X-Align: Cross-Modal Cross-View Alignment for Bird's-Eye-View Segmentation

Shubhankar Borse [*] Marvin Klingner [†] Varun Ravi Kumar [‡] Hong Cai [*]
Abdulaziz Almuzairee [*§] Senthil Yogamani [¶] Fatih Porikli [*]
{sborse, mklingne, vravikum, hongcai, almuzair, syogaman, fporikli}@qti.qualcomm.com

Abstract

Bird's-eye-view (BEV) grid is a typical representation of the perception of road components, e.g., drivable area, in autonomous driving. Most existing approaches rely on cameras only to perform segmentation in BEV space, which is fundamentally constrained by the absence of reliable depth information. The latest works leverage both camera and LiDAR modalities but suboptimally fuse their features using simple, concatenation-based mechanisms.

In this paper, we address these problems by enhancing the alignment of the unimodal features in order to aid feature fusion, as well as enhancing the alignment between the cameras' perspective view (PV) and BEV representations. We propose X-Align, a novel end-to-end cross-modal and cross-view learning framework for BEV segmentation consisting of the following components: (i) a novel Cross-Modal Feature Alignment (X-FA) loss, (ii) an attention-based Cross-Modal Feature Fusion (X-FF) module to align multi-modal BEV features implicitly, and (iii) an auxiliary PV segmentation branch with Cross-View Segmentation Alignment (X-SA) losses to improve the PV-to-BEV transformation. We evaluate our proposed method across two commonly used benchmark datasets, i.e., nuScenes and KITTI-360. Notably, X-Align significantly outperforms the state-of-the-art by 3 absolute mIoU points on nuScenes. We also provide extensive ablation studies to demonstrate the effectiveness of the individual components.

1. Introduction

Bird's-eye-view (BEV) segmentation aims at classifying each cell in a BEV grid around the ego position. As such, it is a key enabler for applications like autonomous driving and robotics. For instance, the BEV segmentation map is a prerequisite for current works on behavior prediction and trajectory planning [17, 47, 54]. It is also an important input modality for learning end-to-end controls (*e.g.*, speed control, steering) in autonomous driving [9].

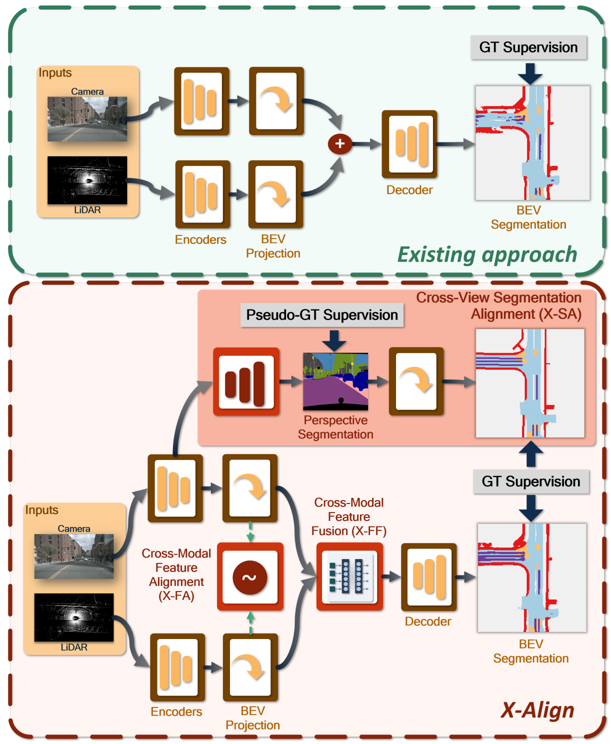

Figure 1: Existing methods for cross-modal BEV segmentation (top) utilize simple concatenation-based fusion (*e.g.*, [33]), while our proposed **X-Align enforces cross-modal feature alignment together with attention-based feature fusion, as well as cross-view segmentation alignment** (bottom). These contributions improve both feature aggregation and PV-to-BEV transformation, leading to more accurate BEV segmentation.

Given the ubiquity of camera sensors, existing BEV segmentation methods predominantly focus on predicting BEV segmentation maps from camera images [39, 55, 56, 64]. However, the lack of reliable 3D information significantly limits the performance of these methods. A possible way to resolve this challenge is to leverage a LiDAR sensor and fuse the measured sparse geometric information with that contained in camera images. While camera-LiDAR fusion has been extensively studied for the task of 3D object detection, such fusion strategies are relatively unexplored for

[*]Qualcomm AI Research, an initiative of Qualcomm Technologies, Inc.
[†]Automated Driving, Qualcomm Technologies International GmbH
[‡]Automated Driving, Qualcomm Technologies, Inc.
[§]University of California, San Diego. Work done at Qualcomm.
[¶]Automated Driving, QT Technologies Ireland Limited

BEV segmentation. The concurrent work of [33] provides the first baseline using a simple concatenation of LiDAR BEV features and camera features projected from perspective view (PV) to BEV via estimated depth and voxel pooling. However, the PV-to-BEV projection can be inaccurate due to errors in depth estimation. As a result, in the concatenation stage, the network may aggregate poorly aligned features across the camera and the LiDAR branches, resulting in suboptimal fusion results.

In this paper, we propose a novel *cross-modal, cross-view alignment strategy, X-Align*, which enforces feature alignment across features extracted from the camera and LiDAR inputs as well as segmentation consistency across PV and BEV to improve the overall BEV segmentation accuracy (cf. Fig. 1). First, we propose a Cross-Modal Feature Alignment (X-FA) loss function that promotes the correlation between projected camera features and LiDAR features, as measured by cosine similarity. In addition, we incorporate attention to the Cross-Modal Feature Fusion (X-FF) of these two sets of modality-specific features instead of using simple concatenation as in [33]. This gives the network a more substantial capability to properly align and aggregate features from the two sensing modalities.

We further impose Cross-View Segmentation Alignment (X-SA) losses during training. We introduce a trainable segmentation decoder based on the intermediate PV camera features to generate PV segmentation. Next, we utilize the same PV-to-BEV transformation [39] that converts PV camera features to BEV to convert the PV segmentation map into a BEV segmentation map, which is then supervised by the ground truth. We also supervise the intermediate PV segmentation map using pseudo-labels generated by a high-quality, off-the-shelf semantic segmentation model. In this way, the camera branch learns to derive intermediate features containing useful PV semantic features, providing richer information for BEV segmentation after being projected to BEV space. Moreover, this provides additional supervision on the PV-to-BEV module, allowing it to learn a more accurate transformation.

Our main contributions are summarized as follows:

- We propose a novel framework, X-Align, that enables better feature alignment and fusion across camera and LiDAR modalities and enforces segmentation alignment across perspective view and bird's eye view.
- Specifically, we propose a Cross-Modal Feature Alignment (X-FA) loss to enhance the correlation between the camera and LiDAR features. We also devise an attention-based Cross-Modal Feature Fusion (X-FF).
- We further propose to enforce Cross-View Segmentation Alignment (X-SA) across the perspective view and bird's eye view, which encourages the model to learn richer semantic features and a more accurate PV-to-BEV projection.
- We conduct extensive experiments on the nuScenes and KITTI-360 datasets with comprehensive ablation studies that demonstrate the efficacy of X-Align. In particular, on nuScenes, we surpass the state-of-the-art in BEV segmentation by 3 absolute mIoU points.

