det} to control the strength of each loss term. Based on our grid search, we set $\lambda_{\text{pred}} = 1$ and $\lambda_{\text{det}} = 3$.

IV. EXPERIMENTS

A. Setup

Dataset. We conducted experiments on the nuScenes dataset [7], comprising 1000 various scenes from Boston and Singapore. The dataset is divided into 700/150/150 scenes for training, validation, and testing, respectively. Each scene has an approximate duration of 20 seconds, and key samples are annotated at a rate of 2Hz, resulting in a total of 1.4 million object bounding boxes. Captured by 6 cameras to cover the surround view, the images are RGB with a resolution of 900×1600 pixels. Annotations cover 10 classes for object detection, including car, truck, bus, trailer, construction vehicle, pedestrian, motorcycle, bicycle, barrier, and traffic cone. The dataset defines a region of interest within a 51.2-meter radius from the ground plane for 3D object detection.

Evaluation Metrics. We follow the official evaluation protocol of nuScenes [7], which includes five True Positive metrics: Average Translation Error (ATE), Average Scale Error (ASE), Average Orientation Error (AOE), Average Velocity Error (AVE), and Average Attribute Error (AAE). Using a 2m center distance threshold for matching, these metrics comprehensively evaluate system performance in 3D object detection across various dimensions such as transformations, scale, direction, speed, and attribute recognition accuracy. Additionally, we also measure mean Average Precision (mAP) and the NuScenes Detection Score (NDS $= \frac{1}{10}[5\ \text{mAP} + \sum_{\text{mTP}\in\mathbb{TP}}(1 - \min(1, \text{mTP}))]$ where $\mathbb{TP}$ is five TP metrics.

Implementation Details. For the Image backbone to process images, we employed the ResNet50 [25] in conjunction with the Feature Pyramid Network (FPN) [26]. The input images were set to a resolution of 256×704, and we used a dimension D of 256. In addition, experiments were conducted with BEVDet4D [3] and BEVDepth [6] as the base. The BEV features, generated through the view transform module, were configured with a shape of 200×200. Each grid cell covered an area of $0.512m \times 0.512m$, with the entire BEV spanning $[-51.2m, 51.2m]$ along the x and y axes. During training, we utilized the AdamW optimizer [27] and trained the model on 8 NVIDIA GeForce RTX 3090 GPUs, with a batch size of 2 samples per GPU. The initial learning rate was set to 2e-4, and we trained for 20 epochs. Subsequently, we continued training for an additional 4 epochs with a reduced learning rate of 2e-5, resulting in a total of 24 epochs.

B. Quantitative Analysis

Effect of Leveraging Predictive Information. We start by experimenting with existing approaches, including BEVDet4D [3] and BEVDepth [6], to measure the effect of our proposed method, which regularizes the detection model with predictive information. As shown in Table I, applying our approach to both existing models significantly improves the overall multi-camera 3D object detection performance. Note that, in this experiment, scores are from our reproduction on nuScenes [7] validation dataset. Also, we use four past observation frames (i.e., $N = 4$) with the current frame.

Comparison with SOTA approaches. Further, in Table II, our model based on BEVDepth [6] shows comparable performance to the current state-of-the-art approaches, including BEVDet, BEVDet4D, STS, BEVDepth, BEVStereo, SOLOFusion, and VideoBEV. Note that we apply CBGS [23] technique to all models. We observe a notable improvement in metrics evaluating translation (mATE), velocity (mAVE), and orientation (mAOE). We would emphasize that adding more observation frames generally improves the overall detection performance, while our method can be easily applicable to other approaches as well. It would be worth exploring experiments with other better baselines (when their codes are released) to evaluate further whether similar improvements are observed. We leave it as our future work.

