

M²-3DLaneNet: Multi-Modal 3D Lane Detection

Yueru Luo¹, Xu Yan¹, Chaoda Zheng¹,
Chao Zheng², Shuqi Mei², Tang Kun², Shuguang Cui¹, Zhen Li^{1*}

¹ The Chinese University of Hong Kong (Shenzhen), The Future Network of Intelligence Institute,
² Tencent Map, T Lab

Abstract

Estimating accurate lane lines in 3D space remains challenging due to their sparse and slim nature. In this work, we propose the **M²-3DLaneNet**, a **Multi-Modal** framework for effective 3D lane detection. Aiming at integrating complementary information from multi-sensors, M²-3DLaneNet first extracts multi-modal features with modal-specific backbones, then fuses them in a unified Bird's-Eye View (BEV) space. Specifically, our method consists of two core components. 1) To achieve accurate 2D-3D mapping, we propose the **top-down BEV generation**. Within it, a Line-Restricted Deform-Attention (LRDA) module is utilized to effectively enhance image features in a top-down manner, fully capturing the slenderness features of lanes. After that, it casts the 2D pyramidal features into 3D space using depth-aware lifting and generates BEV features through pillarization. 2) We further propose the **bottom-up BEV fusion**, which aggregates multi-modal features through multi-scale cascaded attention, integrating complementary information from camera and LiDAR sensors. Sufficient experiments demonstrate the effectiveness of M²-3DLaneNet, which outperforms previous state-of-the-art methods by a large margin, *i.e.*, **12.1%** F1-score improvement on OpenLane dataset. Our code will be released at <https://github.com/JMoonr/mmlane>.

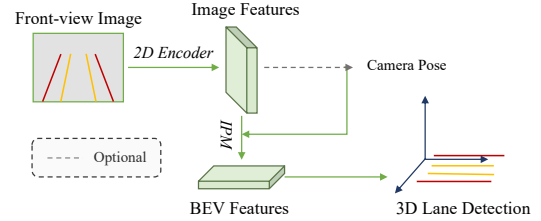
1 Introduction

Accurate and robust lane detection is the foundation for safety in autonomous driving. Over the past few years, camera-based 2D lane detection has been extensively studied and achieved impressive results (Gopalan et al. 2012; Li et al. 2016; Lee et al. 2017; Neven et al. 2018; Zhang et al. 2018; Hou et al. 2019; Liu et al. 2021a; Tabelini et al. 2021a; Han et al. 2022; Zheng et al. 2022). However, detecting lanes on monocular 2D images cannot provide accurate localization in 3D space due to the depth ambiguity.

Fortunately, large-scale datasets with 3D lane annotations (Garnett et al. 2019; Guo et al. 2020; Yan et al. 2022) have been proposed, triggering a surge of interest in developing 3D lane detection algorithms (Garnett et al. 2019; Efrat et al. 2020; Guo et al. 2020; Liu et al. 2022; Chen et al. 2022a; Yan et al. 2022). These methods detect 3D lanes in a vision-centric manner, which takes camera images as inputs and predicts 3D lanes in the bird's eye view (BEV) space.

*Corresponding author

(a) Monocular 3D Lane Detection



(b) M²-3DLaneNet

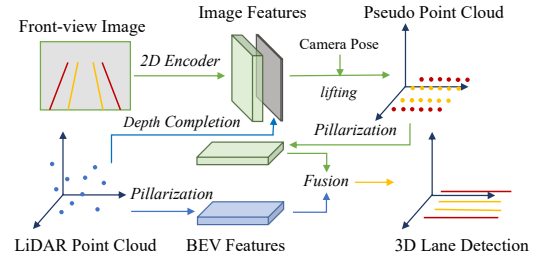


Figure 1: (a) Previous methods (Garnett et al. 2019; Chen et al. 2022a) mainly utilize inverse perspective mapping (IPM) to transform front-view image features to BEV features, through either ground truth or learned camera poses. (b) M²-3DLaneNet first lifts image features to 3D space with the help of LiDAR point cloud, and then fuse multi-modality features in the BEV for the 3D lane detection.

Due to the lack of depth information, these approaches map the 2D features into BEV via an inverse perspective mapping (Figure 1 (a)) based on the flat road assumption, which does not hold for most real-world scenarios (*e.g.*, in the sloping terrain). Besides, they become unreliable in low-light conditions because cameras are sensitive to illumination change. Recently, LiDARs are becoming increasingly popular in autonomous driving systems. Since LiDARs provide accurate and wide-ranging depth information regardless of lighting variations, an intuitive way to address the above issues is to detect 3D lanes using LiDAR point clouds. Thanks to rich 3D geometric information, detecting flat grounds from other objects (*e.g.*, cars, pedestrians) also becomes much easier in point clouds, which is preferable for lane detection (lanes are always on the grounds). Nevertheless, detecting 3D lanes using only textureless LiDAR data is also difficult because

lanes are usually slender and thus submerged by their surrounding surfaces.

Based on the above observation, we aim to improve 3D lane detection by leveraging complementary information from camera images and LiDAR point clouds. In this paper, we propose the M^2 -3DLaneNet, a multi-modal framework for accurate 3D Lane detection, breaking the limitations of previous single-modal approaches, as shown in Figure 1 (b). Taking geometric-rich point clouds and texture-rich images as inputs, M^2 -3DLaneNet first processes them in two modal-specific streams and then predicts the 3D lanes based on multi-scale cross-modal feature fusion. In practice, to obtain 3D lane prediction, we propose to conduct feature fusion in a unified BEV space. Specifically, for the LiDAR inputs, we simply generate the corresponding features in BEV space using a 3D pillar-based backbone (Lang et al. 2019). For the camera stream, the input image is first encoded into multi-scale feature maps and then transformed into the BEV space via our Top-Down BEV generation (**TD-BEV**) module. With explicit consideration of the sparse and slim nature of 3D lanes, the proposed **TD-BEV** module first mines lane information from the 2D feature maps. The texture-rich features are then effectively transformed to BEV space via depth-aware lifting, followed by BEV projection via pillarization and the nearest BEV completion operation. After having the multi-scale BEV features from two modalities, we generate the fused features in a bottom-up manner (**BU-BEV**), which are used for the final lane prediction. Experiments on OpenLane dataset show that our model outperforms the previous state-of-the-art by a large margin. The extensive analysis also confirms the effectiveness of each proposed component.

