2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

CR3DT: Camera-RADAR Fusion for 3D Detection and Tracking

Nicolas Baumann*, Michael Baumgartner*, Edoardo Ghignone*, Jonas Kühne*,
Tobias Fischer†, Yung-Hsu Yang†, Marc Pollefeys†, and Michele Magno*

***Abstract*—To enable self-driving vehicles accurate detection and tracking of surrounding objects is essential. While Light Detection and Ranging (LiDAR) sensors have set the benchmark for high-performance systems, the appeal of camera-only solutions lies in their cost-effectiveness. Notably, despite the prevalent use of Radio Detection and Ranging (RADAR) sensors in automotive systems, their potential in 3D detection and tracking has been largely disregarded due to data sparsity and measurement noise. As a recent development, the combination of RADARs and cameras is emerging as a promising solution. This paper presents Camera-RADAR 3D Detection and Tracking (CR3DT), a camera-RADAR fusion model for 3D object detection, and Multi-Object Tracking (MOT). Building upon the foundations of the State-of-the-Art (SotA) camera-only *BEVDet* architecture, CR3DT demonstrates substantial improvements in both detection and tracking capabilities, by incorporating the spatial and velocity information of the RADAR sensor. Experimental results demonstrate an absolute improvement in detection performance of 5.3% in mean Average Precision (mAP) and a 14.9% increase in Average Multi-Object Tracking Accuracy (AMOTA) on the nuScenes dataset when leveraging both modalities. CR3DT bridges the gap between high-performance and cost-effective perception systems in autonomous driving, by capitalizing on the ubiquitous presence of RADAR in automotive applications. The code is available at: https://github.com/ETH-PBL/CR3DT.**

I. Introduction

Perceiving and tracking the local surroundings is a pivotal task in the field of autonomous driving [1], [2], [3]. This has led to the development of a multitude of complex and high-performance 3D object detection and tracking architectures, primarily designed to perform on well-established datasets such as *KITTI*, *Waymo*, or *nuScenes* [4], [5], [2]. Recent 3D object detection methods mainly utilize two distinct sensor setups:

I **LiDAR-based methods for maximum performance:** Models following this paradigm lean heavily on the LiDAR sensor modality [6], [7], [8] to reach high accuracy scores, requiring high computational power and incurring high costs for the sensors and processing unit.

II **Camera-only methods for cost-effectiveness:** Substituting the expensive but highly performant LiDAR sensor with multiple cameras not only significantly reduces cost, but might allow for wider adoption of self-driving technology within the automotive industry [9], [3] while delivering competitive performance.

While both strategies have their merits, a distinct performance gap exists between camera-only and LiDAR-based models. Notably, in 3D detection tasks, SotA camera-only models achieve an mAP of 62.4 % [10], whilst SotA LiDAR-only models reach 69.5 % [11]. Similarly, SotA camera-only methods for 3D tracking achieve 65.3% AMOTA [12], whilst LiDAR-based methods score 71.5% AMOTA [13]. It is worth noting that these impressive performance numbers are reached by leveraging high-resolution image inputs, incorporating temporal information, or utilizing offline detections, as indicated in Table I. The latter implies that the model utilizes future data, rendering such detection systems impractical for real-time, i.e., online applications. These techniques may be used in various combinations, making fair comparisons about model performance difficult (see Table I). This work targets the challenging task of object perception and tracking in autonomous driving and hence uses a computationally feasible, and fully online model that does not use additional temporal information (i.e., data corresponding to previous frames), to which we refer as the *restricted model class*, further detailed in Section IV.

The RADAR sensor, prevalent in the automotive industry, has only recently emerged as a promising modality to bridge the existing performance gap. Previously, the sensor readings were deemed too noisy and sparse to improve 3D detection and tracking tasks [2], [15]. The absence of RADAR-based solutions in the *nuScenes* tracking challenge [16], a dataset that provides RADAR data, highlights this fact. While the sensor readings offer a spatial point cloud akin to LiDARs, although sparser, they also deliver richer measurements, capturing data points that include velocity and RADAR-reflectivity information. Only recently, RADAR has been used to support tasks such as 3D segmentation [17] and 3D detection [18], [19]. Furthermore, different works have highlighted the RADARs increased robustness to adverse weather conditions [15], [20], [21], especially when compared to LiDARs and cameras.

This paper aims to bridge the performance gap between LiDAR and camera-only methods, by leveraging sensor fusion of the camera and RADAR modality. Thus, we introduce CR3DT a 3D detection and tracking solution synthesizing camera and RADAR data, capitalizing on the added velocity data of the RADAR readings. We opted to fuse these two types of sensor data within the Bird's-Eye View (BEV) space, based on promising results demonstrated by using RADAR

*Nicolas Baumann, Michael Baumgartner, Edoardo Ghignone, Jonas Kühne, and Michele Magno are associated with the Center for Project-Based Learning, D-ITET, ETH Zurich.

†Tobias Fischer, Yung-Hsu Yang, and Marc Pollefeys are associated with the Computer Vision and Geometry Group, D-INFK, ETH Zurich.

Nicolas Baumann, Michael Baumgartner, Edoardo Ghignone, and Jonas Kühne contributed equally to this work. Corresponding author: Jonas Kühne.