Per-Category Analysis. In Table III, we provide a per-category (vehicles, pedestrian, bicycle) detection performance comparison between BEVDepth and ours. Though there is a degradation in mAOE for bicycles, we observe performance improvements in all metrics, including mATE, mASE, mAOE, and mAVE. In particular, pedestrian and bicycle performance improvements are more noticeable, pos-

TABLE II: 3D object detection performance comparison with the recent state-of-the-art approaches. Except for BEVDet [1], other approaches utilize temporal information from consecutive multiple frames. Our model is built upon BEVDepth [6], but we would emphasize that our model can be easily applicable to other BEV-based multi-camera 3D object detection methods as well. Also, in this experiment, we use nuScenes [7] validation set. Note that all models are trained with CBGS (Class-balanced Grouping and Sampling, [23]) enabled.

Method	Backbone	#Frames	Image Resolution	NDS ↑	mAP ↑	mATE ↓	mASE ↓	mAOE ↓	mAVE ↓	mAAE ↓
BEVDet [1]	ResNet50	1	256 x 704	0.379	0.298	0.725	0.279	0.589	0.860	0.245
BEVDet4D [3]	ResNet50	2	256 x 704	0.449	0.316	0.691	0.281	0.549	0.378	0.195
STS [24]	ResNet50	2	256 x 704	0.489	0.377	0.601	0.275	0.450	0.446	0.212
BEVDet4D [3]	ResNet50	8	256 x 704	0.487	0.354	0.607	0.284	0.525	0.286	0.193
BEVDepth [6]	ResNet50	8	256 x 704	0.519	0.399	0.571	0.281	0.463	0.278	0.206
BEVStereo [4]	ResNet50	8	256 x 704	0.527	0.415	0.566	0.284	0.465	0.298	0.195
SOLOFusion [10]	ResNet50	17	256 x 704	0.534	0.427	0.567	0.274	0.511	0.252	0.188
VideoBEV [5]	ResNet50	8	256 x 704	0.535	0.422	0.564	0.276	0.440	0.286	0.198
BEVDepth [6]	ResNet50	2	256 x 704	0.484	0.362	0.617	0.274	0.480	0.393	0.203
BEVDepth [6] w/ Ours	ResNet50	4	256 x 704	0.521	0.393	0.543	0.263	0.455	0.295	0.198
BEVDepth [6] w/ Ours	ResNet50	8	256 x 704	0.530	0.402	0.530	0.271	0.431	0.276	0.201
				(**4.6%**↑)	(**4.0%**↑)	(**8.7%**↓)	(**0.3%**↓)	(**4.9%**↓)	(**11.7%**↓)	(**0.2%**↓)

TABLE III: Categorical comparison of Object Translation, Scale, Orientation and Velocity Estimation Error on nuScenes [7] validation set. Both models trained with CBGS [23].

Models	Agent Type	mATE ↓	mASE ↓	mAOE ↓	mAVE ↓
BEVDepth [6]	Vehicles	0.447	0.175	0.125	0.266
	Pedestrian	0.624	0.304	0.652	0.348
	Bicycle	0.466	0.276	**0.996**	0.164
Ours	Vehicles	**0.435**	**0.173**	**0.109**	**0.248**
	Pedestrian	**0.556**	**0.295**	**0.560**	**0.326**
	Bicycle	**0.390**	**0.264**	1.023	**0.133**

sibly due to the effect of learning temporal cues for their behaviors, which helps to detect small dynamic objects.

Effect of Each Components. Table IV shows the results of various experiments conducted to analyze the module combinations used in our method. In this experiment, ResNet-50 is used as the backbone, image resolution is 256 x 704, and BEVDet4D [3] is used as the base model with 4 frames. Model A represents the baseline performance of BEVDet4D. The results of B demonstrate that the utilization of predicted BEV feature provides more insightful information for understanding the current scene compared to concatenating previous features used in BEVDet4D. Experiment C provides evidence that the Prediction Encoder is a crucial module for utilizing past information to predict the current scene effectively. Experiments D and E demonstrate the usefulness of the Temporal-Spatial encoder and Dense encoder within the Prediction Encoder. F concludes that all these components collectively contribute to enhancing the overall performance of the model.