In summary, our main contributions are threefold:

- We propose the M^2 -3DLaneNet, a multi-modal framework for accurate 3D lane detection. It utilizes complementary features from camera image and LiDAR point cloud for 3D lane detection.
- Two novel components, namely top-down BEV generation and bottom-up BEV fusion, are proposed, in which the former effectively lifts the 2D image features into the BEV space, and the latter aggregate multi-modal features in a cascaded attention.
- We demonstrate that M^2 -3DLaneNet significantly exceeded the previous state-of-the-art on the OpenLane benchmark (+12.1% in the metric of F1-score).

2 Related work

2.1 2D Lane Detection

The goal of 2D lane detection is to detect the positions of lane in the 2D images. Among recent methods, there are four main stream strategies adopted to perform this task: (a) Anchor-based methods (Li et al. 2019; Torres et al. 2020; Qin, Wang, and Li 2020; Zheng et al. 2022) leverage line-like anchors to detect lanes out of special nature of the slender lane. (Torres et al. 2020) utilizes attention between anchors to aggregate global information. (b) Segmentation-based methods (Hou et al. 2019; Neven et al. 2018; Liu et al. 2021a; Wu et al. 2021) aim to predict the lane mask

through pixel-wise classification task. (c) Parametric-based methods (Liu et al. 2021b; Feng et al. 2022; Tabelini et al. 2021b) are in a characteristic way to resolve the traffic line detection, i.e, learn to predict parameters used to fit the polynomial curves of lanes; (Liu et al. 2021b; Tabelini et al. 2021b) model lane shape with polynomial parameters, while (Feng et al. 2022) adopts Bézier curve to capture holistic lane structure. (d) Key-point-based methods (Qu et al. 2021; Wang et al. 2022) formulate lane detection problem from key points perspective and finally get lanes through associating points in the same lane. However, since lane annotations are defined on the 2D images, these methods lack the ability to accurately localize lanes in 3D space.

2.2 3D Lane Detection.

3D lane detection aims at predicting lane lines in the 3D space. Although generating 3D predictions, most deep-learning-based 3D lane detection algorithms still take monocular images as inputs due to the lack of public 3D datasets. Among them, 3D-LaneNet (Garnett et al. 2019) is the first to detect 3D lanes using monocular front-view images. To predict lanes on a 3D plane, it applies inverse perspective mapping (IPM) to transform the front view image into the top-view, using camera poses predicted by a learning branch. Gen-LaneNet (Guo et al. 2020) proposes a virtual top view as a surrogate to address the misalignment between anchor representation and internal features projected by IPM. CLGo (Liu et al. 2022) realizes a two-stage framework with pose estimation and polynomials estimation. These methods rely on IPM to project image features to top-view features. However, IPM will cause distortion when encountering uphill or downhill, which jeopardizes the model to perceive the scene correctly and consistently. By contrast, ONCE (Yan et al. 2022) directly generates 3D lanes without relying on IPM by detecting 2D lanes on images and projecting them into 3D space with the help of depth estimation. And Persformer (Chen et al. 2022a) leverages the deformable attention mechanism to adaptively update features on the BEV plane, which also mitigates the discrepancy introduced by IPM. Nevertheless, these monocular approaches heavily rely on the camera features, which are depth ambiguous and sensitive to the light condition. Although some approaches attempt to detect 3D lanes using LiDAR data (Jung and Bae 2018; Thuy and León 2010; Kammel and Pitzer 2008). They heavily rely on the hand-craft intensity threshold, which is hard to determine across different environments. Fortunately, (Chen et al. 2022a) provides a large-scale 3D lane detection dataset (OpenLane), which is the **first** public 3D lane dataset containing both LiDAR and camera data with pixel-to-point correspondences. Such a new dataset enables us to develop a multi-modal approach to achieve better 3D lane detection.

2.3 Multi-sensor Approaches.

Since cameras and LiDARs capture complementary information, multi-sensor approaches are widely adopted in different fields. In 3D object detection, PointPainting (Vora et al. 2020) provides point clouds with their corresponding 2D semantics. Taking advantage of the attention mech-

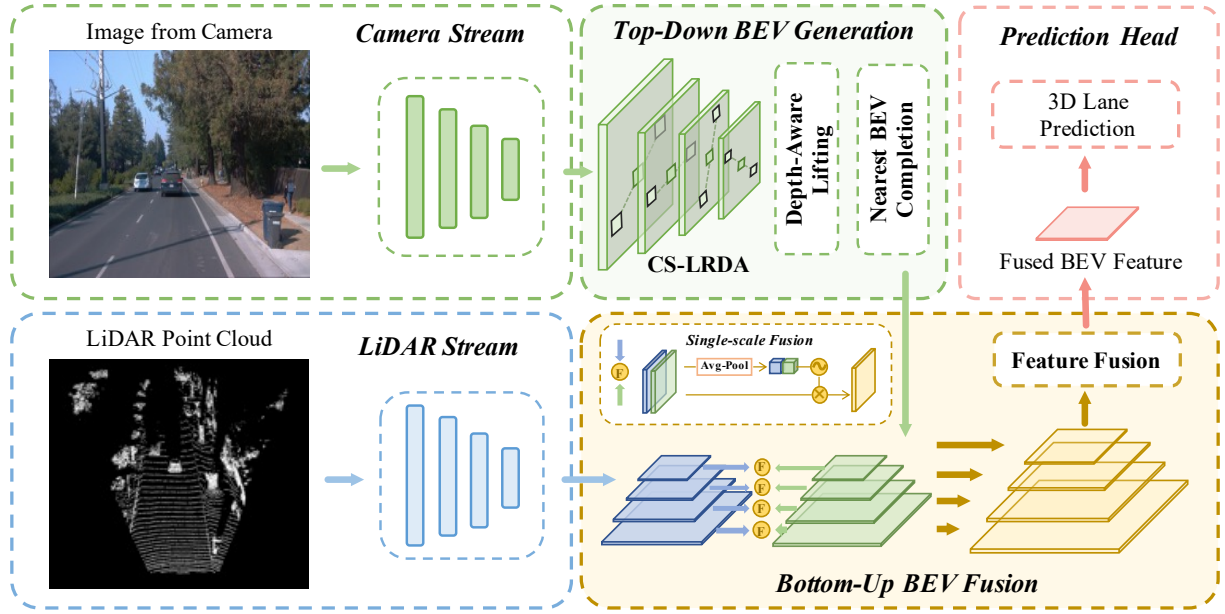


Figure 2: **Overview of the proposed M^2 -3DLaneNet.** It parallelly generates BEV features through taking image and LiDAR point cloud as inputs, where features from the former is gained by top-down BEV generation. Afterward, two BEV features are fused together with bottom-up BEV fusion and finally used to predict 3D lanes.