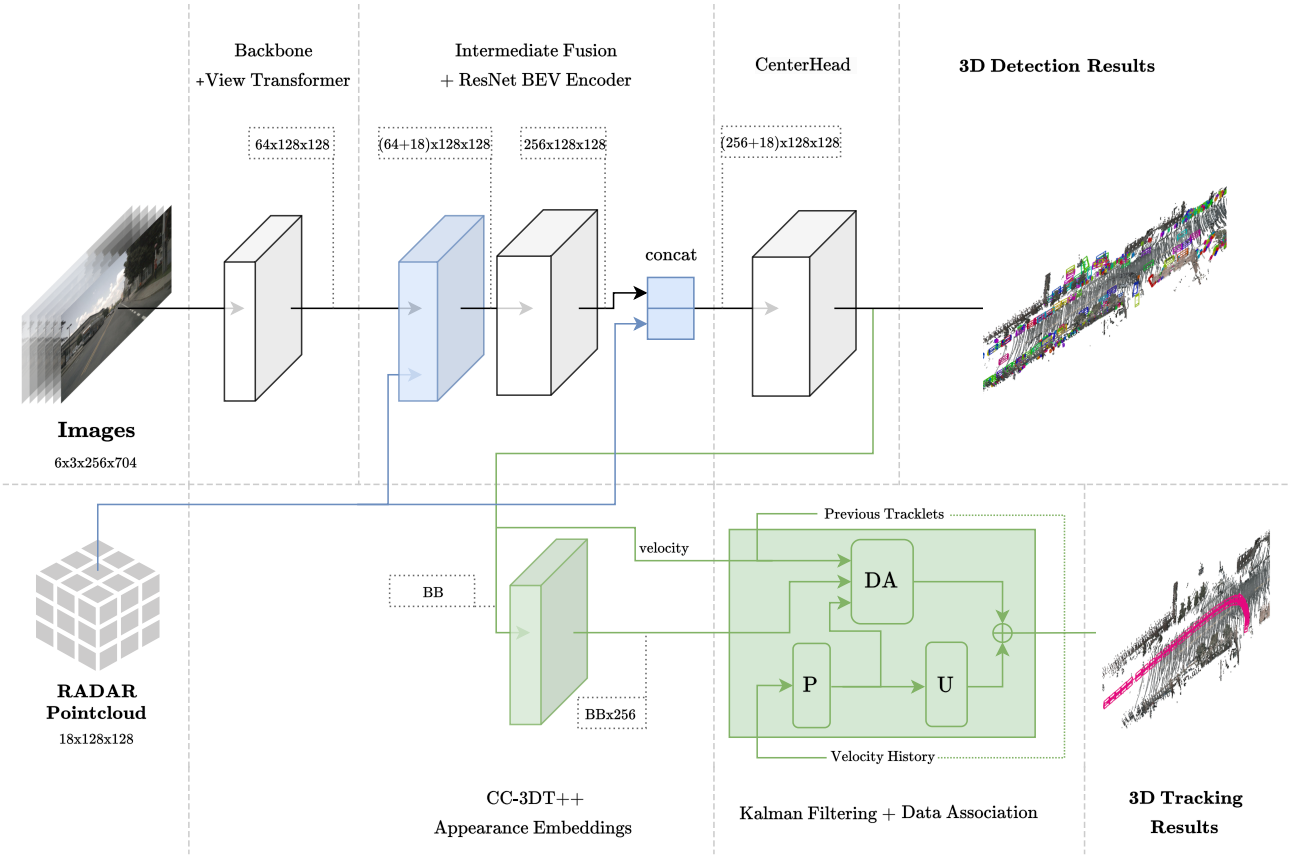

Figure 1: **Architectural Overview.** The model presented in this work extends *BEVDet* [1]. Detection and tracking contributions are highlighted in light blue and green, respectively. Model inputs and outputs are highlighted in bold, with the input from the six different camera views being RGB images with resolution 704×256, and the radar input being encoded in a 128×128 BEV grid, in which each grid cell contains 18 features. The image stream is processed through a *ResNet-50* backbone and then projected into BEV space utilizing a Lift Splat Shoot (LSS) view transformer, while the RADAR stream is encoded in pillars with the feature encodings detailed in Section III-A. The two intermediate outputs are then concatenated and processed through a *ResNet* BEV Encoder as in [1]. After this step, the rasterized RADAR pillars are concatenated once more through a residual connection. The generated BEV features are then passed through a *CenterPoint* detection head [8], which generates the detection result. The output bounding boxes are utilized to select Region of Interests (ROIs) and extract appearance embeddings [3], [14]. Finally, these embeddings are used in the *Data Association (DA)* step, which generates the tracking results leveraging a refined velocity estimate together with the velocity output from the detection module. Such an estimate is obtained with a Kalman Filter (KF), as in the *KF3D* setting of [3], a two-stage state estimation that consists of a *Prediction Step (P)* and an *Updating Step (U)*. Green-dotted lines represent stored data saved across timesteps.

in this domain [17], [18], [19]. Additionally, the BEV space has yielded high-performance outcomes in both camera-only techniques [1], [9], [22], [23] and camera-LiDAR fusion methods [7], [24]. Furthermore, the BEV space as an intermediate representation facilitates the incorporation of the 3D data from the RADAR, a notable advantage over other works that relied on 2D projections [21], [25]. The main points of this work are summarized below:

- **Sensors fusion architecture:** The proposed CR3DT architecture integrates RADAR data using intermediate fusion both before and after the BEV encoding head. It utilizes a quasi-dense appearance embedding head for tracking, trained similarly to [3], [14]. Additionally, the tracker explicitly uses the velocity estimates of the detector for the object association. The model further detailed in Fig. 1, represents one of the first tracking architectures that leverage the complementary camera and RADAR modalities.
- **Detection performance evaluation:** CR3DT achieves an mAP of 35.1 % and a nuScenes Detection Score (NDS) of 45.6 % on the *nuScenes* 3D detection validation set. This outperforms SotA single-frame, camera-only detection models within the *restricted model class* as defined in Section IV by 5.3 % mAP points. With the rich velocity information contained in the RADAR data, the detector furthermore reduces the mean Average Velocity Error (mAVE) by 45.3 % versus the previously mentioned SotA camera-only detector.

