FORMATTING INSTRUCTIONS FOR ICOMP 2024 CONFERENCE SUBMISSIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Robust perception in challenging environments is essential for safe and reliable autonomous driving. Multi-sensor fusion, particularly camera-LiDAR-Radar integration, plays a pivotal role in achieving this goal. Different sensors have specific advantages and disadvantages. Existing pipelines are often constrained by adverse weather conditions, where cameras suffer significant degradation. This paper introduces the Camera Bi-directional LiDAR-Radar (CBILR) fusion pipeline, which leverages the strengths of sensors to enhance LiDAR and Radar point clouds. CBILR innovates with a bi-directional prefusion step between LiDAR and Radar, leading to richer feature representations. Prefusion combines LiDAR and Radar points to compensate for individual sensor weaknesses. Next, the pipeline combines all features together in the bird's eye view (BEV) space, resulting in a comprehensive multi-modal representation. Experiments have demonstrated that CBILR achieves superior robustness in challenging weather scenarios.

1 INTRODUCTION

For self-driving systems, it is crucial to develop a fast and accurate 3D object detector that predicts the bounding boxes and categories of road objects. Nowadays, cameras, LiDARs, and Radars are often used in advanced systems such as drones, robots and autonomous vehicles. Many authors only use particular sensors to solve perception problems. This can lead to a generalization problem, because there is a high probability that one type of sensor will be more relevant than others for certain real-world scenarios. Each sensor has advantages and disadvantages. We can only obtain color and texture information about objects from cameras. It is made by projective transformation of the captured 3D scene into a 2D plane and long stages of post-processing raw images, which is the field of color science. For this reason, cameras cannot provide accurate depth information (especially in low light conditions) compared to Radars and LiDARs that operate directly in 3D space Li et al. (2023a); Liang et al. (2022). However, researchers continue to develop perception algorithms that rely only on cameras because it is a more cost-effective approach Li et al. (2023c); Zhang et al. (2022).

1.1 FUSION APPROACHES

Sensor fusion is an essential topic in many perception systems. A lot of papers Zhong et al. (2021); Zhang et al. (2023a) are devoted to LiDAR-camera fusion because LiDARs have higher resolution, are less sparse than Radars and can provide accurate measurements at close range. Since Radar antennas are often installed horizontally, they cannot capture sufficient vertical height information Yingjie Wang et al. (2023). For voxel representation, a highly sparse point cloud means that some voxels contain too few points for processing.

Although LiDARs can provide accurate geometric information about a scene, they do not perform as well as Radars at long distances and can introduce noise when the object is moving Li et al. (2023b). In Nabati & Qi (2020); Nabati et al. (2021) the authors use Radar-camera fusion for 3D object detection and tracking. Since such sensors in many cases have opposite advantages and disadvantages, it is ideal to use multiple sensors Liu et al. (2023); Chen et al. (2023) for robust performance in a variety of scenarios and conditions. We have developed a fusion pipeline focused on improving sensors that can withstand adverse weather conditions.

There are several strategies for sensor fusion. Early fusion directly combines sensor inputs before feeding them into shared feature extractors. Late fusion processes sensor inputs independently and then combines the output results. Mid-level fusion Liang et al. (2018) provides an intermediate representation for each sensor before the final fusion step.

1.2 BEV Perception

A unified representation is necessary to make it easier to transfer knowledge and combine features from different modalities Li et al. (2022). The vast majority of modern perception methods use a bird's eye view (BEV) representation to describe a 3D scene Roddick (2021); Zhu et al. (2023). BEV is an informal perception standard for autonomous driving scenarios Liang et al. (2022). Data from different modalities are used to provide complementary knowledge such as precise locations from point clouds and rich context from images.

Cameras are typically mounted on vehicles parallel to the ground and facing outward. For this reason, images are captured in a Perspective View (PV), which is orthogonal to BEV. Objects of the same shape and size in 3D space can have very different representation in the image plane because of their distance from the camera. The BEV representation does not have scale and occlusion problems compared to PV representation Li et al. (2023a). The transformation from PV to BEV is the inverse perspective map problem, and it can have more than one solution. Before the deep learning era, many works tackled this problem by using a homography transformation matrix because of its computational efficiency. Inverse Perspective Mapping (IPM) has been proposed to address this challenging mapping problem Mallot et al. (1991); Ma et al. (2022). IPM-based methods assume that all points are on the ground plane sacrificing height variation. In complex real-world scenarios, 3D objects like vehicles possess *height* and such transformations can cause noticeable artifacts.