anism, (Li et al. 2022; Chen et al. 2022c; Bai et al. 2022) adaptively model the 2D-3D mapping through multi-modal fusion. SFD (Wu et al. 2022) applies depth completion before lifting the 2D images into the 3D space. Different from them, we lift multi-scale line-restricted features, providing the detector with richer global features. For the 3D lane detection, there are few previous approaches related to multi-sensor since the lack of dataset. Early work (Bai et al. 2018) adopts multi-modal input on their **private dataset**, but they require supervision from additional high-definition maps. Moreover, they do not explore geometric information in LiDAR point clouds and only extract features through 2D-CNN using a three-channel image gained by LiDAR points. By contrast, M^2 -3DLaneNet explores complementary features from both camera and LiDAR, and enhance multi-modal feature through novel BEV generation and fusion.

3 Method

The architecture of M^2 -3DLaneNet is shown in Figure 2, in which two streams (*i.e.*, camera and LiDAR streams) are adopted for multi-scale feature generation. After that, the multi-scale image features are further enhanced through Top-Down BEV generation (TD-BEV). Then, Bottom-Up BEV fusion (BU-BEV) is applied to aggregate multi-modal features. Finally, the 3D lane prediction is gained by an anchor-based prediction head.

3.1 Two-Stream Architecture

In the initial stage, there is a two-stream backbone in parallel used for multi-scale feature generation.

Camera Stream. For the image stream, RGB images are encoded into multi-scale feature maps. Specifically, follow-

ing (Chen et al. 2022a), we use EfficientNet-B5 (Tan and Le 2019) as our image backbone, which generates features from four different scales.

LiDAR Stream. For LiDAR Stream, the LiDAR point cloud is encoded through PointPillars (Lang et al. 2019). Concretely, it first divides the point cloud into different pillars and then encodes each pillar into a high-dimensional feature vector using a mini-PointNet (Qi et al. 2017). After that, all pillars are scattered into the 2D BEV space and processed by a 2D-CNN. Similar to the camera stream, we generate LiDAR BEV features in four scales, which will be further aggregated with the image BEV features in the bottom-up BEV fusion module.

3.2 Top-Down BEV Generation

Taking features from the camera stream as input, the target of Top-Down BEV generation (TD-BEV) is obtaining multi-scale image BEV features. It firstly enhances the 2D feature maps using the newly proposed Cross-Scale Line-Restricted Deform-Attention (CS-LRDA) module, which aggregates deeper features with shallow ones and explores the long and slim nature of lane lines. After that, it transforms these enhanced features into the BEV space through depth-aware lifting and nearest BEV completion.

Line Restricted Deform-Attention. Deformable attention (DA) is initially proposed in Deformable-DETR (Zhu et al. 2020), which breaks the limitation of traditional grid sampling by selecting sparse and deformable neighbors around each pixel. The core idea of DA is to adaptively attend to relevant regions with predicted offsets, and thus can capture objects on different scales with the attention mechanism. Nonetheless, compared with generic object detection,

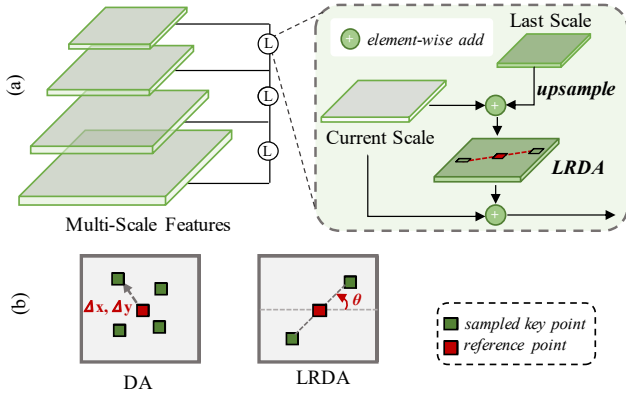


Figure 3: **Cross-Scale Line-Restricted Deform-Attention (CS-LRDA)**. (a) The workflow of CS-LRDA, in which the features from two different scales are merged using our proposed LRDA. (b) The comparison of previous deformable attention (DA) with our line-restricted deformable attention (LRDA). Since this operation is applied from the top to the bottom, the input of the first layer is just itself.

detecting lane lines relies on more fine-grained features because lanes tend to be slim.

Based on this observation, we propose **Line Restricted Deform-Attention (LRDA)** to enhance the feature representation of lane lines (Figure 3 (b)). Unlike previous DA that could attend to any position, LRDA restricts that the sample points lie in a line crossing the key point, where the line is given by a predicted angle. To obtain the position of sample points for a key point, LRDA simply samples on the predicted line using predefined steps. After that, LRDA updates the feature of each key point by applying the attention mechanism between sample points and the key point. Specifically, it can be formulated as

$$\text{LRDA}(\mathbf{q}_i, \mathbf{p}_i, \mathbf{f}) = \sum_{n=1}^N \mathbf{W}_n \cdot \mathbf{F}_{LR}, \quad (1)$$

$$\mathbf{F}_{LR} = \sum_{k=1}^K A_{nik} \cdot \mathbf{W}'_n \mathbf{f}(\mathbf{p}_i + \Delta \mathbf{p}_{nik}), \quad (2)$$

$$\Delta \mathbf{p}_{nik} = (\cos \theta \cdot \mathbf{s}_{nik}, \sin \theta \cdot \mathbf{s}_{nik}), \quad (3)$$