- **Tracking performance evaluation:** CR3DT demonstrates a tracking performance of 38.1% AMOTA on the *nuScenes* tracking validation set. This corresponds to a 14.9% AMOTA points improvement against SotA camera-only tracking models in the *restricted model class* as defined in Section IV. The explicit use of velocity information and further improvements in the tracker significantly reduces the number of ID Switches (IDS) by about 43 % versus the mentioned SotA model.

II. Related Work

Popular autonomous driving datasets such as *KITTI* [4], *Waymo* [5], and *nuScenes* [2] allow model performance comparison of perception systems in the tasks of 3D object detection and tracking. Recent successful models typically leverage the BEV space to tackle these problems. In the following, we discuss previous LiDAR-based models, camera-only models, and camera-RADAR models addressing 3D object detection. Finally, we discuss the SotA models operating in 3D BEV space to perform object tracking.

A. LiDAR-Based 3D Object Detection

One of the first works that successfully used LiDAR data in the 3D object detection task was *VoxelNet* [26], which managed to encode sparse point cloud data into voxels thanks to the introduction of the novel Voxel Feature Encoding (VFE) layer. Successively, *SECOND* [27] tried to remedy some of the performance problems that *VoxelNet* suffered from – mainly due to the incorporation of a 3D convolution module in its BEV encoding layer. *PointPillars* [6] then was a seminal work that built upon *VoxelNet* and *SECOND*, removing the 3D convolution and encoding the point cloud features directly into pillars instead of voxels. More recent 3D object detection approaches like *CenterPoint* [8] typically utilize either *VoxelNet* or *PointPillars* in their feature extraction backbone while improving on the object detection module. *CenterPoint* itself introduced a highly effective two-stage detection architecture, which in the first stage extracts a rotation-agnostic heatmap of object positions, for which then in the second stage the bounding boxes are regressed. It is noteworthy, that these methodologies and architectures are inherently sensor-agnostic and are thus also interesting for utilization with RADAR point cloud data as well as *lifted* image features in BEV space, as used in CR3DT.

B. Camera-Only 3D Object Detection

The search for 3D detection architectures has extensively focused on cost-effective camera-only models. When using camera-based systems in BEV space, 2D object detection techniques are being leveraged as much as possible due to their proven effectiveness. They are extended into the third dimension using multiple cameras spaced around the car. Recent development in this field yielded a technique called LSS [28], a pioneering method that leverages the camera intrinsics and extrinsic to project 2D image features into the BEV space of the car. While LSS utilizes the camera parameters as well as a per pixel attention-style Convolutional Neural Network (CNN) operation in BEV space, there exist other methods such as parameter-free lifting [17] or methods based on deformable attention [9]. In this work, however, we restrict ourselves to the simpler LSS technique that is also utilized in the SotA *BEVDet* series.

Recent models leverage this image-view transformation to employ detection or segmentation heads on the aggregated BEV feature space [1], [9], [23]. They differ mostly in the way the image features are extracted and *lifted*, as well as subsequently encoded in the aggregated BEV space to be prepared for the detection or segmentation head. The *CenterPoint* architecture [8], while initially developed for the LiDAR modality, has been proven to work well with image-based features in BEV space, as shown by *BEVDet* [1], which denotes the current SotA in camera-only online 3D object detection within the *restricted model class* as defined in Section IV. Therefore, our proposed solution builds upon the *BEVDet* architecture, additionally integrating the RADAR modality, and improving performance in terms of mAP, NDS, and mAVE.

C. Camera-RADAR Fusion Models

Recent camera-RADAR fusion architectures have demonstrated the potential of sensor-fusion within the 3D BEV space. *SimpleBEV* [17] represents an impactful work in 3D object segmentation and BEV-based sensor fusion, where the RADAR point clouds are *rasterized* and then concatenated with the BEV image-view features obtained from a *ResNet-101* backbone. This concatenation fuses the spatial RADAR data with the feature-rich camera data after the projection of the image features into 3D BEV space. The proposed technique demonstrated an approximate 8%-point increase in absolute Intersection over Union (IOU) score for the *nuScenes* segmentation task. Following a similar idea, the *BEVDet4D* paradigm introduced in [23] shows the advantages of concatenating different sensor streams in BEV space, with the key difference being, that instead of RADAR data previous camera BEV features are fused with the current ones. Lastly, recent interest in cross-attention from the sensor-fusion community can be seen in [18], where deformable cross-attention can be found as a competitive method for BEV space multi-modal fusion in the tasks of BEV segmentation and 3D detection and tracking. Hence, recent work strongly suggests that fusing RADAR with camera data yields a cost-effective performance increase across different perception tasks for autonomous driving. Following this trend, this work proposes a 3D detection and tracking architecture that evaluates different fusion strategies in BEV space and achieves significant improvement in detection and tracking performance metrics, when compared to the SotA baselines belonging to the same *restricted model class*.

D. Multi Object Tracking

In the context of 3D MOT, numerous solutions have traditionally relied on the LiDAR sensor modality due to its ability to provide a comprehensive 360-degree Field of View (FoV) [8], [29], [30], [31]. In contrast, 3D BEV tracking systems based on camera-only detection models have been relatively rare [32], [33]. This scarcity is, in part, because

camera-based approaches often extend 2D MOT techniques, which rely on tracking by detection, and utilize motion or appearance cues for association between frames. These 2D techniques inherently lack the capacity to leverage the rich spatial information provided by a 360-degree FoV sensor such as LiDAR.

More recent works in camera-only tracking have begun to diverge from 2D MOT techniques, focusing on associating temporal image-feature embeddings of the objects being tracked [14], [34]. These methods capitalize on quasi-dense similarity learning to enable robust and precise tracking. One notable development in this line of research is the *CC-3DT* [3] model, which innovatively extends tracking capabilities to handle joint associations between different cameras, thereby enabling tracking tasks to be performed through cross-camera correlation. We exploit the tracker of *CC-3DT* and enhance it, demonstrating in Table IV that our tuning improves the out-of-the-box performance of the *CC-3DT* tracker across all considered metrics.