In recent years, data-driven methods have been widely used in complex systems such as self-driving vehicles. Data-driven PV-BEV transformation methods can be divided into three main groups: depth-based, MLP-based, and transformer-based approaches Ma et al. (2022). Depth-based methods estimate the depth distribution of the each image pixel along the ray (coming from the camera) that intersects objects in the environment. This allows to elevate the 2D features to 3D, and then obtain the BEV representations from 3D through dimensionality reduction. Depth-based PV-to-BEV methods can be divided into two classes depending on the using representation: point-based and voxel-based methods. Point-based methods are straightforward, they directly utilize depth estimation to convert pixels into point clouds. Examples: Pseudo-LiDAR Wang et al. (2019), Pseudo-LiDAR++ You et al. (2019), AM3D Ma et al. (2019), PatchNet Ma et al. (2020). Voxel-based method discretize the 3D space to build a regular structure for feature transformation. The disadvantage of this approach is the loss of detailed local spatial information within each voxel. The advantage is that voxels are more effective at covering large-scale scene structure, they are more efficient for 3D scene understanding.

Another approach is to utilize a variational encoder-decoder or MLP to learn implicit representations of camera calibrations to project PV features to BEV. MLP plays the role of a universal approximator of the mapping function from PV to BEV Ma et al. (2022). MLP-based methods focus primarily on working with a single image. The drawback of MLP-based methods is that the learned weights are fixed and not data dependent:

$$Y = WX, W \not\sim X$$

Transformer-based methods employ a top-down strategy constructing BEV queries and searching corresponding features in perspective images through cross-attention mechanism. These methods are more expressive, but hard to train.

1.3 BEV representation vs voxel-based

A voxel-based scene representation cannot provide computational efficiency because such representation describes a 3D scene with dense cubic features $\mathbf{V} \in \mathbb{R}^{H \times W \times D \times C}$ where H, W, D are the spatial resolution of the voxel space and C is the feature dimension. BEV provides the 3D scene with a 2D feature map $\mathbf{B} \in \mathbb{R}^{H \times W \times C}$, which encodes the top view of the scene. This represents the positional information of the ground plane by accumulating voxel features along the vertical z-axis. The height dimension contains less information than the other two dimensions Huang et al. (2023). It is important to note that some researches do not directly use the BEV representation.

In Huang et al. (2023), the authors propose a Tri-Perspective View (TPV) representation for the semantic prediction task due to the lack of z-axis information.

1.4 Camera-to-BEV View Transform

Transforming from a camera view to a bird's eye view is complex because the depth associated with each camera feature pixel can be ambiguous. The idea of camera-to-BEV transformation is based on projective geometry. The process of monocular depth estimation involves generating a unique depth value for each pixel in an image. The state-of-the-art approach involves predicting a categorical distribution of depth for each pixel in the image Reading et al. (2021); Liu et al. (2023); Philion & Fidler (2020). This technique is known as *feature lifting* Philion & Fidler (2020).

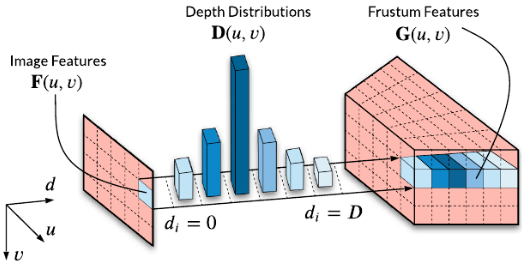


Figure 1: Each feature pixel $\mathbf{F}(u, v)$ is weighted by its depth distribution probabilities $\mathbf{D}(u, v)$ of belonging to D discrete depth bins to generate frustum features $\mathbf{G}(u, v)$ Reading et al. (2021).