2. Related Work

BEV Segmentation: The task of BEV segmentation has mostly been explored using (multiple) camera images as input. Building on top of Perspective View (PV) segmentation [2–4, 18, 61, 62], early works used the homography transformation to convert camera images to BEV, subsequently estimating the segmentation map [13, 34, 50, 65]. As the homography transformation introduces strong artifacts, subsequent works moved towards depth estimation and voxelization [39, 42] for the PV-to-BEV transformation as end-to-end learning [35, 41]. This basic setup has been further explored in various directions: VPN [38] explores domain adaptation, BEVerse [63] and M^{2}BEV [55] explore multi-task learning with 3D object detection, CoBEVT [56] explores fusion of features from vehicles, Gosala *et al*. explore panoptic BEV segmentation [15], while several works explore incorporation of temporal context [17, 44]. Furthermore, CVT [64] uses a learned map embedding and an attention mechanism between map queries and camera features. In contrast to these existing BEV segmentation approaches that only use camera images, we explore the multimodal fusion of LiDAR point clouds and camera images.

Multi-modal fusion for BEV segmentation has been enabled by recently introduced large-scale datasets providing time-synchronized data from multiple sensors [5, 14, 46]. However, most works on these datasets focus on the 3D object detection task [1, 10, 24, 33, 45, 59], while we focus on BEV segmentation. The closest prior art to our work is BEVFusion [33]. While their method also predicts BEV segmentation based on LiDAR point clouds and camera images, they use a simple feature concatenation to fuse multimodal features such that the network implicitly has to connect information from misaligned features. In contrast, we explicitly enforce alignment between multi-modal features. Also, we enforce alignment between PV and BEV segmentation to improve the PV-to-BEV transformation.

Camera-LiDAR Sensor Fusion: The vast majority of fusion methods have been proposed for the 3D object detection task. Initially, two-stage approaches have been proposed, lifting image bounding box proposals into 3D frustum view [37, 40, 53] for fusion with LiDAR. However, research focus has shifted towards end-to-end training, where approaches can roughly be divided into three categories: point-/input-level decoration, feature-level fusion, and proposal-level fusion. Point-level fusion includes methods such as PointAugmenting [51], PointPainting [49], FusionPainting [57], AutoAlign [8], and MVP [60], which extract camera features and use these features to enrich the point-level information, which is subsequently processed by a LiDAR-based detector. Recently proposed FocalSparseCNNs [7] similarly enriches the features in the early feature extraction stage. For proposal-level fusion usually the predicted bounding boxes are refined [27], often making use of an attention mechanism such as in FUTR3D [6] and TransFusion [1]. However, these two fusion types have downsides regarding generalization. While proposal-level fusion is not easily generalizable to other tasks, input-level decoration is not generically extendable to other sensor modalities.

Feature-level fusion aims at fusing extracted features from different sensors, subsequently predicting outputs for one or several tasks [24–26, 33]. Our approach also falls into this category. While [26, 33] use fusion via concatenation, more recent approaches apply attention-based fusion [24, 25]. However, these approaches still try to implicitly learn the interconnection between cross-modal features, while we explicitly encourage alignment between features from different modalities. Also, none of the above-mentioned methods uses attention-based cross-modal fusion for BEV segmentation or output-level segmentation alignment of PV and BEV segmentation.

3. Proposed X-Align Framework

This section describes X-Align, our novel cross-modal and cross-view alignment strategy. We first formally introduce the problem and describe a baseline method in Section 3.1. We provide an overview of X-Align in Section 3.2 and then discuss its components in detail, including Cross-Modal Feature Fusion (X-FF), Cross-Modal Feature Alignment (X-FA), and Cross-View Segmentation Alignment (X-SA) in Sections 3.3, 3.4, 3.5, respectively.

3.1. Problem Formulation and Baseline

Our goal is to develop a framework taking multi-modal sensor data $\mathcal{X}$ as input and predicting a BEV segmentation map $\hat{\boldsymbol{m}} \in \mathcal{S}^{H^{\mathrm{BEV}} \times W^{\mathrm{BEV}}}$, with resolution $H^{\mathrm{BEV}} \times W^{\mathrm{BEV}}$, and the set of classes $\mathcal{S} = \{0, 1, \dots, |\mathcal{S}|\}$. As illustrated in Fig. 2, the set of inputs, $\mathcal{X} = \{\boldsymbol{x}, \boldsymbol{l}\}$, contains RGB camera images in PV, $\boldsymbol{x} \in \mathbb{R}^{N^{\mathrm{cam}} \times H^{\mathrm{cam}} \times W^{\mathrm{cam}} \times 3}$, where N^{cam}, H^{cam}, W^{cam} denote number of cameras, image height, and image width, respectively, as well as a LiDAR point cloud, $\boldsymbol{l} \in \mathbb{R}^{P \times 5}$, with number of points P. Each point consists of its 3-dimensional coordinates, reflectivity, and ring index.

Baseline Method: We first establish a baseline method of fusion-based BEV segmentation based on BEVFusion [33]. As shown in Fig. 2, initial features are extracted from both sensor inputs. For camera images, a camera encoder, $\boldsymbol{E}^{\mathrm{cam}}$, extracts features in PV, $\boldsymbol{f}^{\mathrm{cam}}$. Subsequently, we use a feature pyramid network (FPN) and a PV-to-BEV transformation based on [39] to obtain camera features in BEV space, following BEVDet [19]. For the LiDAR points, we follow SECOND [58] in using voxelization and a sparse LiDAR encoder, $\boldsymbol{E}^{\mathrm{LiDAR}}$. The LiDAR features are projected to BEV space using a flattening operation as in [33].

These operations result in two sets of modality-specific BEV features, $\tilde{\boldsymbol{f}}^{\mathrm{cam}} \in \mathbb{R}^{H^{\mathrm{lat}} \times W^{\mathrm{lat}} \times C^{\mathrm{cam}}}$ and $\tilde{\boldsymbol{f}}^{\mathrm{LiDAR}} \in \mathbb{R}^{H^{\mathrm{lat}} \times W^{\mathrm{lat}} \times C^{\mathrm{LiDAR}}}$, with BEV space feature resolution $H^{\mathrm{lat}} \times W^{\mathrm{lat}}$ and number of channels C^{cam} and C^{LiDAR} for camera and LiDAR features, respectively. The features are then combined (*e.g.*, by simple concatenation as in [16, 23, 33]), resulting in fused features $\tilde{\boldsymbol{f}}^{\mathrm{fused}} \in \mathbb{R}^{H^{\mathrm{lat}} \times W^{\mathrm{lat}} \times C^{\mathrm{fused}}}$, which are further processed by a BEV Encoder and FPN as in SECOND [58]. Finally, the features are processed by a segmentation head with the same architecture as in [33] to ensure comparability. This baseline model is trained using the focal cross-entropy loss [30]:

$$\mathcal{L}^{\mathrm{BEV}} = \mathrm{FocalCE}\left(\hat{\boldsymbol{y}}^{\mathrm{BEV}}, \boldsymbol{y}^{\mathrm{BEV}}\right), \tag{1}$$

where $\hat{\boldsymbol{y}}^{\mathrm{BEV}} \in \mathbb{R}^{H^{\mathrm{BEV}} \times W^{\mathrm{BEV}} \times S}$ are class probabilities and $\boldsymbol{y}^{\mathrm{BEV}} \in \{0, 1\}^{H^{\mathrm{BEV}} \times W^{\mathrm{BEV}} \times S}$ denotes the one-hot encoded ground truth. We can obtain the final classes $\hat{\boldsymbol{m}}$ by a pixel-wise argmax operation on $\hat{\boldsymbol{y}}$ during inference.