III. Model Architecture

A broad overview of the model architecture is depicted in Fig. 1. It is inspired by the *BEVDet* architecture [1] but proposes a novel fusion operation to expand the sensor modalities from camera-only to camera-RADAR. Subsequently, the proposed CR3DT architecture incorporates the *CC-3DT++* tracking head, which explicitly uses the improved velocity estimations of the RADAR-augmented detector in its data association. By projecting the features of the six camera views and five RADARs positioned around the ego-car into the BEV space, we achieve the full 360-degree FoV as in LiDAR-based tracking methods.

A. Sensor Fusion in BEV Space

A visual representation of the RADAR data within the same BEV space as the lifted image features can be seen in Fig. 2. The image features however are not plotted for visibility reasons. Further, it is worth noting that the LiDAR point cloud is solely used to visualize the surrounding BEV space more comprehensively for the reader and is not used for training of any kind. We utilize a *PointPillars* [6] inspired fusion method consisting of aggregation and concatenation without any reduction step, based on our findings in Section V-B. The BEV grid configuration is set to $51.2\,\mathrm{m}$ range with a $0.8\,\mathrm{m}$ resolution, which yields a feature grid of `(128x128)`.

Similar to *PointPillars*, the image features are lifted directly into 64-channel, single cell *pillars* in the BEV grid, leading to an image feature size of `(64x128x128)`. The RADAR data is also directly aggregated (employing averaging) into single pillars, utilizing all 18 RADAR channels, including the point's x, y, and z locations. This implicitly adds the *centroid* of the radar point cloud in each pillar to the list of features to be used. Notice that we did not augment the RADAR data in any way, in contrast to *PointPillars* augmentation efforts for LiDAR point clouds [6]. This is due to the fact that the RADAR point cloud already includes more information compared to LiDAR, as explored in [17].

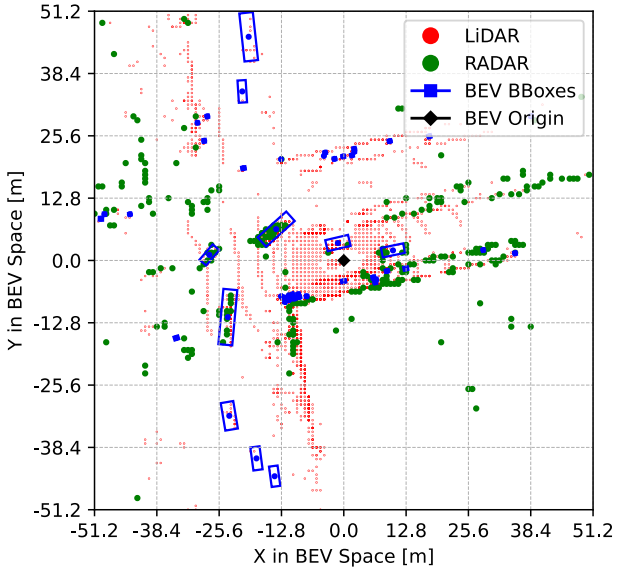

Figure 2: A visualization of the RADAR point cloud aggregated into BEV space for the fusion operation. The LiDAR point cloud is shown for visualization purposes only. The BEV space has a grid configuration of $52.2\,\mathrm{m}$ with a resolution of $0.8\,\mathrm{m}$, resulting in a `(128x128)` dimensional feature space.

Subsequently, the RADAR tensor of size `(18x128x128)` is concatenated to the image tensor of size `(64x128x128)` and both are fed into the BEV feature encoding layer as a `((64+18)x128x128)` tensor. Furthermore, the ablation study discussed in Section V-B showed, that it is beneficial to add a residual connection to the output of the feature encoding layer of dimension `(256x128x128)`, leading to a final input size to the *CenterPoint* [8] detection head of `((256+18)x128x128)`.

B. Tracker Architecture

The tracking architecture, which is integrated in CR3DT and visible in Fig. 1, is based on the *CC-3DT* model [3]. We introduce multiple technical adaptations in the data association step of the KF, which significantly increase the tracking performance in terms of AMOTA, Average Multi-Object Tracking Precision (AMOTP), and IDS.

The data association step is needed to associate the objects of two different frames and is based both on motion correlation and visual feature similarity. During training, 1D visual feature embedding vectors are obtained via quasi-dense multiple-positive contrastive learning as in [14], [34]. Both the detections and feature embeddings are then used in the tracking stage of *CC-3DT* [3], which is agnostic to the extraction process of the feature embeddings. The data association step, referred to as *DA* in Fig. 1, was modified to leverage the improved positional detections and velocity estimates of CR3DT. Specifically, the weighting terms and the formulation of the motion correlation matrix were redefined as detailed in the following paragraphs.

Following the naming convention in *CC-3DT* [3], the detections $\mathcal{D}_t$ are associated with the active tracks of the KF