Comparison with other Fusion Methods. Table V presents our experimental exploration of different fusion strategies for integrating Observed BEV features and Predicted BEV features within the Fusion Module. In this table, we conduct experiments exclusively on the validation dataset, without employing CBGS [23], and utilize four frames based on BEVDet4D [3]. (A) demonstrates the performance of the

TABLE IV: Ablation study on nuScenes validation set. *Abbr.* C: Use of Concatenated BEV frames, F: Use of Fusion Module, S: Spatiotemporal BEV Encoder, M: Use of Multi-resolution Features.

Model	C	F	S	M	NDS ↑	mAP ↑
Model A (BEVDet4D [3])	✔				0.454	0.330
Model B		✔	✔	✔	0.475	0.357
Model C	✔	✔			0.481	0.361
Model D	✔	✔	✔		0.484	0.366
Model E	✔	✔		✔	0.485	0.360
Model F (Ours)	✔	✔	✔	✔	**0.489**	**0.374**

TABLE V: Performance comparison with variants of feature-level fusion methods.

Model	NDS ↑	mAP ↑
A. Ours w/ Channel-wise Attention	0.436	0.312
B. Ours w/ Concat followed by 1D ConvNet	0.480	0.357
C. Ours w/ Concat followed by 2D ConvNet	0.482	0.364
D. Ours (use of Deformable Attention)	**0.489**	**0.374**

Channel-Wise Attention approach, where a learnable BEV feature is employed as the query, and attention is carried out by utilizing the features of the corresponding grid cells of $\mathbf{B}^o$ and $\mathbf{B}^f$ for key and value roles in order to extract information from each BEV grid cell. In (B) and (C), we employ concatenation to combine $\mathbf{B}^o \in \mathbb{R}^{H\times W\times D}$ and $\mathbf{B}^f \in \mathbb{R}^{H\times W\times D}$ to form a BEV feature with a shape of $\mathbb{R}^{H\times W\times 2D}$, followed by fusion using Conv1D and Conv2D methods, respectively, to extract BEV features with dimensions $H \times W \times D$. Lastly, in (D), we implement the Fusion Module approach as previously described. Our experimental findings confirm that the Deformable Attention-based method, where each feature $\mathbf{B}^o$ and $\mathbf{B}^f$ share offsets, is the most suitable fusion method.

C. Qualitative Analysis

In Fig. 3, we visually present the results obtained from the nuScenes validation dataset. Predicted bounding boxes are