3.2. X-Align Overview

Building on top of the previously described baseline, we present our novel cross-modal and cross-view alignment strategy, X-Align (highlighted by red boxes in Fig. 2). First, we improve the simple concatenation-based fusion with a Cross-Modal Feature Fusion (X-FF) module, which leverages attention and mitigates misalignment between features across modalities (Section 3.3). Secondly, we propose a Cross-Modal Feature Alignment (X-FA) loss, $\mathcal{L}^{\text{X-FA}}$, which promotes the correlation between features across modalities (Section 3.4). Finally, we propose losses to enforce Cross-View Segmentation Alignment (X-SA) in Section 3.5, where we supervise a PV segmentation predicted from the intermediate camera features with a loss $\mathcal{L}^{\mathrm{PV}}$ and the PV-to-BEV-projected version of this segmentation with a loss $\mathcal{L}^{\mathrm{PV2BEV}}$. These two losses provide more direct training signals to the PV-to-BEV transformation and encourage richer semantic features in PV before the transformation. Overall, our total optimization objective is

$$\mathcal{L}^{\text{X-Align}} = \lambda_1 \mathcal{L}^{\mathrm{BEV}} + \lambda_2 \mathcal{L}^{\text{X-FA}} + \lambda_3 \mathcal{L}^{\mathrm{PV}} + \lambda_4 \mathcal{L}^{\mathrm{PV2BEV}}, \tag{2}$$

where λ_i, $i \in \{1, 2, 3, 4\}$ are the loss weighting factors.

3.3. Cross-Modal Feature Fusion (X-FF)

In recent BEV segmentation literature [16, 23, 33], it is a common approach to utilize concatenation followed by convolution to combine features from multi-modal inputs, $\tilde{\boldsymbol{f}}^{\mathrm{cam}}$ and $\tilde{\boldsymbol{f}}^{\mathrm{LiDAR}}$, resulting in the aggregated features $\tilde{\boldsymbol{f}}^{\mathrm{fused}}$. However, the lack of reliable depth informa-

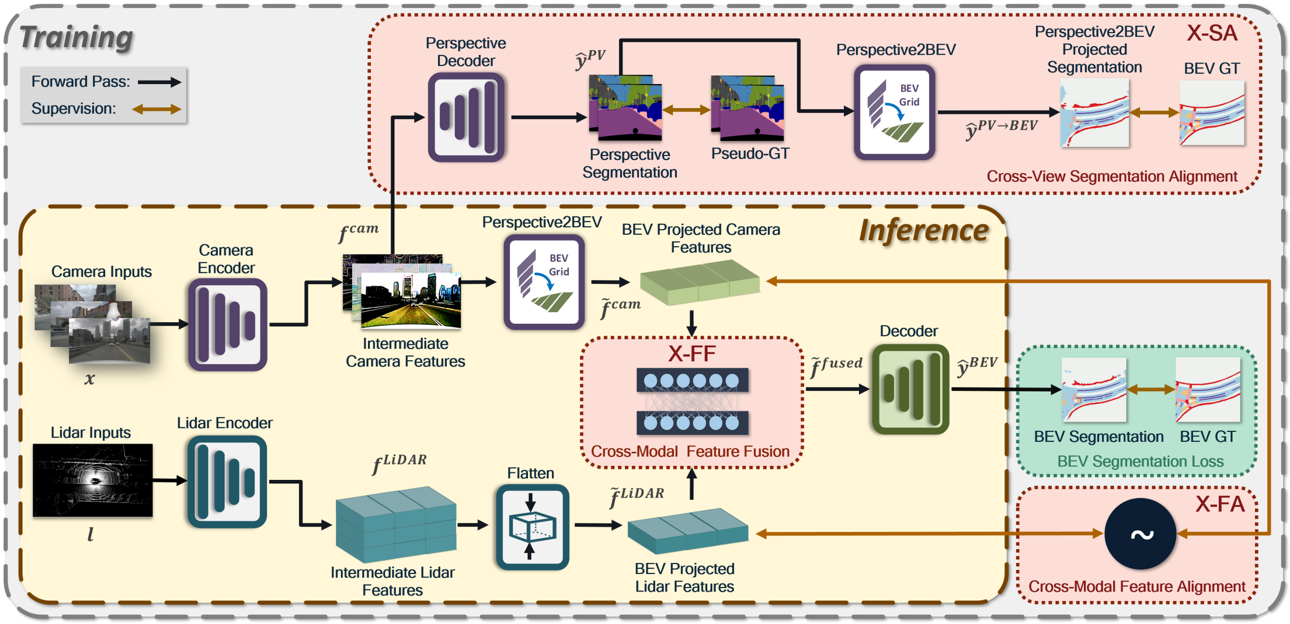

Figure 2: **Our proposed X-Align framework**: We present a cross-modal and cross-view alignment algorithm for the task of BEV segmentation based on camera images and LiDAR point clouds. We apply the Cross-View Segmentation Alignment (X-SA) and Cross-Modal Feature Alignment (X-FA) losses during training. We also propose a Cross-Modal Feature Fusion (X-FF) module to correct pixel inconsistencies between multi-modal features. Our proposed contributions are highlighted in red. During inference, we can remove the blocks which solely contribute to computing loss functions, suggesting that the performance enhancement comes with no added inference cost.

tion can cause inaccurate PV-to-BEV transformation of features, which subsequently results in suboptimal alignment and fusion of multi-modal features. The convolution blocks utilized in existing approaches [16, 23, 33] cannot rectify such misalignment due to their translation invariance. To address this issue, we propose more powerful, Cross-Modal Feature Fusion (X-FF) modules that can account for pixel-wise misalignment, as shown in Fig. 3. Next, we describe in detail our three proposed fusion designs.

Self-Attention: Our proposed X-FF using self-attention is shown in Fig. 3 (left). We first stack features $\tilde{\boldsymbol{f}}^{\text{cam}} \in \mathbb{R}^{H^{\text{lat}}\times W^{\text{lat}}\times C^{\text{cam}}}$ and $\tilde{\boldsymbol{f}}^{\text{LiDAR}} \in \mathbb{R}^{H^{\text{lat}}\times W^{\text{lat}}\times C^{\text{LiDAR}}}$, and tokenize them into $K \times K$ patches with an embedding dimension of L^{embed}. These patches are fed into a multi-head self-attention module [48]. The output is then projected back to the original resolution using a deconvolution block, resulting in the final fused features $\tilde{\boldsymbol{f}}^{\text{fused}} \in \mathbb{R}^{H^{\text{lat}}\times W^{\text{lat}}\times C^{\text{fused}}}$. By using self-attention, our proposed module can correspond to the camera and LiDAR features spatially, accounting for potential misalignment.

Spatial-Channel Attention: In this option, we leverage the recently proposed Split-Depth Transpose Attention (SDTA) [36], as shown in Fig. 3 (middle). It first performs spatial and channel mixing of the stacked camera and LiDAR features via depth-wise and point-wise convolutions. Then, it applies channel attention followed by a lightweight MLP. The output is passed through a deconvolution block to generate the fused features $\tilde{\boldsymbol{f}}^{\text{fused}}$. Spatial and channel mixing together with channel attention provides powerful capacity for the module to better address the misalignment between the camera and LiDAR features.