$\mathcal{T}_t$ at time t with a greedy assignment given an affinity matrix $\mathbf{A}(\mathcal{T}_t, \mathcal{D}_t) \in \mathbb{R}^{\|\mathcal{T}_t\| \times \|\mathcal{D}_t\|}$. The matrix is composed of the appearance embedding similarity matrix $\mathbf{A}_{\texttt{deep}}(\mathcal{T}_t, \mathcal{D}_t)$, the motion correlation matrix $\mathbf{A}_{\texttt{motion}}(\mathcal{T}_t, \mathcal{D}_t)$, and the location correlation matrix $\mathbf{A}_{\texttt{loc}}(\mathcal{T}_t, \mathcal{D}_t)$, weighted according to the following equation:

$$\begin{aligned}\mathbf{A}(\mathcal{T}_t, \mathcal{D}_t) =& w_{\texttt{deep}} \mathbf{A}_{\texttt{deep}}(\mathcal{T}_t, \mathcal{D}_t) \\ &+ w_{\texttt{motion}} \mathbf{A}_{\texttt{motion}}(\mathcal{T}_t, \mathcal{D}_t) \mathbf{A}_{\texttt{loc}}(\mathcal{T}_t, \mathcal{D}_t), \quad (1)\end{aligned}$$

where $w_{\texttt{deep}}$ and $w_{\texttt{motion}} = 1 - w_{\texttt{deep}}$ are scalars. In this work, specific care was dedicated to the tuning of $w_{\texttt{motion}}$ to put more emphasis on the refined motion affinity terms. The matrices $\mathbf{A}_{\texttt{deep}}$ and $\mathbf{A}_{\texttt{loc}}$ were left unchanged. The terms of the affinity matrix $\mathbf{a}_{\texttt{motion}}(\tau_t, d_t)$ corresponding to a single track $\tau_t \in \mathcal{T}_t$ and detection $d_t \in \mathcal{D}_t$ are newly defined as:

$$\mathbf{a}_{\texttt{motion}}(\tau_t, d_t) = \mathbf{a}_{\texttt{vel}} \mathbf{a}_{\texttt{centroid}} + (1 - \mathbf{a}_{\texttt{vel}}) \mathbf{a}_{\texttt{pseudo}} \quad (2)$$

where the centroid correlation $\mathbf{a}_{\texttt{centroid}}$ and the state difference correlation $\mathbf{a}_{\texttt{pseudo}}$ are defined as in [3]. The new velocity correlation weight $\mathbf{a}_{\texttt{vel}}$ is computed as:

$$\mathbf{a}_{\texttt{vel}}(\tau_t, d_t) = \exp(-\frac{1}{r} |\mathbf{v}_{\tau_t} - \mathbf{v}_{d_t}|), \quad (3)$$

where $\mathbf{v}_{\tau_t}$ and $\mathbf{v}_{d_t}$ represent the subset of states related to the velocities (i.e., $\mathbf{v}_s = [v_x, v_y, v_z]^T, s \in \{\tau_t, d_t\}$) of a single track τ_t in the KF and a detection d_t respectively. These terms represent velocities, whereas in [3] pseudo velocities based on the difference of the centroid position between two frames are used. Furthermore, in [3] the pseudo velocities are compared by using the cosine similarity, whereas we use the velocity-based exponential function as shown in Eq. (3).

We refer to our extended implementation of *CC-3DT* with adapted thresholds, weights, and the introduction of our novel velocity similarity weight as *CC-3DT++*. The three changes are motivated by ablation studies, detailed in Table IVa, Table IVb, and Table IVc, respectively.

IV. Experimental Setup

This section details the setup and definitions of the detection and tracking baselines, as well as the training hardware that was utilized to train the described models. Within this work, both detection and tracking evaluations were performed on the well-known *nuScenes* dataset, as it contains both the RADAR and camera data, necessary for our method. We tested our model on the *nuScenes* validation set, enabling comparison with related work. For computational reasons and to facilitate comparison, all models were trained without the usage of Class-Balanced Grouping and Sampling (CBGS).

A. Detection Baseline

Within this work, we built upon the well-established *BEVDet* architecture [1] and refer to it as the detector baseline, or SotA camera-only model for single-frame inputs of the indicated image size. Due to computational resource limitations, all forthcoming 3D BEV detection performance results that involve *BEVDet* based architectures are utilizing a *ResNet-50* image encoding backbone and an image input size of `(3x256x704)`. Furthermore, only the current six input views per inference are used, i.e., neither past nor future temporal image information is utilized in any of our models, adhering to a fully online detection setting. We refer to this combination of settings as the *restricted model class*.

B. Training Hardware

The training was performed on a single GeForce RTX 3090. To replicate the training of the *BEVDet* network, which was performed on 8 Graphics Processing Units (GPUs) with a batch size of 8, training was performed with gradient accumulation over 8 steps. To alleviate the sparsity inherent to RADAR data, five sweeps were used, i.e., the sweep associated with the current camera frames as well as the four previous RADAR sweeps were accumulated. Note that the RADAR sensor has a higher data rate than the camera, hence the radar sweeps that are being accumulated correspond to timestamps that are strictly after the timestamp of the previous image frame. No additional temporal information was used in our fusion models.

V. Results

For the detection results, the mAP, NDS, and mAVE scores are reported, while the tracking results use the AMOTA, AMOTP, and IDS metrics. These scores are a subset of the official *nuScenes* metrics, where NDS and AMOTA incorporate all the other metrics for detection and tracking respectively. For further explanation of the official *nuScenes* metrics we refer to [2].

Detection Models	Input	Resolution	Frames	mAP [%]↑	NDS [%]↑	mAVE [$\frac{m}{s}$]↓	FPS [$\frac{1}{s}$]↑
BEVDet (R50) [1]	C	256×704	1	29.8	37.9	0.86	**30.2**
BEVFormerV2 † [35]	C	256×704	1	34.9	42.8	0.82	-
CR3DT (*ours*)	C+R	256×704	1	**35.1**	**45.6**	**0.47**	28.5
StreamPETR [12]	C	512×1408	8	50.4	59.2	0.26	6.4
BEVDet (R101) [1]	C	640×1600	1	39.7	47.7	0.82	9.3
BEVDet4D-Base [23]	C	640×1600	2	42.1	54.5	**0.30**	1.9
HoP-BEVFormer [10]	C	640×1600	4	45.4	55.8	0.34	-
CRN (R50) [18]	C+R	256×704	4	49.0	56.0	0.34	**20.4**
CRN (R101) [18]	C+R	512×1408	4	**52.5**	**59.2**	0.35	7.2
CenterPoint [8]	L	-	3	**56.7**	**65.3**	-	-

†: Offline 3D detections (i.e., using future information).