In Reading et al. (2021), the model utilize the estimated categorical depth distributions to "lift" an input image into 3D, generating a frustum-shaped point cloud. The frustum feature grid is then transformed into a voxel grid using specific camera calibration parameters, and then collapsed into a BEV feature grid. All steps are well-illustrated in the paper Reading et al. (2021). By associating image features with estimated depths, image information can be projected into 3D space using a frustum feature network. The input to the frustum feature network is an image $\mathbf{I} \in \mathbb{R}^{W_I \times H_I \times 3}$, where W_I, H_I are the image width and height. The network output is a frustum feature grid $\mathbf{G} \in \mathbb{R}^{W_F \times H_F \times D \times C}$, where W_F, H_F are the width and height of the image feature representation, D is the number of discretized depth bins, and C is the number of feature channels. If we have N cameras, the full size of the frustum features is $N \times W_F \times H_F \times D$.

Let's denote (u, v, c) as a coordinate in image features $\mathbf{F}$ and (u, v, d_i) as a coordinate in categorical depth distributions $\mathbf{D}$, where (u, v) is the location of feature pixel, c is the channel index, and d_i is the depth bin index. In order to create a frustum feature grid $\mathbf{G}$, each feature pixel $\mathbf{F}(u, v)$ is weighted by its associated depth bin probabilities in $\mathbf{D}(u, v)$. It adds a new depth axis d_i, as shown in figure 1. The outer product can be used to weight feature pixels:

$$\mathbf{G}(u, v) = \mathbf{D}(u, v) \otimes \mathbf{F}(u, v) \tag{1}$$

where $\mathbf{D}(u, v)$ is the predicted depth distribution and $\mathbf{G}(u, v)$ is a matrix $D \times C$. The outer product is calculated for each pixel to generate frustum features $\mathbf{G} \in \mathbb{R}^{W_F \times H_F \times D \times C}$. The next steps are voxel transformation using the camera calibration matrix Reading et al. (2021) and collapsing to BEV.

For example, BEVFusion Liu et al. (2023) converts camera features into a point cloud, aggregates it with BEV pooling and flattens it along the z-axis. Such algorithms can be related to the Lift-Splat category Philion & Fidler (2020); Reading et al. (2021); Zhou et al. (2023).

1.5 Motivation

In Zhang et al. (2023b) authors made a detailed review of how autonomous vehicles perceive the environment under adverse weather conditions. They summarized the strengths and weaknesses of each sensor, as shown in the figure 2. Camera sensors are the most sensitive to environmental conditions, but not all parts of an image typically contain destructive information. Recent works Liu et al. (2023); Chen et al. (2023) have used a mid-level fusion approach to aggregate features from

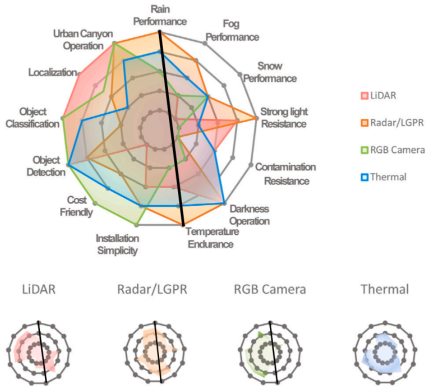


Figure 2: Sensor performance and characteristics Zhang et al. (2023b).

all modalities. Combining the representations of different modalities allows to solve perception problems in adverse weather conditions (see the Table 1).

Table 1: Sensor fusion and target weather conditions. ”L”, ”C” and ”R” represent LiDAR, Camera, and Radar modalities respectively.

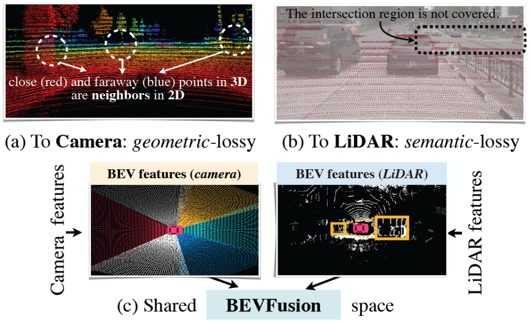
Sensor fusion	Configuration	Target weather
Bi-LRFusion (2023)	R + L	Fog
RadarNet (2020)	R + L	Rain
MVDNet (2021)	R + L	Fog
Liu (2021)	R + C	Rain, fog, nighttime
Rawashdeh (2021)	C + L + R	Snow
SLS-Fusion (2021)	L + C	Fog
Radecki (2016)	L + R + C	Wet conditions

In last time LiDARs and Radars sensors were significantly improved in terms of spatial resolution, accuracy, velocity measurement and resistance to adverse weather conditions Thi.