Pose-Driven Deformable Convolution: This design is illustrated in Fig. 3 (right). We know that the transformation between modalities is a function of their relative poses to the ego vehicle. Hence, we apply an adaptive transformation, *i.e.*, Deformable Convolution (DCNv2) [66], to the stacked multi-modal features $\tilde{\boldsymbol{f}}^{\text{cam}}$ and $\tilde{\boldsymbol{f}}^{\text{LiDAR}}$, which can implicitly learn the cross-modal alignment based on available pose information. More specifically, we process the pose matrices with an MLP to generate a pose embedding $\tilde{\boldsymbol{f}}^{\text{pose}} \in \mathbb{R}^{H^{\text{lat}}\times W^{\text{lat}}\times C^{\text{pose}}}$, which is then concatenated with $\tilde{\boldsymbol{f}}^{\text{cam}}$ and $\tilde{\boldsymbol{f}}^{\text{LiDAR}}$. They are used to generate $K \times K$ offset vectors to be used by the DCNv2 block, which produces the output fused features $\tilde{\boldsymbol{f}}^{\text{fused}}$.

Our proposed X-FF designs provide the network with the suitable capacity to properly align and fuse multi-modal features. While they introduce additional computations, they show more superior accuracy-efficiency trade-offs as compared to naively increasing the complexity of the baseline network, as we shall see in our study in Section 4.3.

3.4. Cross-Modal Feature Alignment (X-FA)

While our proposed X-FF modules can improve feature alignment, they incur additional computations, which may not always be feasible. As such, we propose a second measure to improve feature alignment with a Cross-Modal Fea-

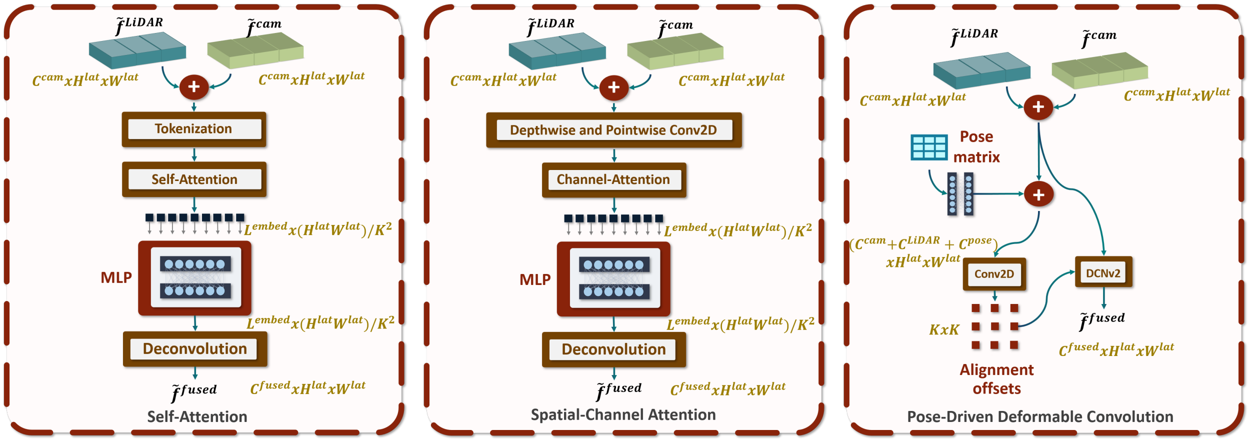

Figure 3: Our three **proposed Cross-Modal Feature Fusion (X-FF) strategies**, using standard self-attention (left), spatial-channel attention (middle), and pose-driven deformable convolution (right), respectively.

ture Alignment (X-FA) loss $\mathcal{L}^{\text{X-FA}}$ that is only applied during training and does not introduce additional computations for inference. It can also be used in conjunction with X-FF.

Consider extracted features in BEV space, $\tilde{\boldsymbol{f}}^{\text{cam}}$ and $\tilde{\boldsymbol{f}}^{\text{LiDAR}}$, from camera and LiDAR branches, respectively. We promote the correlation between the two sets of features by imposing a cosine similarity loss between them:

$$\mathcal{L}^{\text{X-FA}} = \text{CosineSim}\left(\tilde{\boldsymbol{f}}^{\text{cam}}, \tilde{\boldsymbol{f}}^{\text{LiDAR}}\right). \tag{3}$$

In order to apply this loss, there are two requirements. First, the camera features $\tilde{\boldsymbol{f}}^{\text{cam}}$ and LiDAR features $\tilde{\boldsymbol{f}}^{\text{LiDAR}}$ in BEV space need to have the same resolution. In this work, we ensure that the network parameters are chosen accordingly. In case both features have different resolutions, differentiable grid sampling [20] can be used. Second, the channels C^{cam} and C^{LiDAR} should match for Eq. (3) to be applicable. This, however, is generally not the case. As such, we take all features from the lower-dimensional branch and enforce their similarity to several subsets of features from the higher-dimensional branch.

3.5. Cross-View Segmentation Alignment (X-SA)

In addition to encouraging feature alignment, we also impose alignment at the output segmentation level across PV and BEV using our Cross-Modal Segmentation Alignment (X-SA) losses. More specifically, for the camera branch, we take intermediate PV features and feed them through an additional decoder to generate a PV segmentation prediction $\hat{\boldsymbol{y}}^{\text{PV}} \in \mathbb{R}^{N^{\text{cam}} \times H^{\text{cam}} \times W^{\text{cam}} \times |\mathcal{S}|}$ (cf. Fig. 2, top part). We further transform $\hat{\boldsymbol{y}}^{\text{PV}}$ to BEV space by utilizing the PV-to-BEV transformation as for the features, resulting in a projected BEV segmentation map $\hat{\boldsymbol{y}}^{\text{PV}\rightarrow\text{BEV}} \in \mathbb{R}^{H^{\text{BEV}} \times W^{\text{BEV}} \times |\mathcal{S}|}$. The projected BEV segmentation map is supervised w.r.t ground-truth BEV segmentation using a focal cross-entropy loss:

$$\mathcal{L}^{\text{PV2BEV}} = \text{FocalCE}\left(\hat{\boldsymbol{y}}^{\text{PV}\rightarrow\text{BEV}}, \boldsymbol{y}^{\text{BEV}}\right). \tag{4}$$

As for the PV segmentation, since PV ground truth is not always available on BEV perception datasets, we supervise it with a focal loss using pseudo-labels generated by a state-of-the-art model pretrained on Cityscapes [12], as follows:

$$\mathcal{L}^{\text{PV}} = \text{FocalCE}\left(\hat{\boldsymbol{y}}^{\text{PV}}, \boldsymbol{y}^{\text{PV}}\right), \tag{5}$$

By introducing these two additional supervisions, we enforce that across PV and BEV, the segmentations are accurate and aligned through the PV-to-BEV transformation. This benefit is two-fold: First, the module used here is given by the same PV-to-BEV transformation as on the feature level in the main camera branch. Our X-SA loss provides additional supervision to more accurately train this key module. Second, imposing a PV segmentation loss $\mathcal{L}^{\text{PV}}$ encourages the network to learn useful PV semantic features, providing richer semantic information for the downstream BEV features. Our X-SA components, including the additional decoder and the losses, are only used during training and do not require overhead at test time.