where $\mathbf{f} \in \mathbb{R}^{C \times H \times W}$ represents a feature map, i indexes a query component with feature \mathbf{q}_i and a position \mathbf{p}_i , n and k respectively denote the index of attention head and sampled points, and K illustrates the number of sampled points. $\Delta \mathbf{p}_{nik}$ and A_{nik} further represent offsets and weight of the k^{th} sampled point in the n^{th} attention head. Note that A_{nik} is a normalized scalar over K sampled points, and in the range of $[0, 1]$. Therefore, the offsets $\Delta \mathbf{p}_{nik} \in \mathbb{R}^2$ are determined by the predicted angle θ and step \mathbf{s}_{nik} , i.e., the sampling distance between the k^{th} sampled point and the reference point \mathbf{p}_i . Following (Zhu et al. 2020), we conduct the bilinear interpolation to obtain features at fractional sampling positions. Through such a scheme, lane-aware discriminative features can get enhanced and negative influences

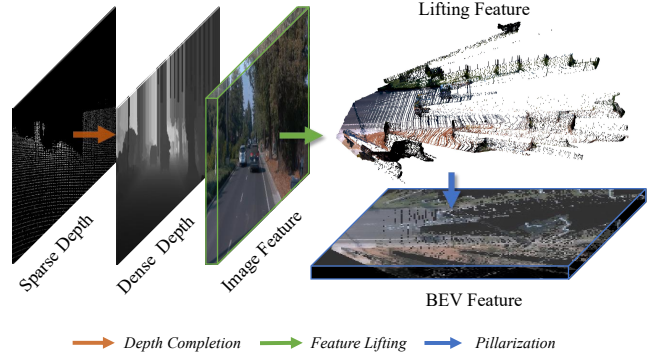


Figure 4: **Depth-Aware Lifting**. It firstly completes the sparse depth to the dense one. After that, it lifts the image feature in the 3D space according to the dense depth and image features. Finally, it conducts pillarization to obtain the BEV features.

from the background could be alleviated in each layer.

To further integrate the features with different receptive fields, as inspired by (Lin et al. 2017; Liu et al. 2018), we apply LRDA in a multi-scale manner, as shown in Figure 3 (a). Concretely, for a feature map from a certain scale, we enhance its representation by merging it with features from the last scale. After that, we conduct LRDA (Eqn. (1)) to enhance the feature maps further. Finally, the original feature map of the current scale is added back to the enhanced one to maintain backbone information. After conducting CS-LRDA, multi-scale updated features are gained.

Depth-Aware Lifting. Given a LiDAR point cloud $P \in \mathbb{R}^{N \times 3}$, we first project it into the image plane, obtaining a sparse depth map \hat{D} . Considering the original LiDAR point cloud only maps to a small proportion of pixels on the image, we apply the depth completion as (Wu et al. 2022) to generate the dense depth map \hat{D} . Then, we transform the features in the image plane into 3D space assisted by the dense depth map, as shown in Figure 4.

In practice, given the input image with the size of (H, W) , we first generate image coordinates \mathbb{C} according to the dense depth map \hat{D} ,

$$\mathbb{C} = \{(u, v, d) \mid u \in [1, W], v \in [1, H]\}, \quad (4)$$

where $d = \hat{D}_{uv}$. Afterward, we transform the image coordinates \mathbb{C} into the 3D space, utilizing the camera intrinsic and extrinsic matrices $K \in \mathbb{R}^{4 \times 4}$ and $T \in \mathbb{R}^{4 \times 4}$. Specifically, given i -th image coordinate $\mathbb{C}_i = (u_i, v_i, d_i)$, its coordinate (x_i, y_i, z_i) in the world system can be calculated as

$$[x_i, y_i, z_i, 1]^T = T^{-1} \cdot K^{-1} \cdot [u_i \times d_i, v_i \times d_i, d_i, 1]^T. \quad (5)$$

After the above transformation, we lift the multi-scale image features into the 3D space. To unify the image features into a common space, we conduct pillarization (Lang et al. 2019) on the above multi-scale features, obtaining multi-scale BEV features as shown in Figure 4.

Nearest BEV Completion. Due to occlusion and limited field-of-view, the above BEV maps from the camera stream

still contain noticeable empty grids. To tackle this problem, we apply the Nearest BEV Completion to complete the empty BEV areas. For each BEV feature map, we first generate an occupancy map, which records the occupancy status of the current lattices. After that, for each empty lattice, we interpolate it with its nearest neighbor. Meanwhile, we record the replacement and its offset with euclidean distance, and they will be used as additional two-channel concatenated on the original BEV features.

3.3 Bottom-Up BEV Fusion

After top-down BEV generation, we adopt a bottom-up manner for multi-modality fusion, as shown in Figure 2. For each scale, we first concatenate the two feature maps from different modalities and then apply the channel-wise attention (Hu, Shen, and Sun 2018) for single-scale fusion to process the merged feature map. In this way, multi-scale BEV feature maps from two modalities are fused into a single pyramid. Afterward, we leverage the feature fusion strategy in (Chen et al. 2022a) to merge the multi-scale fused feature maps together, obtaining a single fused BEV feature map.

3.4 Prediction and Objective

Following (Chen et al. 2022a), we detect 3D lanes in an anchor-based manner. The 3D prediction \mathbf{Y}'_{3d} contains a regression component $\mathbf{Y}^{3d}_{reg} \in \mathbb{R}^{N_s \times 2}$ and two classification components, $\mathbf{Y}^{3d}_{cls} \in \mathbb{R}^1$ and $\mathbf{Y}^{3d}_{vis} \in \mathbb{R}^{N_s \times 1}$. Here \mathbf{Y}^{3d}_{reg} represents the predicted offsets to the anchor. And \mathbf{Y}^{3d}_{cls} and \mathbf{Y}^{3d}_{vis} denote the category and visibility of the anchor, respectively. Here, N_s denotes the number of sampled points along the y-axis, which is a hyperparameter. To enhance the capability of 2D features, we also adopt auxiliary losses for 2D lane detection, which consists of three parts as the 3D one, *i.e.*, regression, classification, and visibility. Moreover, a semantic segmentation loss is adopted on the fused BEV feature supervised by the projection of 3D annotation, which helps the BEV feature learn the semantics of the lane lines. Overall, the final loss is a weighted sum of the above losses:

$$\mathcal{L}_{all} = \lambda_1 \mathcal{L}_{3d}(\mathbf{Y}'_{3d}, \mathbf{Y}^{gt}_{3d}) + \lambda_2 \mathcal{L}_{2d}(\mathbf{Y}'_{2d}, \mathbf{Y}^{gt}_{2d}) + \quad (6)$$

$$\lambda_3 \mathcal{L}_{seg}(\mathbf{Y}'_b, \mathbf{Y}^{gt}_b) \quad (7)$$

where $\mathbf{Y}^{gt}_{(\cdot)}$ denotes the corresponding ground truth, $\mathbf{Y}'_{(\cdot)}$ denotes the corresponding prediction and $\lambda_{(\cdot)}$ is the weight for a specific task.

4 Experiment

Since OpenLane dataset (Chen et al. 2022a) is the **sole public dataset** containing LiDAR-camera input with corresponding alignment, we conduct all experiments on it.

4.1 Datasets

OpenLane is built on Waymo Open Dataset (Sun et al. 2020a), which contains 200K frames and 880K annotated lanes. Each sequence of Waymo (Sun et al. 2020a) is sampled at 10Hz for 20s with 64-beam LiDARs. Considering OpenLane (Chen et al. 2022a) only annotated the lanes in

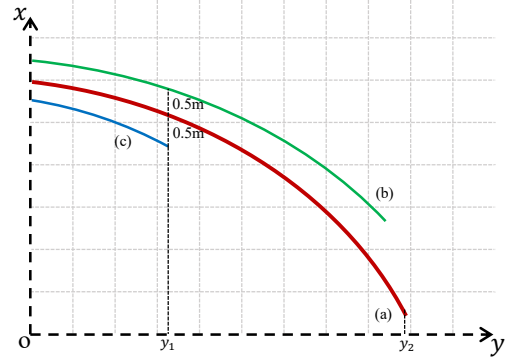


Figure 5: **The failure evaluation in previous works.** Assuming the red, green and blue lines (a, b and c) are ground truth and two different predictions. The error in x-axis increases monotonically as y grows and to 0.5m at y_1 for both b and c. According to the measurement in (Chen et al. 2022a; Guo et al. 2020), line c will be chosen as the best match since its error between y_1 and y_2 is set to be a predefined threshold d_{max} , while its error is larger. We tackle this problem by considering all visible points in the ground truth.

the front-view, we ignore points in LiDAR data that outside the image. In addition, OpenLane provides a category information for each lane, *e.g.*, white dash line, as well as the road curbside. In total, there are 14 categories.

4.2 Evaluation Metrics

Following the official metric of the OpenLane (Chen et al. 2022a), the evaluation of 3D lane detection is formulated as a bipartite matching problem between predictions and ground truth, which is based on *edit distance*. After matching, the corresponding metrics can be calculated in terms of F1 score, category accuracy and x/z error over matched lanes. A right prediction requires the point-wise distances of 75% points between ground truth are less than the max-allowed distance.

In this paper, we modify the original metrics in (Chen et al. 2022a) in a **more challenging** manner. The modifications and reasons are as follows: **1)** We reduce the point-wise max-allowed distance to 0.5m for a successful matching. In the original OpenLane, they set 1.5m as the threshold. However, in the real-world autonomous driving, a 1.5m bias would seriously hamper the safety. **2)** Instead of using 100m as the evaluation range as in (Chen et al. 2022a), we evaluate the 3D lane only inside the furthest distance that point clouds can cover, *i.e.*, 75m in Waymo (Sun et al. 2020b). **3)** We include all points covered in ground truth to conduct point-wise distance measurement, and remain the same $d_p = 1.5m$ for false positive penalization, which further reformulate the original cost as

$$c_i^{pg} = \begin{cases} \|x_i^p - x_i^g\|_2 + \|z_i^p - z_i^g\|_2 & \text{if } v_i^g = 1, \\ d_p & \text{if } v_i^g = 0 \text{ and } v_i^p = 1, \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

where i is the index of the anchor in the y-axis. $(\cdot)^g$ and $(\cdot)^p$ denote the ground truth and the predictive values, respec-

Table 1: **Performance comparison on OpenLane dataset.** L and C represent LiDAR and camera, respectively. [†]3DLaneNet (Garnett et al. 2019) originally predicts camera height and pitch for transformation, while OpenLane has already provided the extrinsic of cameras. Therefore, we keep their prediction as an auxiliary task but exploit ground truth extrinsic during the transformation. *GenLaneNet (Guo et al. 2020) is a two-stage method, where they first train a segmentation network to estimate the foreground and background, and then fix it during second stage training. In our experiments, we directly train their model with their pretrained segmentation network in the first stage. Particularly, the original 3D-LaneNet and Gen-LaneNet only distinguish the lane and background, *i.e.*, without category prediction. For 3D-LaneNet, we modify their predictions as multi-class one and apply the same multi-class loss on it as ours. For Gen-LaneNet, since their pretrained segmentation model is a binary classification network, we do not apply category prediction. Notably, in the table, except X error and Z error, a higher value indicates better performance. The last three columns denote the F1 score under three special cases as (Chen et al. 2022a).