2 Related Work

FUTR3D. In Chen et al. (2023), every modality is encoded in its own coordinate. This framework does not assume any particular modalities and their model architectures. For this reason FUTR3D can work with any selected feature encoders. Researches used three types of data: LiDAR point cloud, Radar point cloud, and multi-view camera images. VoxelNet was used to encode LiDAR point clouds as multi-scale Bird's-eye view (BEV) feature maps $\left\{\mathcal{F}_{\text{lid}}^{j} \in \mathbb{R}^{C \times H_j \times W_j}\right\}_{j=1}^{m}$, where $H_i \times W_i$ is the size of the i-th BEV feature map, m is the count of feature maps. Radar points $\{r_j\}_{j=1}^{N} \in \mathbb{R}^{C_{ri}}$ are pillarized into $0.8\,\mathrm{m}$ pillars. Then MLP Φ_{rad} is used to achieve per-pillar features $\mathcal{F}_{\text{rad}}^{j} = \Phi_{\text{rad}}\left(r_j\right) \in \mathbb{R}^{C_{\text{ro}}}$, where C_{ro} is the number of encoded Radar features. In this way the Radar BEV feature map $\mathcal{F}_{\text{rad}} \in \mathbb{R}^{C_{\text{ro}} \times H \times W}$ is obtained. It is also assumed that there are N surrounding cameras installed in the car. It is supposed that each camera has taken m images. For image feature extraction ResNet is used. It outputs multi-scale features for each image, denoted as $\mathcal{F}_{\text{cam}}^{k} = \left\{\mathcal{F}_{\text{cam}}^{kj} \in \mathbb{R}^{C \times H_j \times W_j}\right\}_{j=1}^{m}$ for the k-th camera. So, after camera backbone there are m image feature maps for each camera. A transformer decoder uses queries to predict 3D bounding boxes. The predicted boxes can be repeatedly sent back into the transformer decoder and MAFS to refine the predictions.

BEVFusion. BEVFusion Liu et al. (2023) is the state-of-the-art fusion pipeline on the nuScenes and Waymo 3D object detection in 2022. BEVFusion performs sensor fusion in a shared BEV space and treats foreground and background, geometric and semantic information equally. There are fundamental differences in the modalities used to express data from various sensors: cameras capture data in perspective view and LiDAR in 3D view. For this reason, authors are looking for *a unified representation* that is suitable for multi-task multi-modal feature fusion. Camera and LiDAR features have drastically different densities. The camera-to-LiDAR projection is semantically lossy, and the LiDAR-to-camera projection creates significant geometric distortion. In this paper, authors propose BEVFusion to unify multi-modal features in a shared bird's-eye view (BEV) representation space for different tasks (see figure 3). The transformation to BEV saves both geometric structure (from LiDAR features) and semantic density (from camera features).


close (red) and faraway (blue) points in 3D are neighbors in 2D
The intersection region is not covered.
BEV features (camera)
BEV features (LiDAR)
Camera features
LiDAR features

(a) To **Camera**: *geometric*-lossy (b) To **LiDAR**: *semantic*-lossy

(c) Shared **BEVFusion** space

Figure 3: BEVFusion unifies camera and LiDAR features in a shared BEV space instead of mapping one modality to the other Liu et al. (2023).

Bi-LRFusion. Radar provides longer detection range than LiDAR, which is essential on highways and expressways. But LiDAR is better at capturing the object's 3D shape. The existing feature-level Radar fusion methods commonly ignore the problems caused by the lack of height information and extreme sparsity of Radar data. Specifically, taking the data from the nuScenes dataset as an example, the 32-beam LiDAR sensor produces approximately 30,000 points, while the Radar sensor only captures about 200 points for the same scene. The height values of the Radar points are simply set as the ego Radar sensor's height. All height values are transformed to the LiDAR coordinate system, but this is not consistent for objects with different heights.