In summary, our complete X-Align framework **X-Align**$_{\text{all}}$ proposes four additions to the baseline: the X-FF feature fusion module, along with three additional training losses: $\mathcal{L}^{\text{X-FA}}$, $\mathcal{L}^{\text{PV}}$, and $\mathcal{L}^{\text{PV2BEV}}$. In cases where the network only takes camera inputs, we apply the two X-SA losses, $\mathcal{L}^{\text{PV}}$ and $\mathcal{L}^{\text{PV2BEV}}$, giving us the **X-Align**$_{\text{view}}$ variant. In addition, in case extra computation is not allowed, we apply all three X-Align losses when training the network, forming the **X-Align**$_{\text{losses}}$ variant. We extensively evaluate these variants as well as combinations of our proposed X-Align components in Section 4.3.

4. Experiments

In this section, we present comprehensive performance evaluations of X-Align and compare it with baselines and the current state of the art. We further conduct extensive ablation studies on various aspects of our proposed approach.

Model	*Backbone*	*Modality*	*Drivable*	*Ped. Cross.*	*Walkway*	*Stop Line*	*Carpark*	*Divider*	*mIoU*
OFT [42]	ResNet-18	C	74.0	35.3	45.9	27.5	35.9	33.9	42.1
LSS [39]	ResNet-18	C	75.4	38.8	46.3	30.3	39.1	36.5	44.4
CVT [64]	EfficientNet-B4	C	74.3	36.8	39.9	25.8	35.0	29.4	40.2
M^2BEV [55]	ResNeXt-101	C	77.2	✗	✗	✗	✗	40.5	✗
BEVFusion [33]	Swin-T	C	81.7	54.8	58.4	47.4	50.7	46.4	56.6
X-Align$_{\text{view}}$	Swin-T	C	**82.4**	**55.6**	**59.3**	**49.6**	**53.8**	**47.4**	**58.0**
PointPillars [22]	VoxelNet	L	72.0	43.1	53.1	29.7	27.7	37.5	43.8
CenterPoint [59]	VoxelNet	L	75.6	48.4	57.5	36.5	31.7	41.9	48.6
PointPainting [49]	ResNet-101, PointPillars	C + L	75.9	48.5	57.1	36.9	34.5	41.9	49.1
MVP [60]	ResNet-101, VoxelNet	C + L	76.1	48.7	57.0	36.9	33.0	42.2	49.0
BEVFusion [33]	Swin-T, VoxelNet	C + L	85.5	60.5	67.6	52.0	57.0	53.7	62.7
X-Align$_{\text{losses}}$	Swin-T, VoxelNet	C + L	85.8	63.1	68.6	53.6	**57.9**	56.7	64.3
X-Align$_{\text{all}}$	Swin-T, VoxelNet	C + L	**86.8**	**65.2**	**70.0**	**58.3**	57.1	**58.2**	**65.7**

Table 1: **Quantitative evaluation on the nuScenes validation set**, in terms of single-class IoUs and the overall mIoU. We compare with existing methods from literature, where the numbers are taken from [33], as their reproduced results are better than the ones originally reported in the papers due to a higher number of trained classes. We also provide information on the backbones and input modalities in the table. Our proposed **X-Align** outperforms all existing approaches in both single-class IoUs and the overall mIoU by a significant margin.

Model	*Encoder*	*Modality*	*mIoU*	*PQ*
PanopticBEV [15]	EffDet-D3	C	25.4	16.0
X-Align$_{view}$	EffDet-D3	C	**27.8**	**16.9**

Table 2: **Quantitative evaluation on KITTI-360** in terms of mIoU and PQ, using the camera-only modality.

4.1. Experimental Setup

Datasets: We evaluate performance on the large-scale nuScenes benchmark [5], which provides ground-truth annotations to support BEV segmentation. It contains 40,000 annotated keyframes captured by a 32-beam LiDAR scanner and six monocular cameras providing a 360° field of view. Following the BEV map segmentation setup from [33], we predict six semantic classes: drivable lanes, pedestrian crossings, walkways, stop lines, carparks, and lane dividers. We further evaluate on KITTI-360 [28], a large-scale dataset with 83,000 annotated frames, including data collected using two fish-eye cameras and a perspective stereo camera. KITTI-360 does not provide dense ground-truth annotations for BEV segmentation. Hence, we use the BEV segmentation annotations from [15] as ground truth. These contain both static classes such as *road* and *sidewalk*, along with dynamic objects such as *cars* and *trucks*.

Evaluation Metrics: For BEV map segmentation, our primary evaluation metric is the mean Intersection Over Union (mIoU). Because some classes may overlap, we apply binary segmentation separately to each class and choose the highest IoU over different thresholds. We then take the mean over all semantic classes to produce the mIoU. This evaluation protocol was proposed in [33]. We additionally use Panoptic Quality (PQ) [21] on KITTI-360 when evaluating panoptic BEV segmentation.

Network Architecture and Training: For evaluation on nuScenes, we build upon BEVFusion [33] for the baseline and train our networks within mmdetection3d [11]. In the camera branch, images are downsampled to 256×704 before going into a Swin-T [31] or ConvNeXt [32] backbone pretrained on ImageNet [43]. The extracted features are fed into several FPN [29] layers and then through a PV-to-BEV transformation based on LSS [39] to be mapped into the BEV space. In the LiDAR branch, we voxelize the points with a grid size of 0.1m and use a sparse convolution backbone [58] to extract the features, which are then flattened onto the BEV space. Given the camera and LiDAR features in BEV space, we utilize our proposed X-FF mechanism from Section 3.3 to fuse them. We use the self-attention module in our main results, providing the best accuracy-computation trade-off (see Fig. 4). The fused features are fed into a BEV encoder and FPN layers similar to those in SECOND [58] and subsequently to a segmentation head as in BEVFusion [33]. Since nuScenes does not provide ground-truth PV segmentation labels, we utilize a SOTA model pre-trained on Cityscapes to generate pseudo-labels for supervising our PV segmentation in X-SA. Specifically, we use an HRNet-w48 [52] trained with InverseForm [4].

On KITTI-360, we take the camera-only PanopticBEV [15] as the baseline, which we retrain using the code and hyperparameters released by the authors. Then, we include our two proposed X-SA losses on top of this baseline to generate the X-Align$_{\text{view}}$ results on KITTI-360.

Additional hyperparameters and training details for all experiments can be found in the Appendix.

4.2. Quantitative Evaluation

nuScenes Camera-LiDAR Fusion: We report segmentation results based on camera-LiDAR fusion in the bottom section of Table 1. We evaluate in the region-bound of [-

Model	PV	X-FA	PV2BEV	X-FF	*Drivable*	*Ped. Cross.*	*Walkway*	*Stop Line*	*Carpark*	*Divider*	*mIoU*	*GFlops*	*fps*
Baseline	✗	✗	✗	✗	85.5	60.5	67.6	52.0	57.0	53.7	62.7	364.3	5.1
	✓	✗	✗	✗	85.7	62.8	68.4	52.4	56.5	56.1	63.7	364.3	5.1
	✗	✓	✗	✗	85.6	62.3	68.2	51.6	56.4	55.9	63.4	364.3	5.1
X-Align$_{\text{view}}$	✓	✗	✓	✗	85.8	63.1	68.6	53.2	57.7	56.4	64.1	364.3	5.1
X-Align$_{\text{losses}}$	✓	✓	✓	✗	85.8	63.1	68.6	53.6	57.9	56.7	64.3	364.3	5.1
	✗	✗	✗	✓	**86.8**	64.3	69.5	54.5	**59.5**	57.6	65.3	367.4	5.0
	✗	✓	✗	✓	**86.8**	65.1	69.8	**60.0**	56.5	58.2	65.4	367.4	5.0
	✓	✗	✓	✓	**86.8**	65.0	**70.0**	55.9	57.0	58.1	65.5	367.4	5.0
X-Align$_{\text{all}}$	✓	✓	✓	✓	**86.8**	**65.2**	**70.0**	58.3	57.1	**58.2**	**65.7**	367.4	5.0

Table 3: **Ablation study on the proposed X-Align components.** In the top part, we show the effects of our proposed losses, *i.e.*, the Cross-Modal Feature Alignment (X-FA) loss and Cross-View Segmentation Alignment (X-SA) comprised of the PV and PV2BEV segmentation losses. In the bottom part, we further show the improvements enabled by our Cross-Modal Feature Fusion (X-FF).