Methods	Modalities	F1	Category Accuracy	X error↓	Z error↓	Up & Down	Curve	Extreme Weather
3DLaneNet [†] (Garnett et al. 2019)	C	42.8	84.5	0.452	0.153	32.7	38.5	43.4
GenLaneNet* (Guo et al. 2020)	C	29.3	-	0.431	0.164	22.8	28.4	27.8
Persformer (Chen et al. 2022a)	C	49.9	86.9	0.419	0.134	40.8	47.0	51.1
M ² -3DLaneNet (ours)	C+L	62.0	90.8	0.236	0.097	53.9	60.8	59.5
<i>Improvement</i>	-	<i>↑12.1</i>	<i>↑3.9</i>	<i>↓0.183</i>	<i>↓0.037</i>	<i>↑13.1</i>	<i>↑12.2</i>	<i>↑8.4</i>

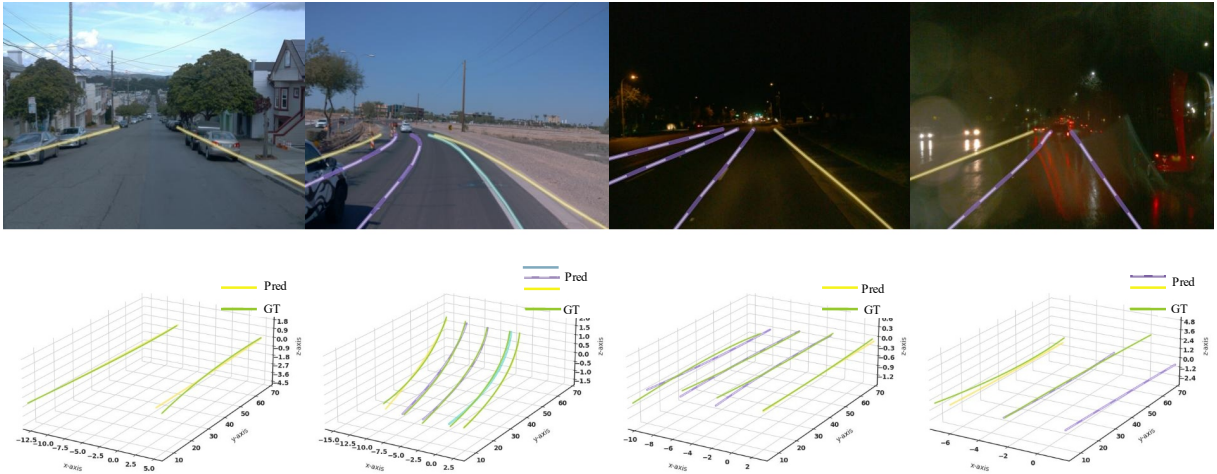


Figure 6: **Visualization.** We show the lane detection of M²-3DLaneNet in the 3D space and the corresponding image in various scenarios, where the green lines represent ground truth and the other colors are determined by different categories. Notably, the yellow line denotes the predicted road curbside, which is a class included in OpenLane.

tively. v^g and v^p indicate the ground truth and predicted visibility, *i.e.*, whether the point lies inside the range of lanes. Notably, in previous works (Chen et al. 2022a; Guo et al. 2020), the cost is calculated only if both ground truth and predicted visibility are accessed. However, it causes the failure evaluation when the model tends to predict the invisible prediction, as illustrated in Figure 5.

4.3 Implementation Details

For all experiments, we use the same hyperparameters and training protocols. Specifically, we use AdamW (Loshchilov and Hutter 2017) as the optimizer, the learning rate is set to 2×10^{-4} at the beginning of training, and we use a cosine annealing scheduler (Loshchilov and Hutter 2016) with $T_{max} = 8$ to update the learning rate. All models are trained with batch size 16 and overall epochs 25. Input im-

ages are resized to 480 x 360 in both training and testing. The data augmentation only includes image random rotation from -10° to 10° . The architecture details will be illustrated in the supplementary material.

4.4 Main Results

We present the main results of our experiments on OpenLane (Chen et al. 2022a) dataset in Table 1. We evaluate 3DLaneNet (Garnett et al. 2019), GenLaneNet (Guo et al. 2020) and Persformer (Chen et al. 2022a) based on their official codes and retrain them on OpenLane dataset. The results show that M²-3DLaneNet effectively utilizes the information of different modalities, which achieves the state-of-the-art performance among all previous methods, surpassing PersFormer (Chen et al. 2022a) by 12.1% in F1 score. We provide visualization of M²-3DLaneNet in Figure 6.

Table 2: **Results with different modalities.** L and C represent LiDAR and camera, respectively. L(-) denotes that the intensity information in the LiDAR point cloud is not used. * We enhance point features with the RGB features as PointPainting (Vora et al. 2020). † We apply AutoAlignV2 (Chen et al. 2022b) based on their paper.

Method	Modalities	F1 score	Cat. Acc.
Camera stream	C	49.9	86.9
LiDAR stream	L(-)	52.2	80.9
LiDAR stream	L	52.6	81.8
PointPainting*	C+L	58.0	85.1
AutoAlignV2†	C+L	57.4	92.0
Ours w/o LiDAR stream	C+L	60.4	90.6
M ² -3DLaneNet(ours)	C+L	62.0	90.8

Table 3: **Ablation studies on TD-BEV and BU-BEV.** The upper part shows the results without the LiDAR stream, while the lower one is illustrated with the LiDAR stream. Here, SSF, LRDA and NBC are single-scale fusion in BU-BEV, line-restricted deformable attention and nearest BEV completion, respectively.

LiDAR stream	SSF	LRDA	NBC	F1
				59.0
		✓		59.7 (+0.7)
			✓	59.6 (+0.6)
		✓	✓	60.4 (+1.4)
✓				60.6 (+1.6)
✓		✓		61.1 (+2.1)
✓			✓	61.3 (+2.3)
✓		✓	✓	61.7 (+2.7)
✓	✓			61.3 (+2.3)
✓	✓	✓	✓	62.0 (+3.0)

4.5 Design Analysis

To understand the effectiveness of our proposed module, we conduct experiments on each component in M²-3DLaneNet.

Results with Different Modalities. We demonstrate the results using different modalities in Table 2. As shown in the table, using the pure camera stream can only achieve a low F1 score of 49.9. When exploiting the LiDAR stream, it has already exceeded the results of the camera stream, even if the intensity information is not used (*i.e.*, L(-)). The reason is that the geometric information in the point cloud provides the model with rich structure information of the scene to perceive the targets. Specifically, the LiDAR provides clear cues as shown in the Figure 7, *i.e.*, the lane lines and road curbs.