In this work authors introduced a bi-directional LiDAR-Radar fusion framework, termed Bi-LRFusion. To fully utilize the advantages of combining LiDAR and Radar, the authors enhance the Radar features in two directions with the help of LiDAR data. It makes Radar features more powerful before the final BEV fusion step. Bi-LRFusion first encodes BEV features for each modality individually. Then, LiDAR-to-Radar (L2R) fusion is proposed to use to enhance the extremely sparse Radar features (see figure 4). This module is focusing on the height information that is completely missing in the Radar data and the local BEV features that are scarce in the Radar data. L2R module consists of two feature fusion blocks: the query-based L2R height feature fusion and the query-based L2R BEV feature fusion, in which they generate the pseudo height features and the pseudo local BEV features, respectively. The grouped LiDAR raw points are aggregated to formulate pseudo-Radar height features, and the grouped LiDAR BEV features are aggregated to produce pseudo-Radar BEV features. Specifically, for each nonempty grid cell on the Radar feature map, they query and group the nearby LiDAR data (including both raw points and BEV features) to obtain more detailed Radar features. Further generated pseudo-Radar height and BEV features are fused to the Radar BEV features through concatenation.

3 Method

Since both LiDARs and Radars operate in 3D space and they are more reliable than cameras under adverse environmental conditions, we first do their prefusion Yingjie Wang et al. (2023). Our pipeline inherits the advantage of the Lidar-to-Radar prefusion mechanism of Bi-LRFusion. We use a specific transformation for a particular sensor to represent the extracted feature in the BEV.

Encoding of LiDAR Features. This process consists of the following steps: voxelization of LiDAR points; taking all points in the same voxel as input and using a multi - layer perception (MLP) to

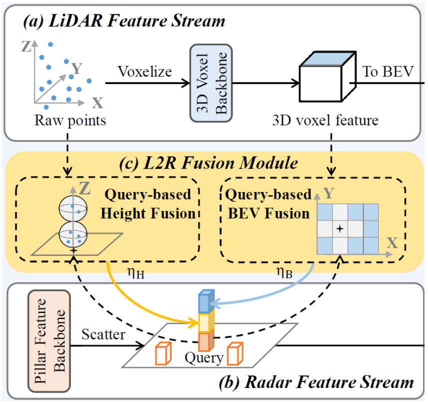


Figure 4: Part of the Bi-LRFusion pipeline – LiDAR-to-Radar (L2R) fusion Yingjie Wang et al. (2023).

extract pointwise features; max pooling to obtain the locally aggregated features for each voxel; 3D Voxel Backbone composed of 3D sparse convolutional layers and 3D sub-manifold convolutional layers Yan et al. (2018); producing a LiDAR BEV feature map by stacking volume features along the Z-axis;

Radar Feature Encoding. By utilizing the Pillar Feature Backbone Lang et al. (2019), the Radar point cloud is converted into a series of pillars.

Camera Feature Encoding. As in BEVFusion, for each image pixel we predict the discrete depth distribution. It forms frustum features (see figure 1) – camera feature point cloud of size $NHWD$, where N is the number of cameras, (H, W) is the size of camera feature map. Then we use BEV pooling operation to flat the features along z-axis.

The figures 5 and 6 illustrate the concept of our pipeline. The LiDAR-to-Radar step enriches the Radar features similar to Yingjie Wang et al. (2023).

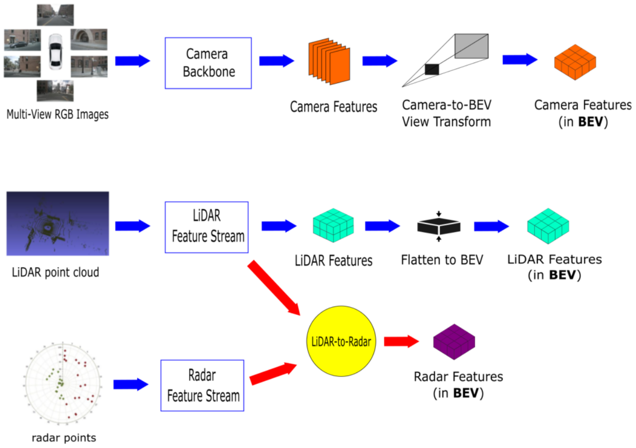


Figure 5: This is the first part of the pipeline. Transformation of raw LiDAR, Radar points, and images into a BEV representation.