TABLE I: Detection results on the *nuScenes* validation set. The top section lists detection models that conform to the resolution and temporal frame settings adopted in this work. The middle section includes models without such restrictions, granting them a significant advantage in detection performance. The last row contains a LiDAR-based model for reference. *C, R, L* denote the sensor modalities, camera, RADAR and LiDAR, respectively. Frames Per Second (FPS) are reported from literature using an RTX 3090 GPU.

A. Detection Results

Table I shows the detection performance of CR3DT compared to the baseline camera-only *BEVDet* (R50) architecture. It is evident that the inclusion of the RADAR sensor modality significantly increases the detection performance. Within the small resolution and temporal frame constraints, CR3DT

manages to achieve a 5.3% mAP percentage point improvement and a 7.7% NDS percentage point improvement against the SotA camera-only *BEVDet*. Even when compared to more complex offline architectures such as *BEVFormerV2*, which uses a transformer-based attention module for perspective view supervision with bi-directional temporal encoders [35], CR3DT surpasses its performance in an online setting.

Furthermore, Table I highlights the benefits of using higher resolutions and incorporating temporal frame information. Models operating under less constrained settings, i.e., higher image resolution or temporal frames, consistently outperform their restricted counterparts. This suggests the CR3DT architecture could achieve better results if not limited by its current constraints, primarily computational ones. For instance, adapting the CR3DT approach to the high-resolution *BEVDet (R101)* might lead to a performance increase. Including temporal information could offer further improvements.

Table I also shows the performance difference between LiDAR and camera-only models. While *CenterPoint* outperforms all camera-based models, integrating camera and RADAR data narrows down this performance gap. The unrestricted *CRN (R101)* comes within 4.2% mAP points of the *CenterPoint* LiDAR baseline, whereas the constrained CR3DT is still 21.3% mAP points behind.

Bins [#]	mAP [%]↑	NDS [%]↑	mAVE [$\frac{m}{s}$]↓
1	**34.66**	**45.14**	**0.45**
10	32.06	42.78	0.52

(a) Fusion Ablation 1: Effect of a z-dimension discretization and subsequent *bev_compressor* module within the sensor fusion step.

Residual Connection	mAP [%]↑	NDS [%]↑	mAVE [$\frac{m}{s}$]↓
Yes	**35.15**	**45.61**	0.47
No	34.66	45.14	**0.45**

(b) Fusion Ablation 2: Effect of a residual connection for the RADAR data to the output of the intermediate fusion step as shown in Fig. 1, without z-discretization.

TABLE II: Detection architecture ablation experiments.

B. Camera-Radar Fusion Ablation

To analyze the camera-RADAR fusion method in more detail, we compare different fusion architectures in Table II. Namely, we evaluate the performance of two different intermediate fusion approaches in Table IIa and the benefit of our residual connection in Table IIb. Our intermediate fusion approaches were inspired by *VoxelNet* [26] and *PointPillars* [6], respectively. The first approach includes a voxelization of the lifted RGB features and the pure RADAR sensor data into cubes of dimension $0.8\,\mathrm{x}\,0.8\,\mathrm{x}\,0.8\,\mathrm{m}$, leading to an alternate feature size of `((64+18)x10x128x128)` and consequently to the necessity of a *bev_compressor* module in the form of a 3D convolution, similar to the one employed in [26]. The latter approach is the one described in Section III, which forgoes the voxelization in the z dimension and the subsequent 3D convolution, and directly aggregates the lifted RGB features and pure RADAR sensor data into pillars, leading to the known feature size of `((64+18)x128x128)`. As can be seen in Table IIa, omitting the discretization in the z dimension and the additional *bev_compressor* in the model architecture leads to an increase in the models mAP and NDS as well as a decrease in its mAVE. Going further with the better model, we add a residual connection to the output of our initial camera-RADAR fusion as seen in the system architecture overview in Fig. 1. Such a residual connection leads to a further increase in the mAP and NDS of our model, although at the cost of an increase in mAVE, as can be seen in Table IIb. It is worth noting though, that the mAVE still outclasses the camera-only *BEVDet* base model, as seen in Table I.

C. Tracking Results

Table III presents the tracking results of our improved *CC-3DT++* tracking model discussed in Section III-B on the *nuScenes* validation set. We report the performance of the tracker on top of both the baseline and SotA camera-only *BEVDet* detector, as well as our CR3DT detection model. Alongside the results of the final *CC-3DT++* tracking architecture, we also report the individual tracking performances utilizing the baseline *CC-3DT* model [3] as well as different intermediate tracking architectures explored in our tracking architecture ablation study in Section V-D.

The proposed CR3DT detector combined with the *CC-3DT++* tracker architecture shows significant improvements in AMOTA and AMOTP compared to the baseline camera-only *BEVDet* detector and *CC-3DT* tracker. Concretely, the individual improvements in the detector and tracker lead to a joint performance gain over the baseline of 14.9 % points in AMOTA and a reduction of $0.11\,\mathrm{m}$ in AMOTP. Furthermore, we see an IDS decrease of about 43 % compared to the baseline.

D. CC-3DT++ Tracking Architecture Ablation

To better understand the impact of our different changes to the original *CC-3DT* [3] tracking architecture and the addition of our velocity-based affinity term, we conduct extensive ablation studies depicted in Table IV. Furthermore, we report the performance of the different tracking architecture configurations on both our baseline *BEVDet* detector as well as our own CR3DT detector in Table III to investigate the effects of our changes more generally.