In the bottom part of the table, we compare results in different multi-modal manners, in which PointPainting (Vora et al. 2020) and AutoAlignV2 (Chen et al. 2022c) are used as our baselines, where we enhance the point cloud features through painting the RGB features or adaptive fusion. Moreover, we ablate the LiDAR stream (*i.e.*, PointPillars (Lang et al. 2019)) and only exploiting camera stream with TD-BEV and BU-BEV. Here, since the LiDAR information is utilized in the depth-aware lifting, it also be-

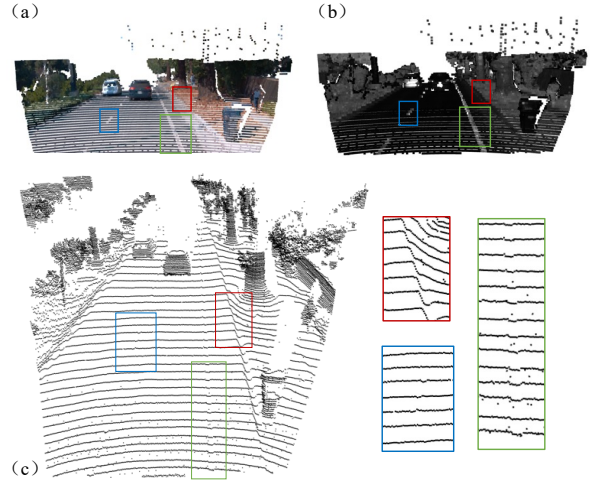


Figure 7: **Why LiDAR works?** (a) shows the point cloud with RGB retrieved from the image and (b) shows the point cloud with intensity. (c) demonstrate the raw point cloud, in which we zoom in on three areas in red, blue, and green boxes. It can be found that the geometry shape of the point cloud changes at the location of lane lines and road curbs.

longs to the multi-modal method and achieves superior results upon single-modal manner. Finally, when using the full architecture, we achieve the state-of-the-art results.

Ablation Study on TD-BEV and BU-BEV. In this section, we verify the effectiveness of TD-BEV and BU-BEV, as shown in Table 3. In the upper part, we illustrate the results without LiDAR stream, the baseline only achieves a low F1 score of 59.0. Note that, in this baseline we replace the LRDA with traditional DA (Zhu et al. 2020) and discard the NBC. Since the depth-aware lifting is still applied, the performance is much higher than the result through the pure camera stream in Table 2. After independently adopting the LRDA and NBC, the performance will be increased by 0.7 and 0.6, and achieve 60.4 (+1.4) through utilizing the both.

In the lower part, we demonstrate the results with the LiDAR stream. Similarly, both LRDA and NBC can greatly enhance the performance, respectively gaining +0.5 and +0.8 boosts upon the baseline (60.6). After exploiting the both components, it archives 61.7 F1 score with +1.1 increase. Finally, single-scale fusion in BU-BEV increases the performance by about 0.7. After assembling all the components, our M²-3DLaneNet gains the best performance.

5 Conclusions

In this work, we present M²-3DLaneNet, a novel Multi-Modal 3D Lane Detection framework that utilizes camera and LiDAR for recognition. By lifting image through generated dense depth map and fuse features in BEV space, M²-3DLaneNet effectively extracts useful feature from different modalities. M²-3DLaneNet also contains a Line-Restricted Deform-Attention and the Nearest BEV Completion to further improve the performance. The architecture is validated on OpenLane dataset and outperforms all previous work.

6 Acknowledgment

This work was supported in part by NSFC-Youth 61902335, by HZQB-KCZYX-2021067, by the National Key R&D Program of China with grant No.2018YFB1800800, by Shenzhen Outstanding Talents Training Fund, by Guangdong Research Project No.2017ZT07X152, by Guangdong Regional Joint Fund-Key Projects 2019B1515120039, by the NSFC 61931024&81922046, by helixon biotechnology company Fund and CCF-Tencent Open Fund.

References

- Bai, M.; Mattyus, G.; Homayounfar, N.; Wang, S.; Lakshminanth, S. K.; and Urtasun, R. 2018. Deep multi-sensor lane detection. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3102–3109. IEEE.
- Bai, X.; Hu, Z.; Zhu, X.; Huang, Q.; Chen, Y.; Fu, H.; and Tai, C.-L. 2022. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1090–1099.
- Chen, L.; Sima, C.; Li, Y.; Zheng, Z.; Xu, J.; Geng, X.; Li, H.; He, C.; Shi, J.; Qiao, Y.; and Yan, J. 2022a. PersFormer: 3D Lane Detection via Perspective Transformer and the OpenLane Benchmark. In *European Conference on Computer Vision (ECCV)*.
- Chen, Z.; Li, Z.; Zhang, S.; Fang, L.; Jiang, Q.; and Zhao, F. 2022b. AutoAlignV2: Deformable Feature Aggregation for Dynamic Multi-Modal 3D Object Detection. *arXiv preprint arXiv:2207.10316*.
- Chen, Z.; Li, Z.; Zhang, S.; Fang, L.; Jiang, Q.; Zhao, F.; Zhou, B.; and Zhao, H. 2022c. AutoAlign: Pixel-Instance Feature Aggregation for Multi-Modal 3D Object Detection. *arXiv preprint arXiv:2201.06493*.
- Efrat, N.; Bluvstein, M.; Oron, S.; Levi, D.; Garnett, N.; and Shlomo, B. E. 2020. 3d-lanenet+: Anchor free lane detection using a semi-local representation. *arXiv preprint arXiv:2011.01535*.
- Feng, Z.; Guo, S.; Tan, X.; Xu, K.; Wang, M.; and Ma, L. 2022. Rethinking Efficient Lane Detection via Curve Modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17062–17070.
- Garnett, N.; Cohen, R.; Pe’er, T.; Lahav, R.; and Levi, D. 2019. 3d-lanenet: end-to-end 3d multiple lane detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2921–2930.
- Gopalan, R.; Hong, T.; Shneier, M.; and Chellappa, R. 2012. A learning approach towards detection and tracking of lane markings. *IEEE Transactions on Intelligent Transportation Systems*, 13(3): 1088–1098.
- Guo, Y.; Chen, G.; Zhao, P.; Zhang, W.; Miao, J.; Wang, J.; and Choe, T. E. 2020. Gen-lanenet: A generalized and scalable approach for 3d lane detection. In *European Conference on Computer Vision*, 666–681. Springer.
- Han, J.; Deng, X.; Cai, X.; Yang, Z.; Xu, H.; Xu, C.; and Liang, X. 2022. Laneformer: Object-aware Row-Column Transformers for Lane Detection. *arXiv preprint arXiv:2203.09830*.
- Hou, Y.; Ma, Z.; Liu, C.; and Loy, C. C. 2019. Learning lightweight lane detection cnns by self attention distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1013–1021.
- Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141.
- Jung, J.; and Bae, S.-H. 2018. Real-time road lane detection in urban areas using LiDAR data. *Electronics*, 7(11): 276.
- Kammel, S.; and Pitzer, B. 2008. Lidar-based lane marker detection and mapping. In *2008 IEEE intelligent vehicles symposium*, 1137–1142. IEEE.
- Lang, A. H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; and Beijbom, O. 2019. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12697–12705.
- Lee, S.; Kim, J.; Shin Yoon, J.; Shin, S.; Bailo, O.; Kim, N.; Lee, T.-H.; Seok Hong, H.; Han, S.-H.; and So Kweon, I. 2017. Vpgnet: Vanishing point guided network for lane and road marking detection and recognition. In *Proceedings of the IEEE international conference on computer vision*, 1947–1955.
- Li, J.; Mei, X.; Prokhorov, D.; and Tao, D. 2016. Deep neural network for structural prediction and lane detection in traffic scene. *IEEE transactions on neural networks and learning systems*, 28(3): 690–703.
- Li, X.; Li, J.; Hu, X.; and Yang, J. 2019. Line-cnn: End-to-end traffic line detection with line proposal unit. *IEEE Transactions on Intelligent Transportation Systems*, 21(1): 248–258.
- Li, Y.; Yu, A. W.; Meng, T.; Caine, B.; Ngiam, J.; Peng, D.; Shen, J.; Lu, Y.; Zhou, D.; Le, Q. V.; et al. 2022. Deep-fusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17182–17191.
- Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.
- Liu, L.; Chen, X.; Zhu, S.; and Tan, P. 2021a. Condlanenet: a top-to-down lane detection framework based on conditional convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3773–3782.
- Liu, R.; Chen, D.; Liu, T.; Xiong, Z.; and Yuan, Z. 2022. Learning to predict 3d lane shape and camera pose from a single image via geometry constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 1765–1772.
- Liu, R.; Yuan, Z.; Liu, T.; and Xiong, Z. 2021b. End-to-end lane shape prediction with transformers. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 3694–3702.