50m, 50m]×[-50m, 50m] around the ego car following the standard evluation procedure on nuScenes [25, 33, 39, 55, 64]. We compare our complete method **X-Align**$_{\text{all}}$ with existing SOTA approaches on BEV segmentation and see that X-Align significantly outperforms them. Specifically, X-Align achieves a new record mIoU of **65.7%** on nuScenes BEV segmentation and consistently improves across all the classes, thanks to the proposed novel cross-modal and cross-view alignment strategies. We used the Self-Attention block illustrated in Figure 3 as our preferred X-FF strategy, as it provided the optimal trade-off in Figure 4. For this strategy, the computational overhead (**0.8%**) and increase in latency (**2%**) is minimal as observed in Table 1. We further show the performance of X-Align but without using the more advanced fusion module, *i.e.*, **X-Align**$_{\text{losses}}$, in the second last row of Table 1. X-Align still significantly outperforms existing methods even without introducing additional computational complexity during inference.

nuScenes Camera-Only: To demonstrate the efficacy of our X-SA scheme, we evaluate an instance of X-Align, *i.e.*, **X-Align**$_{\text{view}}$, using only the camera branch for BEV map segmentation. The results and comparisons are shown in the top part of Table 1. It can be seen that **X-Align**$_{\text{view}}$ considerably outperforms the existing best performance, achieving a record mIoU of **58.0%**, surpassing the prior camera-only SOTA by **1.4** points in mIoU. This shows the benefit of X-SA, which enhances the intermediate semantic features and the PV-to-BEV transformation without incurring computational overhead at inference.

KITTI-360 Camera-Only: We present additional camera-only results on KITTI-360 in Table 2, for the task of panoptic BEV segmentation. In this case, we use PanopticBEV [15] as our baseline. Additionally, we include our two novel X-SA losses to train our **X-Align**$_{\text{view}}$ model.[1] It can be seen that our proposed approach improves the baseline in terms of both mIoU and PQ scores.

Overall, our proposed X-Align consistently improves upon the existing methods across modalities and classes on both nuScenes and KITTI-360, demonstrating the efficacy of our proposed X-FF, X-FA, and X-SA components.

[1]The baseline scores are obtained by training PanopticBEV using the authors' code in https://github.com/robot-learning-freiburg/PanopticBEV.

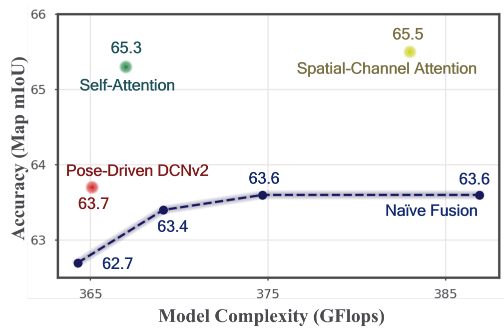

Figure 4: **Accuracy-Computation Analysis**: We compare our proposed Cross-Modal Feature Fusion (X-FF) designs with simply scaling up the fusion mechanism (adopted by existing methods [23, 33]) in terms of accuracy and computation complexity.

4.3. Ablation Study

We conduct an ablation study on the different X-Align components and summarize our results in Table 3. We evaluate model variants using different combinations of our proposed novel losses, including Cross-Modal Feature Alignment (X-FA) loss and our Cross-View Segmentation Alignment (X-SA) comprised of our PV and PV2BEV losses. Additionally, we investigate the effect of our Cross-Modal Feature Fusion (X-FF).

Our X-Align$_{\text{view}}$ variant, leveraging both PV and PV2BEV losses, improves the mIoU from **62.7%** to **64.1%** when compared to the baseline. Specifically, the PV loss alone contributes to a 1-point mIoU improvement. When using the X-FA loss, we increase the baseline mIoU from **62.7%** to **63.4%**. Finally, adding all the losses together, our X-Align$_{\text{losses}}$ variant boosts the mIoU score to **64.3%**, significantly surpassing the baseline's mIoU. Notably, these improvements rely on new training losses and have no computational overhead during inference.

Next, we study the effect of our X-FF module. Without any new losses, X-FF enables a 2.6-point mIoU improvement over the baseline (with a minor increase of computation, cf. Table 3 and Fig. 4). This shows that simple concate-

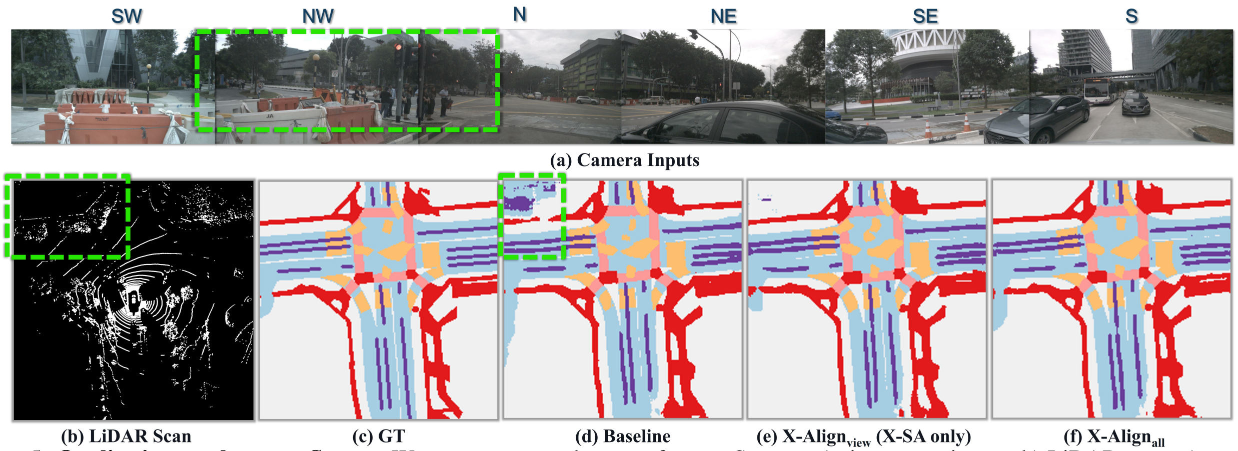

Figure 5: **Qualitative results on nuScenes.** We present a sample scene from nuScenes: a) six camera inputs, b) LiDAR scan, c) ground-truth BEV segmentation map, d) baseline BEV segmentation, e) BEV segmentation using **X-Align**$_{\mathrm{view}}$, and d) BEV segmentation **X-Align**$_{\mathrm{all}}$. We observe that the baseline model prediction is highly erroneous in the region highlighted in green. We highlight this region of interest in the input views as well. By using the two X-SA losses, **X-Align**$_{\mathrm{view}}$ can already correct substantial errors in the baseline prediction, and the **X-Align**$_{\mathrm{all}}$ model further improves accuracy.

nation is a key limitation in the baseline, which fails to fuse the camera and LiDAR features properly. Finally, using all our novel loss functions together with the X-FF module, we arrive at the full X-Align$_{\mathrm{all}}$ model, which achieves the overall best performance of **65.7%** mIoU, significantly higher than the baseline's **62.7%** mIoU. Our extensive ablation study results show that each of our proposed components in X-Align provides a meaningful contribution to improving the SOTA BEV segmentation performance.