We ran three main experimental studies, the first was an investigation of the matching score threshold utilized in the greedy matching algorithm of *CC-3DT*, which is tightly coupled to the affinity scores calculated in Eq. (1). We observe significant performance gains both in an increase of AMOTA and a stark decrease in IDS for a slightly lower threshold, with a stark performance decrease for too low of a threshold, as indicated in Table IVa. Secondly, we explored different weightings of the embeddings correlation and motion correlation terms in Eq. (1). As shown in Table IVb, we find that a larger weight on the motion correlation term leads to an AMOTA gain, although at the cost of an IDS increase. Lastly, we combined the best threshold and weighting configuration

Experiment Name	Ablation Parameters			BEVDet: Tracking Performance			CR3DT: Tracking Performance		
	threshold	weight	vel. sim.	AMOTA [%]↑	AMOTP [m]↓	IDS [#] ↓	AMOTA [%]↑	AMOTP [m]↓	IDS [#] ↓
CC-3DT	✗	✗	✗	23.2 (+0.0)	1.48 (+0.00)	2491 (+0)	31.2 (+0.0)	1.34 (+0.00)	2809 (+0)
CC-3DT + Abl. 1	✓	✗	✗	27.7 (+4.5)	1.50 (+0.02)	1140 (-1351)	34.2 (+3.0)	1.39 (+0.05)	**1291 (-1518)**
CC-3DT + Abl. 2	✗	✓	✗	24.5 (+1.3)	**1.45 (-0.03)**	2861 (+370)	33.1 (+1.9)	**1.31 (-0.03)**	3649 (+840)
CC-3DT + Abl. 1 & 2	✓	✓	✗	30.3 (+7.1)	1.49 (+0.01)	1122 (-1369)	37.6 (+6.4)	1.37 (+0.03)	1537 (-1272)
CC-3DT++ (*ours*)	✓	✓	✓	**30.5 (+7.3)**	1.49 (+0.01)	**1121 (-1370)**	**38.1 (+6.9)**	1.37 (+0.03)	1432 (-1377)

TABLE III: Tracking results on the *nuScenes* validation set for different tracker configurations based on both the baseline *BEVDet* and our CR3DT detector. The tracking architectures correspond to the baseline *CC-3DT* tracker as well as the best intermediate models found in the ablation studies shown in Table IV and our final *CC-3DT++* tracker. We observe similar performance gains with both detection backbones for our general tracker improvements over the respective baseline. Although notably, CR3DT benefits more from our novel velocity similarity term explained in Section III-B.

Matching Score Threshold	AMOTA [%] ↑	IDS [#] ↓
0.50 (default)	32.3	2542
0.30	**34.2**	**1291**
0.18	29.5	1424

(a) Abl. 1: Effect of different matching score thresholds in the data association step of the tracker.

$\boldsymbol{w}_{\texttt{deep}}$	$\boldsymbol{w}_{\texttt{motion}}$	AMOTA [%] ↑	IDS [#] ↓
0.5	0.5 (default)	32.3	**2542**
0.75	0.25	28.7	3134
0.25	0.75	**33.1**	3649

(b) Abl. 2: Effect of various affinity matrix weights in Eq. (1).

$\mathbf{a}_{\texttt{motion}}$ Tradeoff Parameter	AMOTA [%] ↑	IDS [#] ↓
cosine similarity (default)	37.6	1537
velocity similarity (*ours*)	**38.1**	**1432**

(c) Abl. 3: Effect of our newly introduced velocity similarity term as the trade-off term in Eq. (2) compared to the original *CC-3DT* cosine similarity term. This is applied to the model with the best-shown threshold and affinity weights from the two previous ablations.

TABLE IV: Tracking architecture ablation experiments conducted on the CR3DT detection backbone. All ablations include a bugfix in the original tracker architecture leading to a small performance gain in the default configuration.

and examined the motion correlation term in Eq. (2) itself. Trying to leverage the much-improved velocity estimations of our detector, we exchanged the original trade-off parameter (the cosine similarity between the predicted motion direction and the observed motion direction in the xy-plane) with a new velocity similarity term, which compares the predicted with the observed velocity directly as explained in Eq. (3). As hypothesized, we observe a further performance increase in both AMOTA and IDS in Table IVc.

It additionally has to be noted, that we fixed a minor bug in the original implementation of *CC-3DT*, leading to the performance increase observed in the default configurations shown in the ablations in comparison to Table III (32.3% AMOTA vs. 31.2% AMOTA) .

Examining Table III now, we can draw more general conclusions regarding our refined tracker. First, we see a similar trend with both detection backbones concerning our general tuning improvements, underlining the merit of a proper analysis of matching scores within the tracker. Secondly, and more interestingly, we see a performance gain on both backbones when introducing our new velocity correlation term into the matching score, albeit a smaller gain on the *BEVDet* detector. This detail is particularly highlighted by the relative improvement in IDS: while the *BEVDet*-based tracker shows almost no change (-0.1%), the camera-RADAR results show a significant decrease in IDS (-6.8%) when the velocity similarity term is used. This shows promise for such a correlation term in trackers in general and also highlights the positive effect of our improved velocity detections, due to the inclusion of RADAR data in CR3DT, for downstream tasks.

E. Computational Results

The computational results were obtained by measuring the inference time required for both the detection and tracking separately, on an RTX 3090 GPU. The combined latency of the camera-only baseline *BEVDet+CC-3DT* (88.1 ms, 11.35 FPS) is slightly faster than the proposed CR3DT (90.04 ms, 11.11 FPS). This is due to the additional computation of the RADAR fusion. Hence, the inclusion of the RADAR modality in CR3DT accounts for a 2.2% relative increase in latency compared to the camera-only baseline. However, note that the tracking inference is not optimized for latency and accounts for 55 ms in either the camera-only or CR3DT pipeline. Hence the detectors, i.e., *BEVDet* and CR3DT alone, yield a latency of 33.1 ms (30.21 FPS) and 35.04 ms (28.54 FPS), respectively, as shown in Table I.