Because BEV features can be spatially misaligned, we use a BEV encoder consisting of several convolutions and residual blocks (see figure 6). As a BEVFusion, this pipeline can be used for different tasks such as segmentation and 3D object detection.

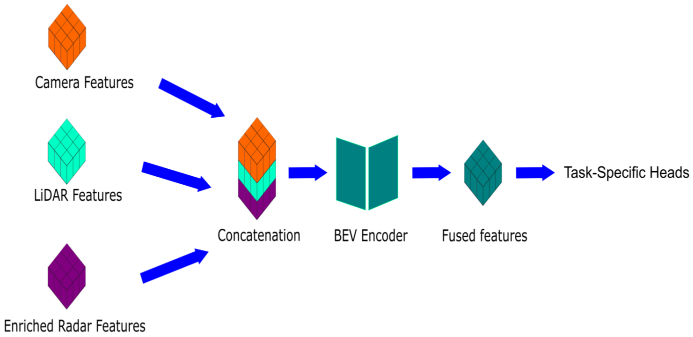


Figure 6: We concatenate all the BEV representations together, encode the result and then send it to specific heads.

4 Experiments

The Nuscenes Dataset is widely used dataset for vision-centric perception with six calibrated cameras covering a 360-degree horizontal FOV, 1 LiDAR and 5 Radars. The camera image resolution is 1600×900. Nuscenes consists of 1000 scenes, each one of them is 20 seconds long. 850 scenes are for training/validation and 150 for testing.

The most commonly used criterion for BEV Detection is average precision (AP) and the mean average precision (mAP) over different classes. The Average Precision (AP) metric is extended from 2D to the 3D space:

$$AP = \int_0^1 \max \left\{ p\left(r' \mid r' \geq r\right) \right\} dr \tag{3}$$

where $p(r)$ is the precision-recall curve. The difference between 2D AP and 3D AP is the matching criteria between ground truth and predictions when calculating precision and recall.

Instead of IoU to select TP, NuScenes proposes AP_{center} where a predicted object is matched to a ground truth object if the distance of their center locations on the ground (BEV) plane is below a certain threshold d. The AP_{center} is calculated under different distance thresholds: $\mathbb{D} = \{0.5, 1, 2, 4\}$ meters. The mAP is computed by averaging the AP_{center} over all matching thresholds and all classes $\mathbb{C}$: $\mathrm{mAP} = \frac{1}{|\mathbb{C}||\mathbb{D}|} \sum_{c \in \mathbb{C}} \sum_{d \in \mathbb{D}} \mathrm{AP}_{c,d}$. NuScenes Detection Score (NDS) is further proposed to take both AP_{center} and the error of other parameters, i.e. size, heading, velocity, into consideration.

In our experiments we compared BEVFusion Liu et al. (2023) and BiFusion Yingjie Wang et al. (2023) with our method (see the Table 2).

Table 2: Results of expirements. ”L”, ”C” and ”R” represent LiDAR, Camera, and Radar modalities.

Model	mAP	NDS
Bi-LRFusion (R + L)	62.3	65.54
BEVFusion (C + L)	68.57	71.40
CBILR (C + R + L)	**71.09**	**73.36**

Experiments show that it is important to use all modalities in a clever way. Combining different modalities helps to overcome the limitations of individual sensors.

5 Conclusion

This work has demonstrated CBILR, a promising multi-sensor fusion framework that aims to improve perception robustness for autonomous vehicles. It has addressed the critical challenge of limited sensor performance in adverse weather conditions, a significant hurdle on the path to achieving truly autonomous navigation. CBILR aims to overcome the limitations of existing fusion methods by using the Bi-LRFusion module. This module promotes a mutually beneficial LiDAR/radar relationship, allowing each to benefit from the other's strengths.

The experiments show that using multiple sensors for fusion increases reliability in challenging weather conditions. Previous works uniformly combine all sensors together. They do not consider the weaknesses of different sensors. By utilizing Bi-LRFusion and promoting a thorough understanding of the environment, CBILR strives to lead the way into a new era of strong and adaptable perception. This effort aims to bring autonomous vehicles closer to the ultimate goal of safe and reliable operation in all conditions.

REFERENCES

4d lidars vs 4d radars: Why the lidar vs radar comparison is more relevant today than ever. `https://www.thinkautonomous.ai/blog/fmcw-lidars-vs-imaging-radars/`. Accessed: 2024-02-21.

Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. Futr3d: A unified sensor fusion framework for 3d detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 172–181, 2023.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9223–9232, 2023.

Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12697–12705, 2019.

Hongyang Li, Chonghao Sima, Jifeng Dai, Wenhai Wang, Lewei Lu, Huijie Wang, Jia Zeng, Zhiqi Li, Jiazhi Yang, Hanming Deng, et al. Delving into the devils of bird's-eye-view perception: A review, evaluation and recipe. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023a.

Huadong Li, Minhao Jing, Jiajun Liang, Haoqiang Fan, and Rehne Ji. Sparse beats dense: Rethinking supervision in radar-camera depth completion. *arXiv:2312.00844*, 2023b.

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 *AAAI Technical Track on Computer Vision II*, volume 37, pp. 1477–1485, 2023c.

Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. *Advances in Neural Information Processing Systems*, 35:18442–18455, 2022.

Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 641–656, 2018.

Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. *Advances in Neural Information Processing Systems*, 35:10421–10434, 2022.

Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pp. 2774–2781. IEEE, 2023.

Xinzhu Ma, Zhihui Wang, Haojie Li, Pengbo Zhang, Wanli Ouyang, and Xin Fan. Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6851–6860, 2019.

Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pp. 311–327. Springer, 2020.

Yuexin Ma, Tai Wang, Xuyang Bai, Huitong Yang, Yuenan Hou, Yaming Wang, Yu Qiao, Ruigang Yang, Dinesh Manocha, and Xinge Zhu. Vision-centric bev perception: A survey. *arXiv preprint arXiv:2208.02797*, 2022.

Hanspeter A Mallot, Heinrich H Bülthoff, JJ Little, and Stefan Bohrer. Inverse perspective mapping simplifies optical flow computation and obstacle detection. *Biological cybernetics*, 64(3):177–185, 1991.

Ramin Nabati and Hairong Qi. Centerfusion: Center-based radar and camera fusion for 3d object detection. *arXiv preprint arXiv:2011.04841*, 2020.

Ramin Nabati, Landon Harris, and Hairong Qi. Cftrack: Center-based radar and camera fusion for 3d multi-object. *arXiv preprint arXiv:2107.05150*, 2021.

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*, pp. 194–210. Springer, 2020.

Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8555–8564, 2021.

Thomas Roddick. *Learning Birds-Eye View Representations for Autonomous Driving*. PhD thesis, 2021.

Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8445–8453, 2019.

Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

Jiajun Yingjie Wang, Deng, Yao Li, Jinshui Hu, Cong Liu, Yu Zhang, Jianmin Ji, Wanli Ouyang, and Yanyong Zhang. Bi-lrfusion: Bi-directional lidar-radar fusion for 3d dynamic object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Yurong You, Yan Wang, Wei-Lun Chao, Divyansh Garg, Geoff Pleiss, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving. *arXiv preprint arXiv:1906.06310*, 2019.

Ce Zhang, Chengjie Zhang, Yiluan Guo, Lingji Chen, and Michael Happold. Motiontrack: end-to-end transformer-based multi-object tracking with lidar-camera fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 151–160, 2023a.

Tianyuan Zhang, Xuanyao Chen, Yue Wang, Yilun Wang, and Hang Zhao. Mutr3d: A multi-camera tracking framework via 3d-to-2d queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4537–4546, 2022.

Yuxiao Zhang, Alexander Carballo, Hanting Yang, and Kazuya Takeda. Perception and sensing for autonomous vehicles under adverse weather conditions: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:146–177, 2023b.

Huazan Zhong, Hao Wang, Zhengrong Wu, Chen Zhang, Yongwei Zheng, and Tao Tang. A survey of lidar and camera fusion enhancement. In *10th International Conference of Information and Communication Technology*, 2021.

Hongyu Zhou, Zheng Ge, Zeming Li, and Xiangyu Zhang. Matrixvt: Efficient multi-camera to bev transformation for 3d perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8548–8557, 2023.

Zijian Zhu, Yichi Zhang, Hai Chen, Yinpeng Dong, Shu Zhao, Wenbo Ding, Jiachen Zhong, and Shibao Zheng. Understanding the robustness of 3d object detection with bird's-eye-view representations in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21600–21610, 2023.