- Liu, S.; Qi, L.; Qin, H.; Shi, J.; and Jia, J. 2018. Path aggregation network for instance segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, 8759–8768.
- Loshchilov, I.; and Hutter, F. 2016. Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983.
- Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101.
- Neven, D.; De Brabandere, B.; Georgoulis, S.; Proesmans, M.; and Van Gool, L. 2018. Towards end-to-end lane detection: an instance segmentation approach. In 2018 IEEE intelligent vehicles symposium (IV), 286–291. IEEE.
- Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, 652–660.
- Qin, Z.; Wang, H.; and Li, X. 2020. Ultra fast structure-aware deep lane detection. In European Conference on Computer Vision, 276–291. Springer.
- Qu, Z.; Jin, H.; Zhou, Y.; Yang, Z.; and Zhang, W. 2021. Focus on local: Detecting lane marker from bottom up via key point. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 14122–14130.
- Sun, P.; Kretschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. 2020a. Scalability in perception for autonomous driving: Waymo open dataset. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2446–2454.
- Sun, P.; Kretschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. 2020b. Scalability in perception for autonomous driving: Waymo open dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2446–2454.
- Tabelini, L.; Berriel, R.; Paixao, T. M.; Badue, C.; De Souza, A. F.; and Oliveira-Santos, T. 2021a. Keep your eyes on the lane: Real-time attention-guided lane detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 294–302.
- Tabelini, L.; Berriel, R.; Paixao, T. M.; Badue, C.; De Souza, A. F.; and Oliveira-Santos, T. 2021b. PolyLaneNet: Lane estimation via deep polynomial regression. In 2020 25th International Conference on Pattern Recognition (ICPR), 6150–6156. IEEE.
- Tan, M.; and Le, Q. 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning, 6105–6114. PMLR.
- Thuy, M.; and León, F. 2010. Lane detection and tracking based on lidar data. Metrology and Measurement Systems, (3).
- Torres, L. T.; Berriel, R. F.; Paixão, T. M.; Badue, C.; De Souza, A. F.; and Oliveira-Santos, T. 2020. Keep your Eyes on the Lane: Attention-guided Lane Detection. CoRR.
- Vora, S.; Lang, A. H.; Helou, B.; and Beijbom, O. 2020. Pointpainting: Sequential fusion for 3d object detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 4604–4612.
- Wang, J.; Ma, Y.; Huang, S.; Hui, T.; Wang, F.; Qian, C.; and Zhang, T. 2022. A Keypoint-based Global Association Network for Lane Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1392–1401.
- Wu, D.; Liao, M.; Zhang, W.; and Wang, X. 2021. Yolop: You only look once for panoptic driving perception. arXiv preprint arXiv:2108.11250.
- Wu, X.; Peng, L.; Yang, H.; Xie, L.; Huang, C.; Deng, C.; Liu, H.; and Cai, D. 2022. Sparse Fuse Dense: Towards High Quality 3D Detection with Depth Completion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5418–5427.
- Yan, F.; Nie, M.; Cai, X.; Han, J.; Xu, H.; Yang, Z.; Ye, C.; Fu, Y.; Mi, M. B.; and Zhang, L. 2022. ONCE-3DLanes: Building Monocular 3D Lane Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 17143–17152.
- Zhang, J.; Xu, Y.; Ni, B.; and Duan, Z. 2018. Geometric constrained joint lane segmentation and lane boundary detection. In proceedings of the european conference on computer vision (ECCV), 486–502.
- Zheng, T.; Huang, Y.; Liu, Y.; Tang, W.; Yang, Z.; Cai, D.; and He, X. 2022. CLRNNet: Cross Layer Refinement Network for Lane Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 898–907.
- Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2020. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159.