VI. Conclusion

This work presents CR3DT, an efficient camera-RADAR fusion model tailored to 3D object detection and MOT. By integrating RADAR into the SotA camera-only *BEVDet* architecture and introducing the *CC-3DT++* tracking architecture, CR3DT demonstrates a substantial increase in both detection and tracking accuracies — 5.35% mAP and 14.9% AMOTA points, respectively. This approach introduces a promising direction in the field of perception for autonomous

driving, which leads to a cost-effective, performant, and LiDAR-free system. It bridges the performance gap between the high-performance LiDAR-based systems and the more cost-effective camera-only solutions. While this study did not specifically investigate the RADARs inherent resilience to challenging weather conditions, the growing emphasis on RADAR in recent research underscores its potential value in ensuring robust perception under adverse environmental scenarios. Therefore, future work could investigate potential robustness benefits of CR3DT.

ACKNOWLEDGEMENT

We extend our gratitude to Dr. Christian Vogt of the Center for Project-Based Learning at ETH Zurich for his fruitful discussions and proofreading.

REFERENCES

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

[2] 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.

[3] T. Fischer, Y.-H. Yang, S. Kumar, M. Sun, and F. Yu, "Cc-3dt: Panoramic 3d object tracking via cross-camera fusion," in *6th Annual Conference on Robot Learning*, 2022.

[4] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 3354–3361.

[5] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2446–2454.

[6] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 12 697–12 705.

[7] T. Liang, H. Xie, K. Yu, Z. Xia, Z. Lin, Y. Wang, T. Tang, B. Wang, and Z. Tang, "Bevfusion: A simple and robust lidar-camera fusion framework," *Advances in Neural Information Processing Systems*, vol. 35, pp. 10 421–10 434, 2022.

[8] 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.

[9] 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.

[10] 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," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 3758–3767.

[11] T. Lu, X. Ding, H. Liu, G. Wu, and L. Wang, "Link: Linear kernel for lidar-based 3d perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1105–1115.

[12] S. Wang, Y. Liu, T. Wang, Y. Li, and X. Zhang, "Exploring object-centric temporal modeling for efficient multi-view 3d object detection," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 3598–3608.

[13] Y. Chen, Z. Yu, Y. Chen, S. Lan, A. Anandkumar, J. Jia, and J. M. Alvarez, "Focalformer3d: focusing on hard instance for 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8394–8405.

[14] H.-N. Hu, Y.-H. Yang, T. Fischer, T. Darrell, F. Yu, and M. Sun, "Monocular quasi-dense 3d object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 1992–2008, 2022.

[15] S. Yao, R. Guan, X. Huang, Z. Li, X. Sha, Y. Yue, E. G. Lim, H. Seo, K. L. Man, X. Zhu, and Y. Yue, "Radar-camera fusion for object detection and semantic segmentation in autonomous driving: A comprehensive review," *IEEE Transactions on Intelligent Vehicles*, p. 1–40, 2024.

[16] "Nuscenes camera radar tracking leaderboard," https://www.nuscenes.org/tracking?externalData=all&mapData=all&modalities=Camera%2C%20Radar, accessed: 2023-07-25.

[17] A. W. Harley, Z. Fang, J. Li, R. Ambrus, and K. Fragkiadaki, "Simple-bev: What really matters for multi-sensor bev perception?" in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 2759–2765.

[18] Y. Kim, J. Shin, S. Kim, I.-J. Lee, J. W. Choi, and D. Kum, "Crn: Camera radar net for accurate, robust, efficient 3d perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 17 615–17 626.

[19] W. Xiong, J. Liu, T. Huang, Q.-L. Han, Y. Xia, and B. Zhu, "Lxl: Lidar excluded lean 3d object detection with 4d imaging radar and camera fusion," *IEEE Transactions on Intelligent Vehicles*, 2023.

[20] Y. Zhang, A. Carballo, H. Yang, and K. Takeda, "Perception and sensing for autonomous vehicles under adverse weather conditions: A survey," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 196, pp. 146–177, 2023.

[21] T. Broedermann, C. Sakaridis, D. Dai, and L. Van Gool, "Hrfuser: A multi-resolution sensor fusion architecture for 2d object detection," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2023.

[22] 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.

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

[24] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. L. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2023, pp. 2774–2781.

[25] F. Nobis, M. Geisslinger, M. Weber, J. Betz, and M. Lienkamp, "A deep learning-based radar and camera sensor fusion architecture for object detection," in *2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*. IEEE, 2019, pp. 1–7.

[26] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4490–4499.

[27] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, 2018.

[28] J. Philion and S. 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*. Springer, 2020, pp. 194–210.

[29] D. Held, J. Levinson, and S. Thrun, "Precision tracking with sparse 3d and dense color 2d data," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 1138–1145.

[30] X. Weng, J. Wang, D. Held, and K. Kitani, "3d multi-object tracking: A baseline and new evaluation metrics," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 359–10 366.

[31] C. Luo, X. Yang, and A. Yuille, "Exploring simple 3d multi-object tracking for autonomous driving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 488–10 497.

[32] M. Chaabane, P. Zhang, J. R. Beveridge, and S. O'Hara, "Deft: Detection embeddings for tracking," *arXiv preprint arXiv:2102.02267*, 2021.

[33] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," in *European conference on computer vision*. Springer, 2020, pp. 474–490.

[34] T. Fischer, T. E. Huang, J. Pang, L. Qiu, H. Chen, T. Darrell, and F. Yu, "Qdtrack: Quasi-dense similarity learning for appearance-only multiple object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[35] C. Yang, Y. Chen, H. Tian, C. Tao, X. Zhu, Z. Zhang, G. Huang, H. Li, Y. Qiao, L. Lu, J. Zhou, and J. Dai, "Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 17 830–17 839.