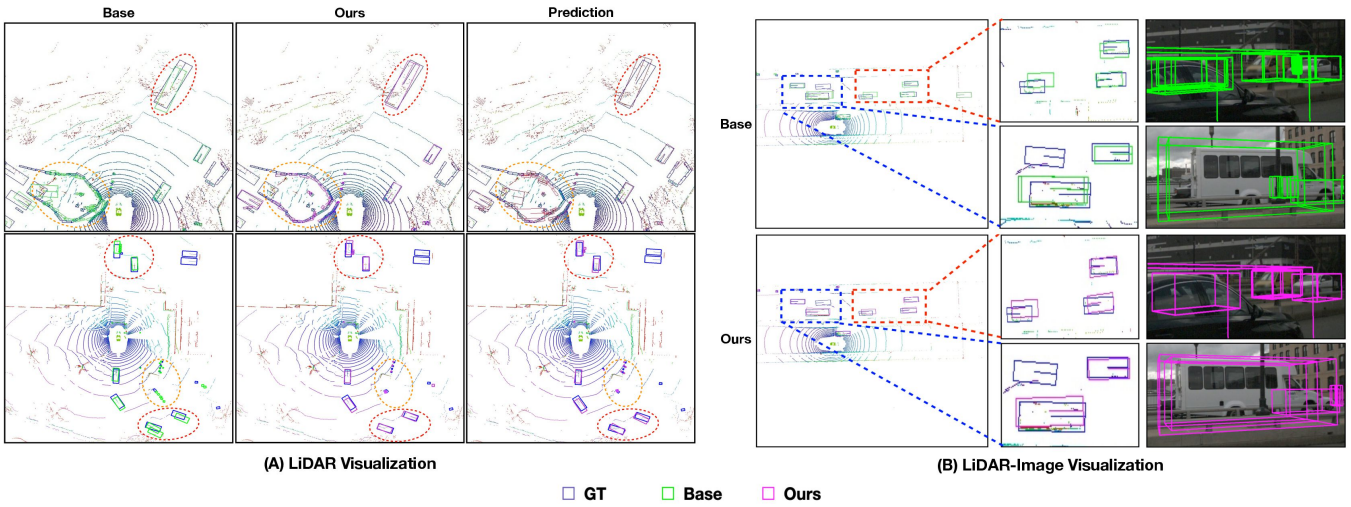


Fig. 3: Examples of detected objects (visualization on LiDAR and LiDAR-Image). (A) shows results from the base model, ours, and prediction (i.e., poses of predicted objects given only past observations). (B) provides a clearer view of the results from the base model and ours through visualization on LiDAR and image.

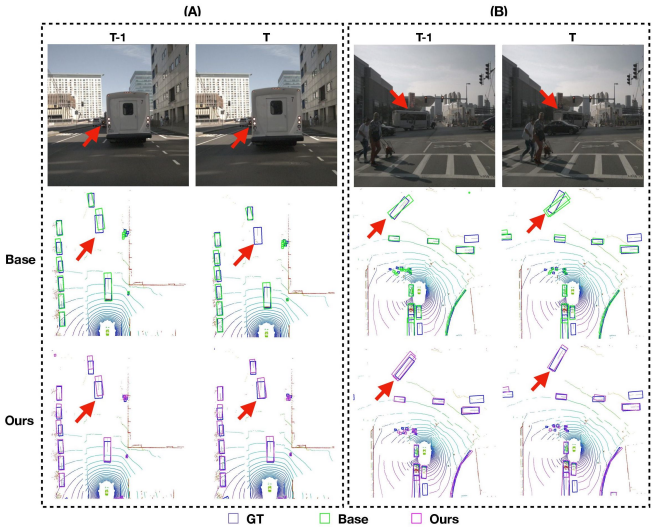


Fig. 4: Visualization of two scenarios: one with occlusion occurring over time (A) and the other with an object making a left turn (B). The results from both scenarios demonstrate the contribution of predictions to object detection.

represented in green, while the corresponding ground truth bounding boxes are shown in blue. In (A), the region within the red circle highlights an example where the Baseline model fails to accurately locate the object. However, by leveraging predictions that track object positions from previous features and combining them with the current feature, we demonstrate visually that more accurate detection can be achieved. The area delineated by the orange dashed line demonstrates how predictions can significantly reduce ghost boxes in regions where objects are densely concentrated.

(B) shows the results of the base model and our proposed model on objects with motion or occlusion. The visualization on the enlarged LiDAR and the corresponding image are shown together. The results show that our model is more accurately positioning the box.

As shown in Figure 4, the utilization of prediction information enables continuous detection even for occluded objects (see A) and improves the detection certainty for agents changing their direction (see B). This confirms that learning objects' motion further enhances the use of temporal cues, thereby enhancing overall detection accuracy. It demonstrates how predicting the current scene through the Temporal Context Extraction Module can be beneficial for detection.

V. CONCLUSION

In this paper, we propose a novel model as a way to utilize historical information to predict the current scene and improve detection performance in the current scene using the predictive information. Our model is designed to focus on the points that guide in better understanding the present by utilizing the continuous and temporally correlated features in the input data of autonomous driving or robotics. Through experiments on the NuScenes dataset, we have demonstrated the effectiveness of using predicted BEV features in multi-view 3D object detection. Our approach has shown performance improvements over the base model in terms of mAP, NDS metrics, as well as in translation, scale, orientation, and velocity. Furthermore, our model can be easily applied to 3D detection models that utilize images from previous timesteps in a plug-and-play manner. Future research could explore methods for refining predicted BEV features to leverage higher-quality prediction information.