4.4. Accuracy-Computation Analysis

In Fig. 4, we report the accuracy-computation trade-off by utilizing our different X-FF fusion strategies, including self-attention, spatial-channel attention, and pose-driven deformable convolution (DCNv2). It can be seen that when using spatial-channel attention, we achieve the highest accuracy improvement at a higher computational cost, while pose-driven DCNv2 introduces the least amount of additional cost but provides less performance gain. Using self-attention, on the other hand, provides the best trade-off between performance and complexity.

We further compare by naively scaling up the complexity of the baseline fusion, *e.g.*, by adding more layers and channels in the convolution blocks, shown by the blue curve. It can be seen that the baseline's performance saturates, and all our proposed fusion methods achieve better trade-offs as compared to the baseline. This again verifies that the baseline fusion using simple concatenation and convolutions does not provide the suitable capacity for the model to align and aggregate multi-modal features.

4.5. Qualitative Results

In Fig. 5, we present qualitative results on a sample test scene from nuScenes, showing both LiDAR and camera inputs. We compare the BEV segmentation maps of different models, including the baseline, X-Align$_{\mathrm{view}}$ (only using the two X-SA losses), and the full X-Align, *i.e.*, X-Align$_{\mathrm{all}}$.

In this scene, the baseline wrongly predicts the building in the NW image as part of a road in the BEV segmentation output due to the inaccurate PV-to-BEV transformation, cf. Fig. 5(d). Since the building is not captured in the LiDAR scan (see Fig. 5(b)), the LiDAR branch also cannot correct the camera projection later in the fusion. However, by utilizing our Cross-View Segmentation Alignment (X-SA), this erroneous projection can be largely rectified, as shown in Fig. 5(e). The remnants of this error are then completely removed when we apply our proposed alignment and fusion schemes, X-FA and X-FF, which enables proper fusion of the visual information from the camera and the geometric information from the LiDAR. We can see in Fig. 5(f) that our full X-Align Model can accurately predict the BEV segmentation map. We refer readers to the Appendix for more visual examples.

5. Conclusions

In this paper, we proposed a novel framework, X-Align, which addresses cross-view and cross-modal alignment in BEV segmentation. It enhances the alignment of unimodal features to aid feature fusion and the alignment between perspective view and bird's-eye-view representations. Our experiments show that X-Align improves performance on nuScenes and KITTI-360 datasets, in particular outperforming previous SOTA by 3 mIoU points on nuScenes. We also verified the effectiveness of the X-Align components via an extensive ablation study. As part of future work, we believe that X-Align can further benefit other multi-modal perception tasks.

References

[1] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust Lidar-Camera Fusion for 3D Object Detection with Transformers. In *Proc. of CVPR*, pages 1090–1099, 2022. 2, 3

[2] Shubhankar Borse, Hong Cai, Yizhe Zhang, and Fatih Porikli. Hs3: Learning with proper task complexity in hierarchically supervised semantic segmentation. 2021. 2

[3] Shubhankar Borse, Hyojin Park, Hong Cai, Debasmit Das, Risheek Garrepalli, and Fatih Porikli. Panoptic, instance and semantic relations: A relational context encoder to enhance panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1269–1279, 2022. 2

[4] Shubhankar Borse, Ying Wang, Yizhe Zhang, and Fatih Porikli. Inverseform: A loss function for structured boundary-aware segmentation. In *Proc. of CVPR*, pages 5901–5911, 2021. 2, 6

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proc. of CVPR*, pages 11621–11631, 2020. 2, 6

[6] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. FUTR3D: A Unified Sensor Fusion Framework for 3D Detection. *arXiv preprint arXiv:2203.10642*, 2022. 3

[7] Yukang Chen, Yanwei Li, Xiangyu Zhang, Jian Sun, and Jiaya Jia. Focal Sparse Convolutional Networks for 3D Object Detection. In *Proc. of CVPR*, pages 5428–5437, 2022. 3

[8] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinghong Jiang, Feng Zhao, Bolei Zhou, and Hang Zhao. AutoAlign: Pixel-Instance Feature Aggregation for Multi-Modal 3D Object Detection. *arXiv preprint arXiv:2201.06493*, 2022. 3

[9] Kashyap Chitta, Aditya Prakash, and Andreas Geiger. NEAT: Neural Attention Fields for End-to-End Autonomous Driving. In *Proc. of ICCV*, pages 15793–15803, 2021. 1

[10] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *arXiv preprint arXiv:2205.15997*, 2022. 2

[11] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. `https://github.com/open-mmlab/mmdetection3d`, 2020. 6

[12] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Proc. of CVPR*, pages 3213–3223, 2016. 5

[13] Noa Garnett, Rafi Cohen, Tomer Pe'er, Roee Lahav, and Dan Levi. 3D-LaneNet: End-to-End 3D Multiple Lane Detection. In *Proc of ICCV*, pages 2921–2930, 2019. 2

[14] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision Meets Robotics: The KITTI Dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[15] Nikhil Gosala and Abhinav Valada. Bird's-Eye-View Panoptic Segmentation Using Monocular Frontal View Images. *IEEE RA-L*, 7(2):1968–1975, 2022. 2, 6, 7

[16] Adam W Harley, Zhaoyuan Fang, Jie Li, Rares Ambrus, and Katerina Fragkiadaki. A Simple Baseline for BEV Perception Without LiDAR. *arXiv preprint arXiv:2206.07959*, 2022. 3, 4

[17] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. FIERY: Future Instance Prediction in Bird's-Eye View From Surround Monocular Cameras. In *Proc. of ICCV*, pages 15273–15282, 2021. 1, 2

[18] Hanzhe Hu, Yinbo Chen, Jiarui Xu, Shubhankar Borse, Hong Cai, Fatih Porikli, and Xiaolong Wang. Learning implicit feature alignment function for semantic segmentation. *arXiv preprint arXiv:2206.08655*, 2022. 2

[19] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. BEVDet: High-performance Multi-camera 3D Object Detection in Bird-Eye-View. *arXiv preprint arXiv:2112.11790*, 2021. 3

[20] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial Transformer Networks. In *Proc. of NIPS*, pages 2017–2025, 2015. 5

[21] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proc. of CVPR*, pages 9404–9413, 2019. 6

[22] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proc. of CVPR*, pages 12697–12705, 2019. 6

[23] Qi Li, Yue Wang, Yilun Wang, and Hang Zhao. Hdmapnet: A local semantic map learning and evaluation framework. *arXiv preprint arXiv:2107.06307*, 2021. 3, 4, 7

[24] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V. Le, Alan Yuille, and Mingxing Tan. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proc. of CVPR*, pages 17182–17191, 2022. 2, 3

[25] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. BEVFormer: Learning Bird's-Eye-View Representation from Multi-Camera Images via Spatiotemporal Transformers. In *Proc. of ECCV*, 2022. 3, 7