ACKNOWLEDGMENT

This work was supported by Autonomous Driving Center, Hyundai Motor Company R&D Division. We thank Mincheol Chang and Daewon Chae for their helpful discussions and feedback. This work was partly supported by the National Research Foundation(NRF) grant (RS-2023-00212484, 15% and NRF-2021R1A6A1A13044830, 15%) and by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (IITP-2024-2020-0-01819 (10%), 2022-0-00043 (30%), and RS-2022-00167194, 15%).

REFERENCES

[1] J. Huang, G. Huang, Z. Zhu, and D. Du, "Bevdet: High-performance multi-camera 3d object detection in bird-eye-view," *arXiv preprint arXiv:2112.11790*, 2021.

[2] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *European conference on computer vision*. Springer, 2022, pp. 1–18.

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

[4] Y. Li, H. Bao, Z. Ge, J. Yang, J. Sun, and Z. Li, "Bevstereo: Enhancing depth estimation in multi-view 3d object detection with temporal stereo," 2023, pp. 1486–1494.

[5] C. Han, J. Sun, Z. Ge, J. Yang, R. Dong, H. Zhou, W. Mao, Y. Peng, and X. Zhang, "Exploring recurrent long-term temporal fusion for multi-view 3d perception," *arXiv preprint arXiv:2303.05970*, 2023.

[6] Y. Li, Z. Ge, G. Yu, J. Yang, Z. Wang, Y. Shi, J. Sun, and Z. Li, "Bevdepth: Acquisition of reliable depth for multi-view 3d object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 2, 2023, pp. 1477–1485.

[7] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[8] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *European Conference on Computer Vision*. Springer, 2020, pp. 194–210.

[9] Y. Wang, V. Guizilini, T. Zhang, Y. Wang, H. Zhao, , and J. M. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *The Conference on Robot Learning (CoRL)*, 2021.

[10] J. Park, C. Xu, S. Yang, K. Keutzer, K. Kitani, M. Tomizuka, and W. Zhan, "Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection," in *International Conference on Learning Representations*, 2023.

[11] Y. Liu, J. Yan, F. Jia, S. Li, Q. Gao, T. Wang, X. Zhang, and J. Sun, "Petrv2: A unified framework for 3d perception from multi-camera images," *arXiv preprint arXiv:2206.01256*, 2022.

[12] Y. Z. J. G. S. C. Y. W. Shaoheng Fang, Zi Wang, "Tbp-former: Learning temporal bird's-eye-view pyramid for joint perception and prediction in vision-centric autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023.

[13] N. M. S. D. J. H. V. B. R. C. A. K. Anthony Hu, Zak Murez, "Fiery: Future instance segmentation in bird's-eye view from surround monocular cameras," in *Proceedings of the International Conference on Computer Vision*, 2021.

[14] B. J. J. C. S. C. Z. Y. J. Q. H. Z. Q. L. Yihan Hu, Wenxin Shao, "Hope: Hierarchical spatial-temporal network for occupancy flow prediction," *arXiv preprint arXiv:2206.10118*, 2022.

[15] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, L. Lu, X. Jia, Q. Liu, J. Dai, Y. Qiao, and H. Li, "Planning-oriented autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[16] Z. Zong, D. Jiang, G. Song, Z. Xue, J. Su, H. Li, and Y. Liu, "Temporal enhanced training of multi-view 3d object detector via historical object prediction," *arXiv preprint arXiv:2304.00967*, 2023.

[17] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3d object detection and tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 11 784–11 793.

[18] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[19] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[20] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *TPAMI*, 2019.

[21] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.

[22] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 734–750.

[23] B. Zhu, Z. Jiang, X. Zhou, Z. Li, and G. Yu, "Class-balanced grouping and sampling for point cloud 3d object detection," *arXiv preprint arXiv:1908.09492*, 2019.

[24] Z. Wang, C. Min, Z. Ge, Y. Li, Z. Li, H. Yang, and D. Huang, "Sts: Surround-view temporal stereo for multi-view 3d detection," *arXiv preprint arXiv:2208.10145*, 2022.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[26] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[27] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.