[26] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep Continuous Fusion for Multi-sensor 3D Object Detection. In *Proc. of ECCV*, pages 641–656, 2018. 3

[27] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. BEVFusion: A Simple and Robust LiDAR-Camera Fusion Framework. *arXiv preprint arXiv:2205.13790*, 2022. 3

[28] Yiyi Liao, Jun Xie, and Andreas Geiger. KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *Proc. of PAMI*, 2022. 6

[29] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proc. of CVPR*, pages 2117–2125, 2017. 6

[30] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection. In *Proc. of ICCV*, pages 2980–2988, 2017. 3

[31] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In *Proc. of ICCV*, pages 10012–10022, 2021. 6

[32] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *Proc. of CVPR*, pages 11976–11986, 2022. 6

[33] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation. *arXiv preprint arXiv:2205.13542*, 2022. 1, 2, 3, 4, 6, 7

[34] Abdelhak Loukkal, Yves Grandvalet, Tom Drummond, and You Li. Driving Among Flatmobiles: Bird-Eye-View Occupancy Grids From a Monocular Camera for Holistic Trajectory Planning. In *Proc. of WACV*, pages 51–60, 2021. 2

[35] Chenyang Lu, Marinus Jacobus Gerardus van de Molengraft, and Gijs Dubbelman. Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks. *IEEE RA-L*, 4(2):445–452, 2019. 2

[36] Muhammad Maaz, Abdelrahman Shaker, Hisham Cholakkal, Salman Khan, Syed Waqas Zamir, Rao Muhammad Anwer, and Fahad Shahbaz Khan. EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications. *arXiv preprint arXiv:2206.10589*, 2022. 4

[37] Ramin Nabati and Hairong Qi. CenterFusion: Center-Based Radar and Camera Fusion for 3D Object Detection. In *Proc. of WACV*, pages 1527–1536, 2021. 2

[38] Bowen Pan, Jiankai Sun, Ho Yin Tiga Leung, Alex Andonian, and Bolei Zhou. Cross-view Semantic Segmentation for Sensing Surroundings. *IEEE Robotics and Automation Letters*, 5(3):4867–4873, 2020. 2

[39] Jonah Philion and Sanja Fidler. Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D. In *Proc. of ECCV*, pages 194–210, 2020. 1, 2, 3, 6, 7

[40] Charles R. Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas. Frustum PointNets for 3D Object Detection From RGB-D Data. In *Proc. of CVPR*, pages 918–927, 2018. 2

[41] Thomas Roddick and Roberto Cipolla. Predicting Semantic Map Representations From Images Using Pyramid Occupancy Networks. In *Proc. of CVPR*, pages 11138–11147, 2020. 2

[42] Thomas Roddick, Alex Kendall, and Roberto Cipolla. Orthographic Feature Transform for Monocular 3D Object Detection. In *Proc. of BMVC*, pages 1–10, 2018. 2, 6

[43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet Large Scale Visual Recognition Challenge. *Proc. of IJCV*, 115(3):211–252, 2015. 6

[44] Avishkar Saha, Oscar Mendez, Chris Russell, and Richard Bowden. Enabling Spatio-temporal Aggregation in Birds-Eye-View Vehicle Estimation. In *Proc. of ICRA*, pages 5133–5139, 2021. 2

[45] Hao Shao, Letian Wang, Ruobing Chen, Hongsheng Li, and Yu Liu. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. *arXiv preprint arXiv:2207.14024*, 2022. 2

[46] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, and Aleksei Timofeev. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *Proc. of CVPR*, 2020. 2

[47] Ömer Şahin Taş, Florian Kuhnt, J Marius Zöllner, and Christoph Stiller. Functional System Architectures Towards Fully Automated Driving. In *Proc. of IV*, pages 304–309, 2016. 1

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All You Need. In *Proc. of NIPS*, pages 5998–6008, Dec. 2017. 4

[49] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential Fusion for 3D Object Detection. In *Proc. of CVPR*, pages 4604–4612, 2020. 2, 6

[50] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. A Geometric Approach to Obtain a Bird's Eye View From an Image. In *Proc of ICCV - Workshops*, pages 1–10, 2019. 2

[51] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. PointAugmenting: Cross-Modal Augmentation for 3D Object Detection. In *Proc of CVPR*, pages 11794–11803, 2021. 2

[52] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep High-resolution Representation Learning for Visual Recognition. *Proc. of PAMI*, 43(10):3349–3364, 2020. 6

[53] Zhixin Wang and Kui Jia. Frustum ConvNet: Sliding Frustums to Aggregate Local Point-Wise Features for Amodal 3D Object Detection. In *Proc. of IROS*, pages 1742–1749, 2019. 2

[54] Pengxiang Wu, Siheng Chen, and Dimitris N. Metaxas. MotionNet: Joint Perception and Motion Prediction for Autonomous Driving Based on Bird's Eye View Maps. In *Proc. of CVPR*, pages 11385–11395, 2020. 1

[55] Enze Xie, Zhiding Yu, Daquan Zhou, Jonah Philion, Anima Anandkumar, Sanja Fidler, Ping Luo, and Jose M. Alvarez. M$\hat{2}$BEV: Multi-Camera Joint 3D Detection and Segmentation with Unified Birds-Eye View Representation. *arXiv preprint arXiv:2204.05088*, 2022. 1, 2, 6, 7

[56] Runsheng Xu, Zhengzhong Tu, Hao Xiang, Wei Shao, Bolei Zhou, and Jiaqi Ma. CoBEVT: Cooperative Bird's Eye View Semantic Segmentation with Sparse Transformers. *arXiv preprint arXiv:2207.02202*, 2022. 1, 2

[57] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Zhou Bin, and Liangjun Zhang. FusionPainting: Multimodal Fusion with Adaptive Attention for 3D Object Detection. In *Proc. of ITSC*, pages 3047–3054, 2021. 3

[58] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely Embedded Convolutional Detection. *Sensors*, 18(10):3337, 2018. 3, 6

[59] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3D Object Detection and Tracking. In *Proc. of CVPR*, pages 11784–11793, 2021. 2, 6

[60] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal Virtual Point 3D Detection. In *Proc. of NeurIPS*, pages 16494–16507, 2021. 3, 6

[61] Yizhe Zhang, Shubhankar Borse, Hong Cai, and Fatih Porikli. Auxadapt: Stable and efficient test-time adaptation for temporally consistent video semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2339–2348, 2022. 2

[62] Yizhe Zhang, Shubhankar Borse, Hong Cai, Ying Wang, Ning Bi, Xiaoyun Jiang, and Fatih Porikli. Perceptual consistency in video segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2564–2573, 2022. 2

[63] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. BEVerse: Unified Perception and Prediction in Birds-Eye-View for Vision-Centric Autonomous Driving. *arXiv preprint arXiv:2205.09743*, 2022. 2

[64] Brady Zhou and Philipp Krähenbühl. Cross-View Transformers for Real-Time Map-View Semantic Segmentation. In *Proc. of CVPR*, pages 13760–13769, 2022. 1, 2, 6, 7

[65] Minghan Zhu, Songan Zhang, Yuanxin Zhong, Pingping Lu, Huei Peng, and John Lenneman. Monocular 3D Vehicle Detection Using Uncalibrated Traffic Cameras through Homography. In *Proc. of IROS*, pages 3814–3821, 2021. 2

[66] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable Convnets V2: More Deformable, Better Results. In *Proc. of CVPR*, pages 9308–9316, 2019. 4